



ECOLE KHAWARIZMI
DE L'INFORMATIQUE & DE GESTION
TEL : 036 84 68 46

APPRENDRE

DELPHI 7

PAR L'EXEMPLE

SOMMAIRE

Chapitre 1 : Introduction à Delphi7	4
Leçon1 : Delphi7, généralités	5
Leçon2 : Les premiers pas.....	9
TP1 : Jeu de couleurs	9
TP2 : Calcul de la moyenne	11
TP3 : Calcul du factoriel	12
TP4 : Chargement d'une image	13
Leçon3 : Avancer plus	14
TP1 : Interface des couleurs.....	14
TP2 : Lancement d'un exécutable à partir d'une fiche Delphi	16
TP3 : Les commentaires	16
TP4 : La manipulation de l'interface par clavier	17
Leçon4 : Messages, Multimedia et autres	18
TP1 : Les messages de dialogue	18
TP2 : Les fichiers multimedia.....	19
TP3 : Les animations	19
TP4 : Les boucles de temps	20
TP5 : Dessiner une fonction.....	20
TP6 : Barre de progression	20
Leçon5 : Menu principal et barres d'outils	21
TP1 : Le menu principal	21
TP2 : Les barres d'outils	22
Leçon6 : La programmation des bases de données.....	23
TP1 : Création d'un alias de base de données.....	23
TP2 : Création d'une table	24
TP3 : Connexion et affichage d'une base de données dans une fiche Delphi	25
TP4 : Les champs calculés et les boucles dans une BD (exemple : facture)	25
TP5 : Les champs calculés et les boucles dans une BD (exemple : Bulletin)	28
TP6 : Le DBNavigator	29
Leçon7 : Les index secondaires et leurs utilisation	30
TP1 : La création des index secondaires.....	30
TP2 : Le tri d'une table	31
TP3 : le filtrage d'une table	32
TP4 : la recherche simple dans une table	33
TP5 : les relations maître-détail	33
Leçon8 : Introduction aux requêtes SQL	35
Présentation générale	35
TP1 : Les requêtes SQL les plus utilisées	36
TP2 : La consultation par des requêtes paramétrées.....	38
TP3 : Les graphes.....	39
Leçon9 : Les états d'impression	40
Introduction	40
TP1 : imprimer les enregistrements d'une table (liste des produits)	41
Leçon10 : Delphi est un RAD.....	43
TP1 : une fiche simple d'impression d'une table	43
TP2 : Une fiche simple de connexion et d'affichage et d'impression d'une table	44
TP3 : Une fiche maître-détail entre deux tables	45
TP4 : TP Exemple	45

Chapitre 2 : Logiciel gestion commerciale.....	46
Préparation du Logiciel	47
Fiche Menu	48
Fiche Produit.....	50
Fiche Client	52
Fiche Fournisseur	54
Fiche Facture Achat	56
Fiche Facture Vente	61
Fiche Consultation Achats	67
Fiche Consultation Ventes	69
Fiche Statistiques Reglement.....	71
Fiche Mot de Passe	72
Fiche Protection	73
Fiche Archivage	74
Fiche Inventaire	75
Fiche Cloture	77
Fiche À propos	79
Chapitre 3 : Logiciel gestion scolarité.....	80
Préparation du Logiciel	81
Fiche Menu	82
Fiche Etudiant	83
Fiche Matière	87
Fiche Professeur	90
Fiche Spécialité	92
Fiche Bulletin	94
Fiche Consultation Etudiant	99
Fiche Statistiques des Étudiants	100
Fiche Impression.....	103
Fiche Mot de Passe	105
Fiche Protection	106
Fiche Archivage	107
Fiche Cloture	108
Fiche À propos	109
Chapitre 4 : Trucs & astuces	111
Créer une installation	112
Conception des fichiers help pour les logiciels.....	114
Le skin Data	115
Annexe	117
Les composants de base (Propriétés et venements)	118

CHAPITRE 1

INTRODUCTION

A

DELPHI7

LEÇON 1 : DELPHI7, GÉNÉRALITÉS

1- INTRODUCTION

Delphi est un environnement de programmation visuel orienté objet permettant de développer des applications sous Windows. Il représente la suite logique de la famille turbo Pascal avec ses nombreuses versions (précisément le pascal objet). Delphi est un outil moderne, puissant, faisant appel à une conception visuelle des applications, à la programmation orientée objet, à une bibliothèque de composants très riche (la VCL: Visual Components Library), aux fichiers DLL (Dynamic Link Library) et API (Application Programming Interface) de Windows; Delphi se classe comme l'un des meilleurs environnement de développement rapide des applications (RAD) dans le monde informatique.

Delphi, c'est aussi un produit qui évolue avec le temps. Chaque nouvelle version possède son lot de nouveautés. Voici les versions successives :

- Delphi 1 est le premier environnement pascal RAD sous Windows 3.11 (visuel).
- Delphi 2 marque le passage au mode 32 bits avec Windows 95.
- Delphi 3 supporte déjà les Active X, ainsi que l'aide à la saisie dynamique.
- Delphi 4 propose les tableaux dynamiques.
- Delphi 5 ouvre l'accès natif aux bases de données ADO et interbase.
- Delphi 6 introduit dbExpress et les composants portables de la CLX.

Et en fin, Delphi 7, Delphi 8 et récemment Delphi 2006 et 2007 qui sont sans doute les versions les plus enrichies de toute l'histoire du produit.

2- ENVIRONNEMENT DELPHI

Démarrage de Delphi

Vous pouvez démarrer Delphi de plusieurs façons :

- Double-cliquer sur l'icône Delphi (si vous avez créé un raccourci).
- Choisir Programmes ⇒ Borland Delphi 7 ⇒ Delphi 7 dans le menu Démarrer de Windows.
- Choisir Exécuter dans le menu Démarrer de Windows, puis entrer Delphi32.
- Double-cliquer sur Delphi32.exe dans le répertoire Delphi\Bin.

L'EDI

Lorsque vous démarrez Delphi pour la première fois, vous voyez apparaître quelques-uns des principaux outils proposés dans l'EDI (environnement de développement intégré). Dans Delphi, l'EDI comprend des barres d'outils, des menus, la palette de composants, l'inspecteur d'objets, l'arborescence d'objets, l'éditeur de code, l'explorateur de code, le gestionnaire de projet et bien d'autres outils.

La vue arborescente de l'objet affiche une représentation hiérarchique des relations parent enfant des composants.

La barre des menus et les barres d'outils permettent d'accéder à tout un ensemble de fonctions qui vous aident à écrire vos applications.

La palette de composants contient des composants (visuels ou non) prêts à être utilisés dans vos projets, regroupés par familles (pages à onglets). C'est la VCL de Delphi.

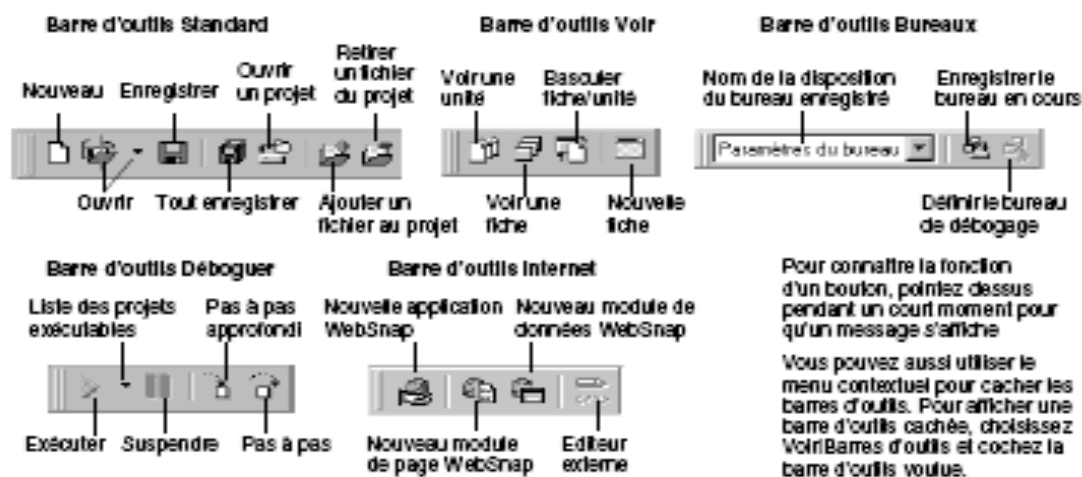
L'éditeur de code, ou l'unité, permet d'afficher et de modifier le code source de votre application. La saisie du code source est assistée (semi-automatique).

Le concepteur de fiche contient une fiche vierge pour démarrer la conception de l'interface utilisateur de votre application. C'est le support des composants utilisés. Une application peut comporter plusieurs fiches (fenêtres). La bascule fiche/unité se fait par la touche **F12**

L'inspecteur d'objets permet de changer des propriétés et d'utiliser le gestionnaire d'événements de l'objet sélectionné. On affiche l'inspecteur d'objet par **F11** ou **ENTREE**

L'explorateur de code montre les classes, variables et routines de votre unité et vous permet de naviguer rapidement.

Les barres d'outils de Delphi permettent un accès rapide aux opérations et aux commandes les plus utilisées. La plupart des opérations accessibles dans les barres d'outils se retrouvent dans les menus déroulants.



De nombreuses opérations possèdent des équivalents clavier ainsi que des boutons dans la barre d'outils. Lorsqu'un raccourci clavier est disponible, il est toujours inscrit à côté de la commande dans le menu déroulant.

	Voir la liste des unités (Ctrl + F12)		Exécuter (F9)		Tout enregistrer (Le projet et les fichiers associés)
	voir la liste des fiches (Shift + F12)		Ajouter un fichier au projet		Enregistrer (enregistrer uniquement la fiche actuelle)
	Basculer Unité/Fiche (F12)		Retirer un fichier du projet		Ouvrir un fichier (fiche ou projet)
	Nouvelle Fiche		Ouvrir un projet		Nouveau (fenêtre de choix: Application, console, fiche, ...etc.)

L'inspecteur d'objet

Chaque composant possède des attributs spécifiques (propriétés, événements et méthodes) vous permettant de contrôler votre application. Utilisez l'inspecteur d'objets pour définir des propriétés de conception, créer des gestionnaires d'événements, filtrer la visibilité des propriétés et des événements, ce qui établira le lien entre l'apparence visuelle de votre application et le code qui la fait s'exécuter.

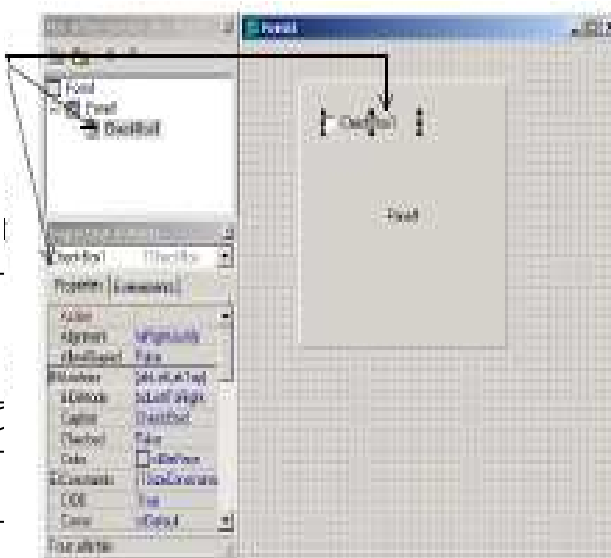
Vous pouvez changer la manière dont un composant s'affiche (propriétés) et se comporte (événements) dans votre application en utilisant l'inspecteur d'objets. Lorsqu'un composant est sélectionné sur la fiche, ses propriétés et ses événements sont affichés dans l'inspecteur d'objets.

L'arborescence d'objets, l'inspecteur d'objets et le concepteur de fiche fonctionnent ensemble. Quand vous cliquez sur un objet de votre fiche, la focalisation change à la fois dans l'arborescence et dans l'inspecteur d'objets et réciproquement.

Appuyez sur **ALT-SHIFT-F11** pour passer la focalisation à l'arborescence d'objets.

Utiliser cette liste déroulante pour sélectionner un objet. Si un objet est sélectionné, les membres de sa classe (propriétés et événements) seront affichés.

Sélectionnez une propriété et changez sa valeur dans la colonne de droite.



3- NOTIONS DE PROJET DELPHI ET DE SES FICHIERS

Un projet est l'état non compilé d'une application (Application = Logiciel). Chaque projet compilé devient une application. Un projet se compose d'un certain nombre de fichiers. Une sage habitude est de consacrer complètement un répertoire (dossier) à chaque application qu'on souhaite programmer (chaque projet). Ce répertoire contiendra tous les fichiers constituant le projet (le nombre de fichiers augmentera au fur et à mesure que le projet s'étoffera).

Un projet non compilé (c'est-à-dire avant sa transformation en logiciel), contient ces fenêtres (à l'état non compilées également) : les fiches, c'est la partie visuelle d'une application. Cette fiche, lors de la compilation du projet en application, sera transformée en fenêtre.

A chaque fiche est adjointe une (et une seule) unité, c'est-à-dire un texte écrit en langage Pascal, qui contiendra tout ce qui se rapporte à cette fiche : ce qu'elle contient (boutons, menus, ...), ce qui doit se passer dans certaines situations (lorsqu'on clique sur un bouton par exemple), et encore bien d'autres choses.

Chaque fiche est stockée dans un fichier comportant l'extension 'DFM' (exemple : bulletin.dfm).

Chaque unité est stockée dans un fichier comportant l'extension 'PAS' (exemple : bulletin.pas).

La fiche et son unité associée portent le même nom (mais pas la même extension) (Delphi ne demande ce nom d'enregistrement qu'une seule fois et l'utilise automatiquement pour l'unité et la fiche, ainsi vous n'avez aucun risque de vous tromper). Par exemple, si une fiche est stockée dans le fichier nommé « bulletin.dfm », son unité associée sera stockée dans le fichier nommé « bulletin.pas ». voir la figure suivante :

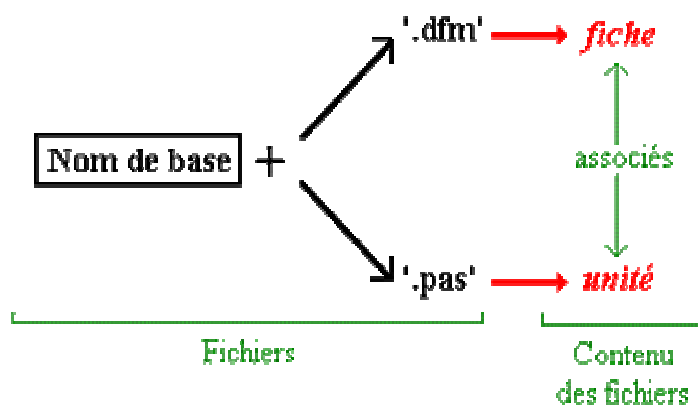


Figure : la nomination des fches & unités

Il est également possible d'utiliser des unités qui n'ont pas de fiche associée. Ceci est utile par exemple pour rassembler des morceaux de programme qui n'ont aucun rapport avec une quelconque fiche. Le fichier qui contient cette unité porte l'extension 'PAS', comme s'il avait une fiche associée, mais il n'en a pas : il n'y a pas de fichier nommé identiquement, mais avec l'extension 'DFM'.

Un projet sous Delphi est constitué d'un fichier-projet (portant l'extension 'DPR'), d'unités et de fiches. Chaque fiche a son unité correspondante, mais certaines unités n'ont pas, comme on l'a vu ci-dessus, de fiche associée (il y a toujours au moins autant d'unités que de fiches dans un projet, et souvent plus). Pensez en outre à donner des noms descriptifs, ou au moins significatifs lors des enregistrements des unités ou des fiches (Ainsi, « unit1.pas » est un bien mauvais choix !).

Lors de la compilation du projet (transformation en application), d'autres fichiers seront créés : des fichiers DCU (fichiers portant l'extension 'DCU', essayez de vous habituer à ces abus de langage). Ces fichiers sont la forme compilée des fichiers de même nom qui portent les extensions 'PAS' et 'DFM' : un 'PAS' (avec son 'DFM' associé s'il y en a un) est transformé en 'DCU' lors de la compilation. Un fichier 'EXE' est enfin créé si la compilation atteint son terme (si tous les fichiers 'DCU' ont pu être créés). Ce fichier est nommé avec le même nom de base que celui du fichier projet (fichier DPR).

Le schéma ci-dessous représente la transformation d'un projet en application, avec les éléments que vous devez désormais connaître. Ce projet comporte deux fiches (avec les unités associées) ainsi qu'une unité seule :

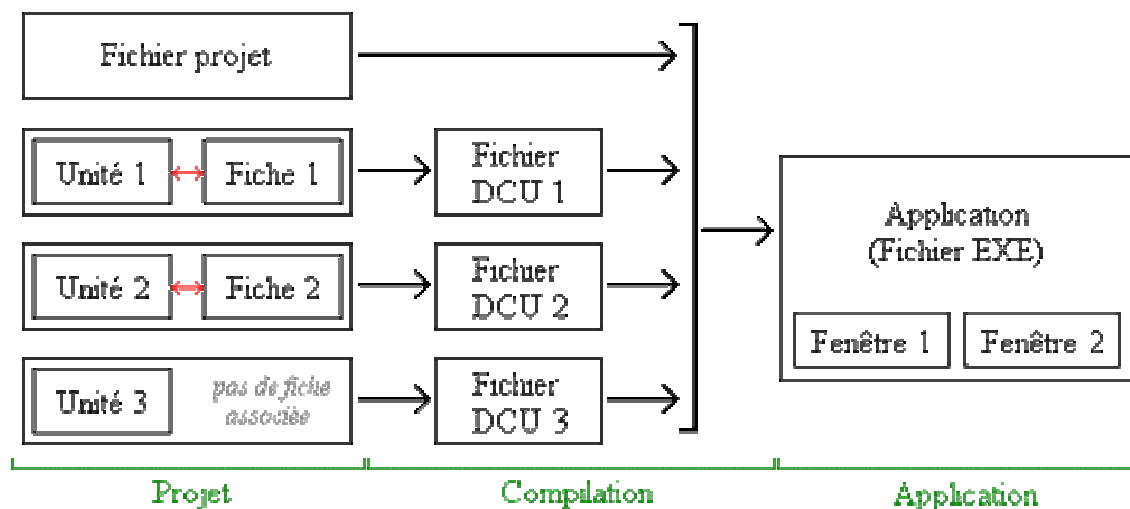


Figure : un projet Delphi, du source à l'exécution.

Lors des enregistrements successifs d'un projet, d'autres fichiers avec des extensions ~DP, ~DF, ~PA, ~DC, ~DPR, ~DFM, ~PAS, ~DCU sont créés : ce sont des copies de sauvegarde (des versions plus anciennes) des fichiers portant le même nom, sans le signe tilde (~)

Exemple : « optionsaffich.~PA » est une copie de sauvegarde de « optionsaffich.pas ».

Enfin, d'autres fichiers ayant le même nom que le fichier projet (celui qui porte l'extension DPR) mais avec l'extension .RES, .OPT, .CFG, .DOF, .DSK, .DSM sont créés dans certaines circonstances. Vous n'avez pas à vous soucier de ces fichiers.

Le tableau suivant donne une liste aussi complète que possible des fichiers pouvant être rencontrés dans le répertoire d'un projet Delphi :

Extension du fichier	Description et Commentaires
DPR	(D elphi P ROject) Contient l'unité principale du projet
PAS	(P AScal) Contient une unité écrite en Pascal. Peut avoir un .DFM correspondant
DFM	(D elphi F or M : fiche Delphi) Contient une fiche (une fenêtre). Le .PAS correspondant contient toutes les informations relatives au fonctionnement de cette fiche, tandis que le .DFM contient la structure de la fiche (ce qu'elle contient, sa taille, sa position, ...). Sous Delphi 5, les .DFM deviennent des fichiers texte qu'il est possible de visualiser et de modifier. La même manipulation est plus délicate mais possible sous Delphi 2 à 4.
DCU	(D elphi C ompiled U nit : Unité compilée Delphi) Forme compilée et combinée d'un .PAS et d'un .DFM optionnel
~???	Tous les fichiers dont l'extension commence par ~ sont des fichiers de sauvegarde, pouvant être effacés pour faire place propre.
EXE	Fichier exécutable de l'application. Ce fichier est le résultat final de la compilation et fonctionne sous Windows exclusivement. Pour distribuer le logiciel, copier ce fichier est souvent suffisant.
RES	(R ESsource) Fichier contenant les ressources de l'application, tel son icône. Ce fichier peut être édité avec l'éditeur d'images de Delphi. Ces notions seront abordées plus loin dans ce guide.
DOF DSK CFG	Fichiers d'options : suivant les versions de Delphi, ces fichiers contiennent les options du projet, les options d'affichage de Delphi pour ce projet, ...

LEÇON 2 : LES PREMIERS PAS

TP 1 : JEU DE COULEURS

Interface



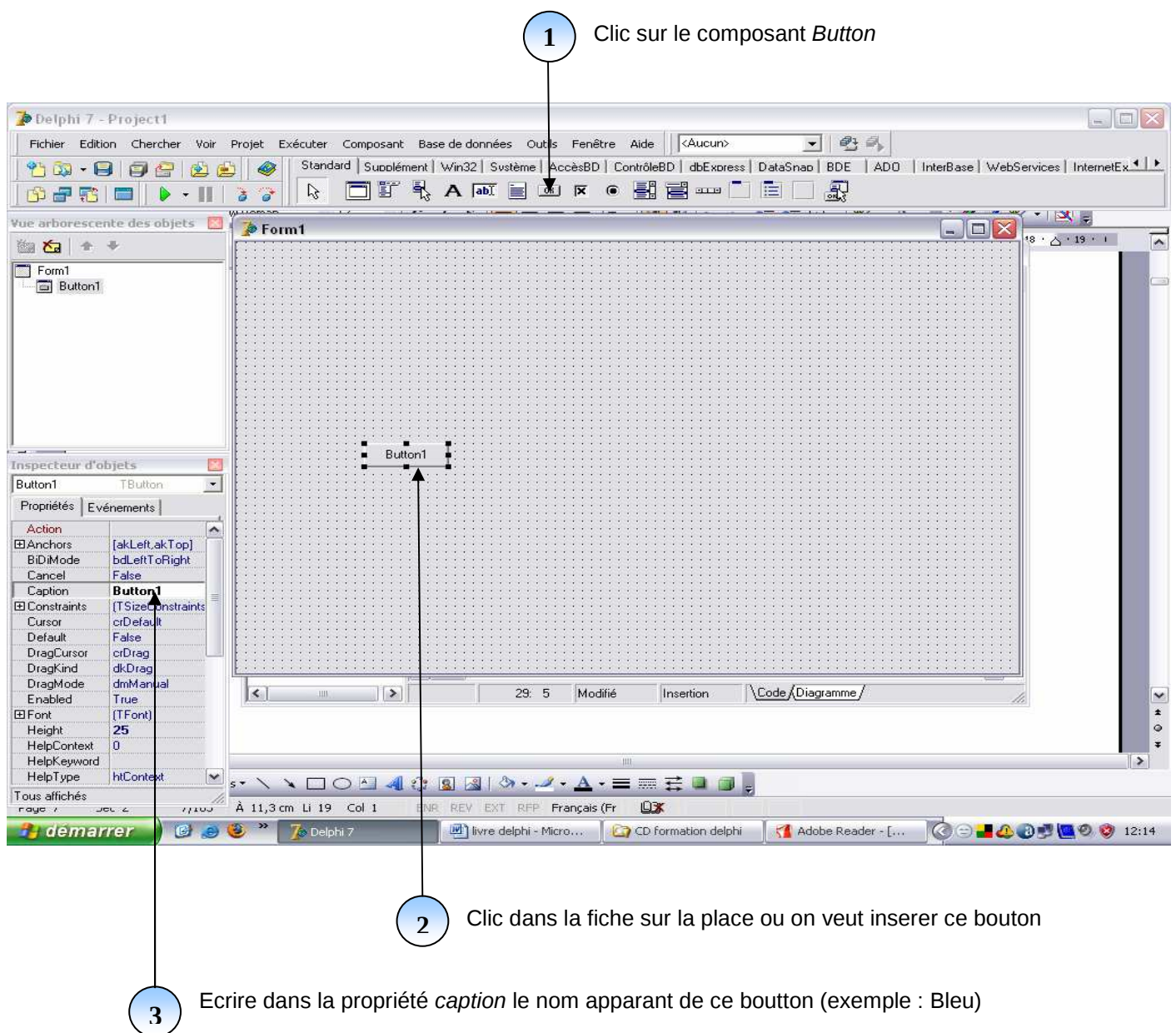
Étapes :

Insérer le premier bouton dans la fiche comme suite :

1 Clic sur le composant *Button*

2 Clic dans la fiche sur la place où on veut insérer ce bouton

3 Ecrire dans la propriété *caption* le nom apparaissant de ce bouton (exemple : Bleu)



Remarque de base

Pour simplifier la présentation des étapes de création des interfaces des exemples suivants, on présente les objets à insérer sur une fiche par un tableau contenant : les noms des composants, leurs propriétés à manipuler (dans leurs inspecteurs d'objets) et enfin les valeurs à donner à ces propriétés.

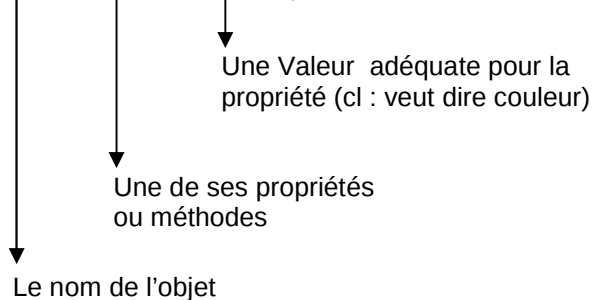
Donc, l'étape précédente peut être présentée comme suite :

1-insérer les composants suivants de la palette *Standard* :

Composant	Propriété	Valeur
Button1	Caption	bleu
Button2	Caption	rouge
Button3	Caption	vert

2-double clique sur le bouton1 (Bleu) et écrire dans la procédure :

Form1.Color:=CLBlue;

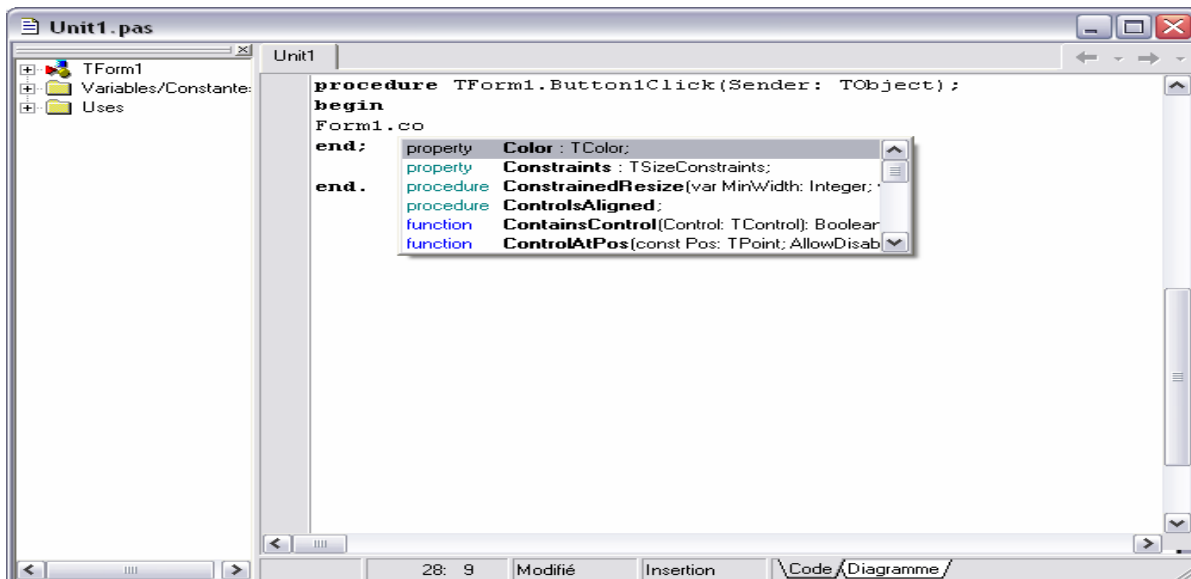


Remarque

Si vous écrivez :

Form1.

Delphi, grâce à son assistant de saisie semi-automatique, vous propose la liste des membres (propriétés et méthodes) de la classe saisie (Form1), et on peut soit choisir dans la liste la propriété ou méthode voulue, ou taper les premières lettres de l'élément voulu (exemple : **co** pour écrire **color**) et lorsqu'il sera sélectionné dans la liste on tape *Entrée* pour que Delphi l'écrive dans l'instruction.



3- taper **F12** pour revenir à la fiche et double clique sur le bouton2 (Rouge) et écrire dans la procédure :

Form1.color:=CLRed;

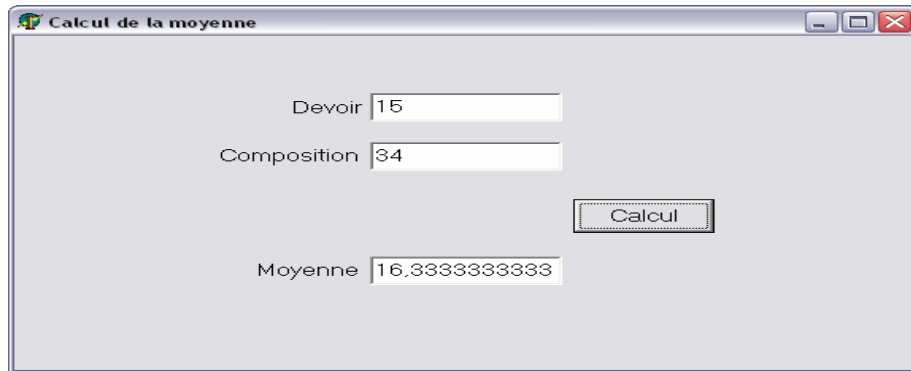
4- taper **F12** pour revenir à la fiche et double clique sur le bouton3 (Vert) et écrire dans la procédure :

Form1.color:=CLGreen;

5- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et vérifier le bon fonctionnement de la fiche.

TP 2 : CALCUL DE LA MOYENNE

Interface



Étapes :

1-insérer les composants suivants de la palette Standard :

Composant	Propriété	Valeur
Edit1	text	
Edit2	text	
Edit3	text	
Label1	Caption	Devoir
Label2	Caption	Composition
Label3	Caption	Moyenne
Button1	Caption	Calcul

2- double clique sur le bouton1 (calcul) et écrire dans la procédure :

edit3.text:= floattostr ((strtofloat (edit1.text) + strtofloat (edit2.text))/3);

3- Lancer l'exécution par la touche F9, ou par le bouton ; Et verifier le bon fonctionnement de la fiche.

Remarque 1:

1- le casting=changement de type =conversion de type (strtofloat, floattostr, inttostr, strtoint)

2- une autre solution possible (à écrire dans la procédure du bouton1 "calcul")

(Supprimer le **begin** et **end** de la procédure et coller tous le code source suivant)

```
Var    dev,comp,moy :real ;
Begin
dev:= strtofloat(edit1.text);
comp:= strtofloat(edit2.text);
moy:= (dev+comp)/3;
edit3.text:= floattostr(moy);
End ;
```

Remarque 2 :

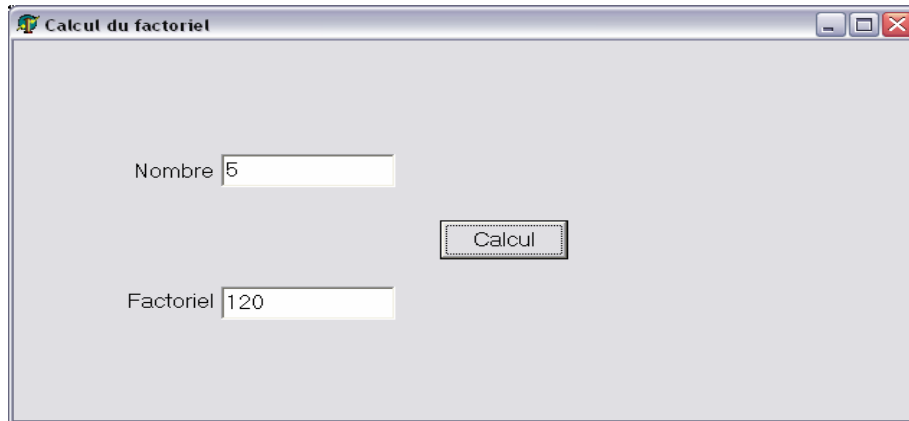
Le Casting = c'est aussi la conversion de type dans une affectation ou un test.

Exemple:

```
Edit3.text :=floattostr((strtofloat(edit1.text) + strtofloat(edit2.text))/3) ;
If strtofloat(edit3.text)<10 then
Showmessage('étudiant ajourné')
Else
Showmessage('étudiant admis') ;
```

TP3 : CALCUL DU FACTORIEL

Interface:



Étapes

1-insérer les composants suivants de la palette standard:

Composant	Propriété	Valeur
Edit1	Text	
Edit2	Text	
Label1	Caption	Nombre
Label2	Caption	Factoriel
Button1	Caption	Calcul

2-double clique sur le bouton1 (calcul) et écrire dans la procédure :
(Supprimer le **begin** et **end** de la procédure et coller tous le code source suivant)

```
var a,b,fact :integer;
Begin
a:= strtoint(edit1.text);
if a>=0 then
begin
fact:=1;
for b:=1 to a do
fact:=fact*b;
edit2.text:= inttostr (fact);
End
Else
Showmessage ('pas de factoriel pour un nombre négatif');
end;
```

3- Lancer l'exécution par la touche F9, ou par le bouton  ; Et verifier le bon fonctionnement de la fiche.

TP4 : CHARGEMENT D'UNE IMAGE

Interface:



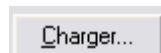
Étapes

1- insérer un composant **image** de la palette « supplément ».

2- choisir dans son inspecteur d'objet la propriété :

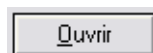


3- dans la fenetre qui s'affiche, on clic sur le bouton :

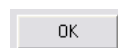


Pour chercher l'image voulue dans le disque dur.

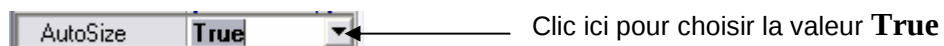
3- dans la nouvelle fenetre, et après avoir selectionner l'image voulue on clic sur le bouton :



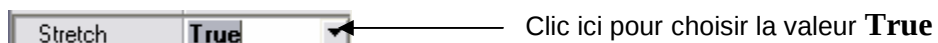
Ensuite sur le bouton :



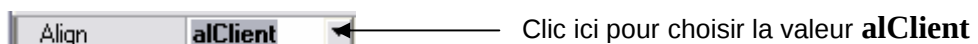
4- pour que l'image occupe toute la surface de son cadre on modifie sa propriété :



5- pour que l'image s'agrandie si elle a une taille plus petite que la surface de son cadre on modifie sa propriété :



Aussi modifier sa propriété :

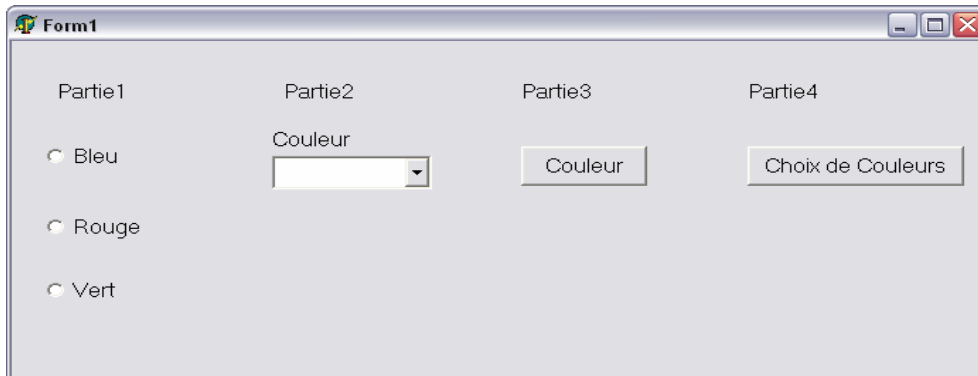


Pour que cette image aucupe tous l'espace de la fenetre actuelle.

LEÇON 3 : AVANCER PLUS

TP1 : INTERFACE DE COULEURS

Interface :



Étapes :

Partie1 :

1- insérer les composants suivants de la palette standard:

Composant	Propriété	Valeur
Radiobutton1	Caption	Bleu
Radiobutton2	Caption	Rouge
Radiobutton3	Caption	Vert

2- double clique sur le Radiobutton1 (Bleu) et écrire dans la procédure :

```
Form1.color:=clblue;
```

3- taper **F12** et double clique sur le Radiobutton2 (ROUGE) et écrire dans la procédure :

```
Form1.color:=clred;
```

4- taper **F12** et double clique sur le Radiobutton3 (Vert) et écrire dans la procédure :

```
Form1.color:=clgreen;
```

5- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et verifier le bon fonctionnement de la fiche.

Partie 2 :

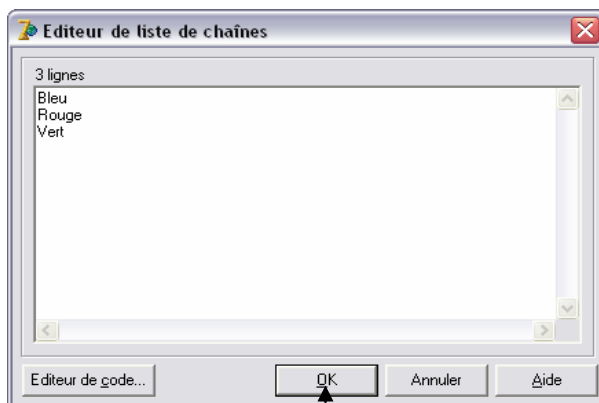
1- insérer un composant Label1 (palette standard) et écrire dans sa propriété caption : couleur.

2- insérer un composant Combobox1 (palette standard) et écrire sa propriété text : (vide).

3- clique sur le composant Combobox1 et dans son inspecteur d'objet on choisit la propriété :



Et dans la fenêtre qui s'affiche on écrit les éléments de la liste ligne par ligne (bleu, rouge, vert) comme suite :



4- double clique sur le Combobox1 et écrire dans la procédure (attention au lettres majuscules):

```

if combobox1.text='Bleu' then
form1.color:=clblue;
if combobox1.text='Rouge' then
form1.color:=clred;
if combobox1.text='Vert' then
form1.color:=clgreen;

```

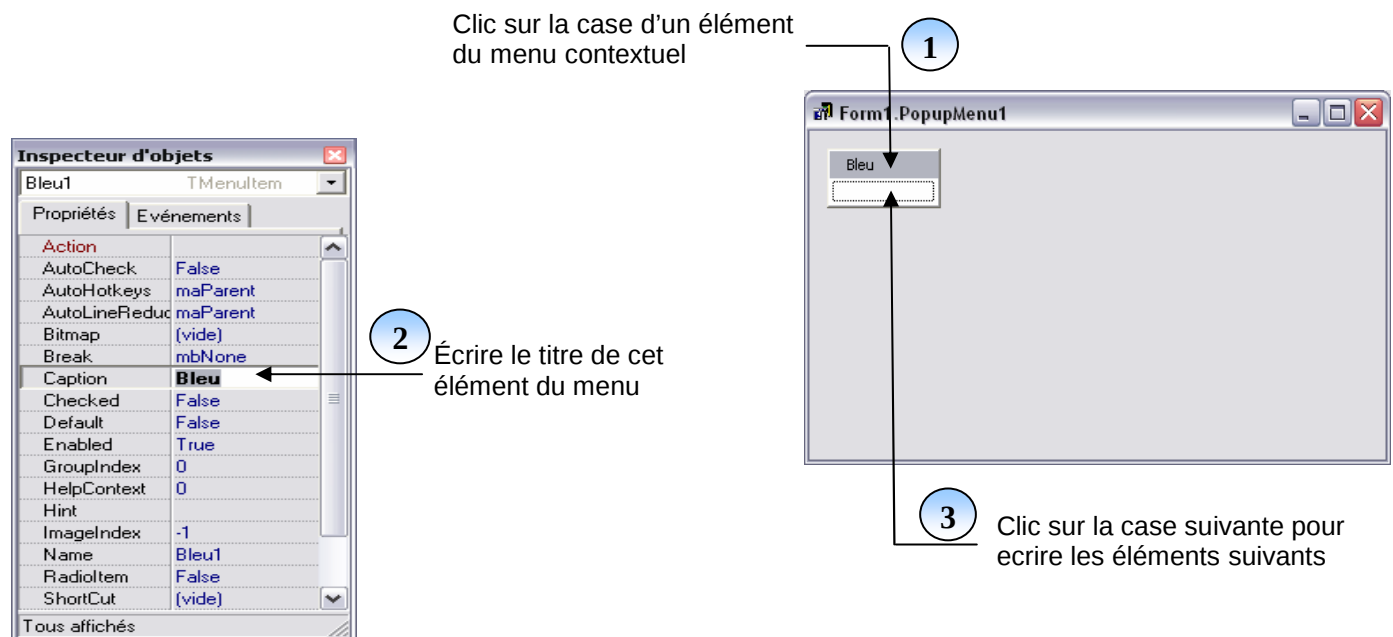
5- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et verifier le bon fonctionnement de la fiche.

Partie 3 :

1- insérer un composant button1 (palette standard) et écrire dans sa propriété caption : couleur.

2- insérer un composant popupmenu1  (palette standard), c'est un menu contextuel qui s'affiche lors du clique par le bouton droit de la souris sur un composant lié à ce menu.

3- double clique sur le popupmenu1, une fenêtre d'édition du menu contextuel s'affiche :



De cette façon, on écrit les éléments suivant :



A la fin fermer la fenêtre de l'editeur de menu.

4- double clique sur le Popupmenu1 et double clique sur l'élément « Bleu » et écrire dans la procédure :

```
Form1.color:=clblue ;
```

Et taper **F12** pour revenir à la fiche.

5- double clique sur le popupmenu1 et double clique sur l'élément « Rouge » et écrire dans la procédure :

```
Form1.color:=clred ;
```

Et taper **F12** pour revenir à la fiche.

6- double clique sur le popupmenu1 et double clique sur l'élément « Vert » et écrire dans la procédure :

```
Form1.color:=clgreen ;
```

Et taper **F12** pour revenir à la fiche.

7-dans la fiche clique sur le bouton1 (couleur) et dans son inspecteur d'objet choisir la propriété :



Clic ici pour choisir le menu : **PopupMenu1**

Cette propriété assure que si on clique par le bouton droit de la souris sur le Bouton1 le popupmenu1 s'affiche.

8- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et verifier le bon fonctionnement de la fiche.

Partie 4 :

- 1-insérer un composant Colordialog1 (palette Dialogues).
- 2-insérer un composant button2 (palette standard) et sa propriété caption ⇨ choix de couleur.
- 3-Double clique sur le composant bouton2 et écrire :

```
Colordialog1.execute;
Form1.color:=colordialog1.color;
```

- 4- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et vérifier le bon fonctionnement de la fiche.

Remarque :

Delphi propose des boîtes de dialogues divers pour contrôler et enrichir d'interface des applications par des API et des DLL Windows ; ces boîtes figurent dans la palette des composants dans l'onglet dialogues; par exemple :

Fontdialog : choix du font des composants de l'interface (la police du traitement de texte).

Colordialog : choix de la couleur des composants de l'interface.

Opendialog : donne la boîte de dialogue Windows d'ouverture des fichiers

Savedialog : donne la boîte de dialogue Windows d'enregistrement des fichiers.

Finddialog : recherche une chaîne de caractères dans un texte.

Replacedialog : recherche et remplacement d'une chaîne de caractère par une autre dans un texte.

Printdialog : permet la configuration de l'imprimante en cas d'impression des états.

Nb : pour afficher une boîte de dialogue quelconque on écrit : nomboite.execute ;

Exemple : fontdialog1.execute ;

Chaque boîte de dialogue a une propriété spéciale qui donne la possibilité de manipuler le résultat d'affichage de cette boîte ; exemple : la couleur choisie dans la boîte de dialogue Colordialog sera mise dans la propriété color de ce composant et on peut l'utiliser sur une fiche comme suite :

```
Form1.Color :=Colordialog1.Color ;
```

TP2 : LANCEMENT D'UN LOGICIEL (EXÉCUTABLE) À PARTIR D'UNE FICHE DELPHI

Pour lancer une application à partir de Delphi (exemple : une calculatrice) on procède comme suite:

- 1- on insère un button1 dans une fiche delphi.
- 2- double clique sur ce bouton et écrire dans la procédure :

```
WinExec('calc.exe',sw_show) ;
```

Remarque1:

Sw : show window (une fonction d'affichage des fenêtres).

Show : afficher.

Winexec : windows exécution

Remarque 2 :

Si Delphi ne reconnait pas le logiciel (Delphi ne reconnait que les logiciels installés par défaut avec l'installation de Windows) on doit écrire le chemin complet de ce logiciel.

Exemple : lancement du Microsoft word :

```
WinExec('C:\Program Files\Microsoft Office\Office\Winword.exe',sw_show) ;
```

TP3 : LES COMMENTAIRES

Les commentaires se placent entre accolades (pour un commentaire en plusieurs lignes), ou parenthèses et étoiles (pour commenter en plusieurs lignes) ou derrière 2 slashes (pour commenter en une seule ligne).

Exemple :

```
{Ceci est un commentaire}
```

```
(*Ceci est un autre commentaire*)
```

```
// Tout ce qui suit et jusqu'à la fin de ligne est aussi un commentaire
```

TP4 : LA MANIPULATION DE L'INTERFACE PAR CLAVIER

Interface



Étapes :

1-insérer les composants suivants de la palette Standard comme suite :

Composant	Propriété	Valeur
Edit1	text	
Edit2	text	
Edit3	text	
Label1	Caption	Devoir
Label2	Caption	Composition
Label3	Caption	Moyenne

2- clique sur le composant « Edit1 » et choisir son événement



Double clique ici

Et écrire dans sa procédure :

```
If key = vk_return then
Begin
If (strtofloat(edit1.text)<0)or(strtofloat(edit1.text)>20) then
Showmessage('note de devoir incorrecte')
else
Edit2.setfocus;
End;
```

3- tapez **F12** et clique sur le composant « Edit2 » et choisir son événement



Double clique ici

Et écrire dans sa procédure :

```
If key = vk_return then
Begin
If (strtofloat(edit2.text)<0)or(strtofloat(edit2.text)>40) then
Showmessage('note de composition incorrecte')
Else
Begin //Pour calculer la moyenne en ne gardant que deux chiffres après la virgule on écrit :
Edit3.text:= floattostr (round((strtofloat (edit1.text) + strtofloat (edit2.text))/3*100)/100 );
Edit3.setfocus;
End;
End;
```

4- tapez **F12** et clique sur le composant « Edit3 » et choisir son événement

```
If key = vk_f10 then
Begin
Edit1.clear;
Edit2.clear;
Edit3.clear;
Edit1.setfocus;
End;
```

5- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et verifier le bon fonctionnement de la fiche.

Remarque1 :

Key = touche (c'est l'une des touches du clavier).

vk = value key (valeur de la touche tapée).

Return = retour à la ligne (donc : vk_return = la touche entrée)

Setfocus = donner la main pour travailler par clavier (donc : mettre le curseur du clavier sur un composant)

Edit1.setfocus = mettre le curseur du clavier sur le composant edit1.

Remarque2 :

L'événement préféré pour programmer les touches (frappes) du clavier est : onkeydown

LEÇON 4 : MESSAGES, MULTIMEDIA ...

TP1 : LES MESSAGES DE DIALOGUE

Delphi propose 2 types de messages de dialogue :

1-les messages simples :

C'est un message d'information

Exemple



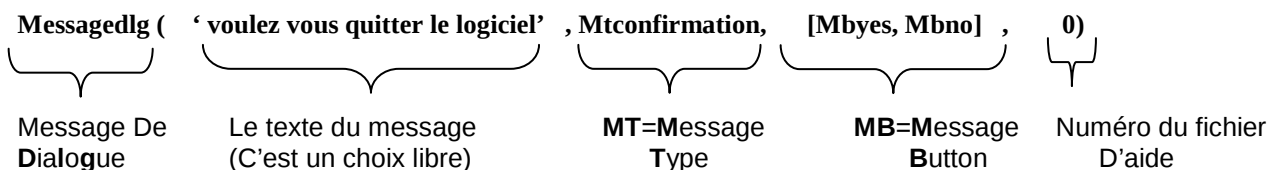
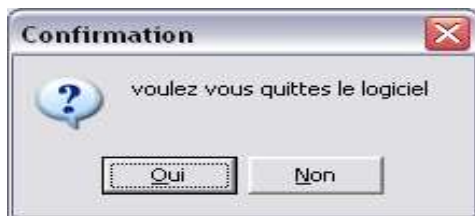
- L'instruction qui appelle ce message est :

Showmessage('bonjour tous le monde') ;

- on peut insérer un bouton de la palette standard et double clique sur ce dernier et écrire l'instruction au-dessus.

2-les messages avec options :

Exemple :



Les types de messages possibles :

Mtconfirmation = Confirmation
 Mtinformation = Information
 Mterror = Erreur
 Mtwarning = Avertissement

Les boutons des messages possibles :

mbYes = Un bouton avec le texte "Oui".
 mbNo = Un bouton avec le texte "Non".
 mbOK = Un bouton avec le texte "OK".
 mbCancel = Un bouton avec le texte "Annuler".
 mbAbort = Un bouton avec le texte "Abandonner".
 mbRetry = Un bouton avec le texte "Réessayer".
 mbIgnore = Un bouton avec le texte "Ignorer".
 mbAll = Un bouton avec le texte "Tous".
 mbNoToAll = Un bouton avec le texte "Non à tout".
 mbYesToAll = Un bouton avec le texte "Oui à tout".
 mbHelp = Un bouton avec le texte "Aide".

Le Numéro du fichier d'aide

0 = pas de fichier d'aide

Exemple :

```
If MessageDlg ('voulez vous quitter le logiciel ', mtconfirmation, [mbytes, mbno, mbcancel] ,0) =mryes then
Application.terminate
Else
Showmessage('vous pouvez continuer ') ;
```

TP 2: LES FICHIERS MULTIMÉDIA

Interface :



Étapes :

- 1- insérer le composant : mediaplayer1 (palette système).
- 2- modifier sa propriété :

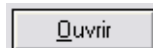


- 3-dans sa propriété :



Et choisir le fichier vidéo : C:\program files\borland\delphi5\demos\coolstuff\speedis.avi.

Et clique sur



- 4-lancer l'exécution et clique sur le bouton



Remarque

On peut lancer automatiquement le mediaplayer par un double clic sur un bouton et écrire dans sa procédure :

MediaPlayer1.play ;

Dans ce cas on doit cacher le mediaplayer par la modification de sa propriété :

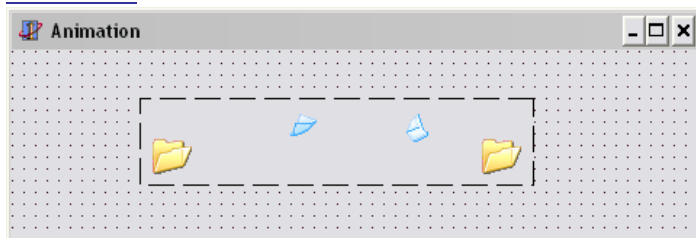


Remarque :

Les fichiers son de Windows sont dans le chemin : C:\windows\media

TP3 : LES ANIMATIONS

Interface :



Étapes :

- 1- insérer un composant Animate1 (palette win32)
- 2- choisir le type d'animation à utiliser dans sa propriété :



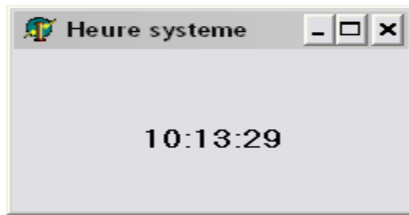
- 3- pour voir l'animation on modifie la propriété



- 4- répéter les étapes 2 et 3 pour voir les autres animations.

TP 4 : LES BOUCLES DE TEMPS

Interface :

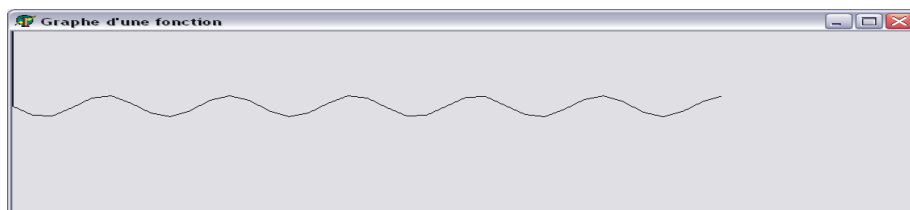


Étapes :

- 1- insérer un composant Timer1 (palette système)
- 2- insérer un composant Label1 (palette standard)
- 3- double clique sur le composant Timer1 et écrire dans la procédure :
Label1.caption :=timetostr(time)
- 4- Lancer l'exécution par la touche **F9**, ou par le bouton  ; Et verifier le bon fonctionnement de la fiche.

TP 5 : DESSINER UNE FONCTION

Interface :



Étapes :

- 1- insérer un composant Timer1 (palette système)
- 2- double clique sur ce composant et écrire dans la procédure :
Form1.canvas.lineto (i*15, trunc(sin(i)*15)+100);
i:=i+1 ;
- 3- en haut de l'unité on trouve la partie des variables globales comme suite :
 Var form1 :tform ;
i :integer ; //insérer la déclaration d'une variable globale
 Implementation

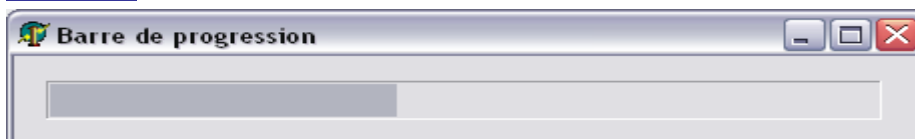
- 4- lancer l'exécution

Remarques:

- 1- le dessin sur écran se fait en pixel (**P**icture **E**lement) donc pour avoir un dessin visible il faut amplifier les coordonnées (x,y),en plus il faut décaler le dessin vers le milieu de la fiche par l'ajout de 100 pixels aux coordonnées (y) comme suite :
Form1.canvas.lineto (i*15, trunc(sin(i)*15)+100);
- 2- le composant Timer a une propriété intéressante : *Interval*
 (On peut donner dans cette propriété la fréquence de répétition du Timer)
- 3- le canvas est une feuille transparente sur chaque fiche Delphi qui aide à dessiner des graphes.

TP 6 : BARRE DE PROGRESSION

Interface :



Étapes :

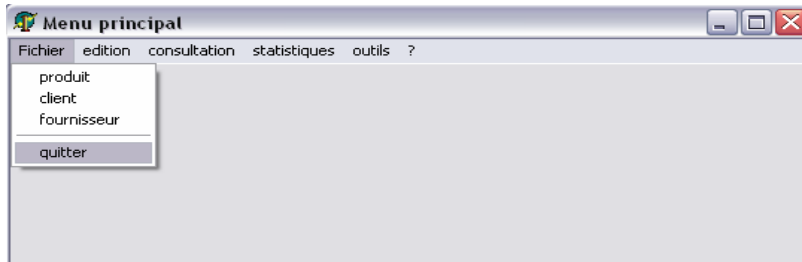
- 1- insérer un composant « Timer1 » de la palette « Système » et modifier sa propriété : interval $\Rightarrow$ 50.
- 2- insérer un composant « ProgressBar1 »  de la palette « Win32 » et modifier ses propriétés :

- 3- Double clique sur le timer1 et écrire dans sa procédure :
progressbar1.position:=progressbar1.position+1;
- 4- Lancer l'exécution pour voir la progression dans cette barre.

LEÇON 5 : MENU PRINCIPAL, BARRES D'OUTILS ...

TP 1 : LE MENU PRINCIPAL

Interface :

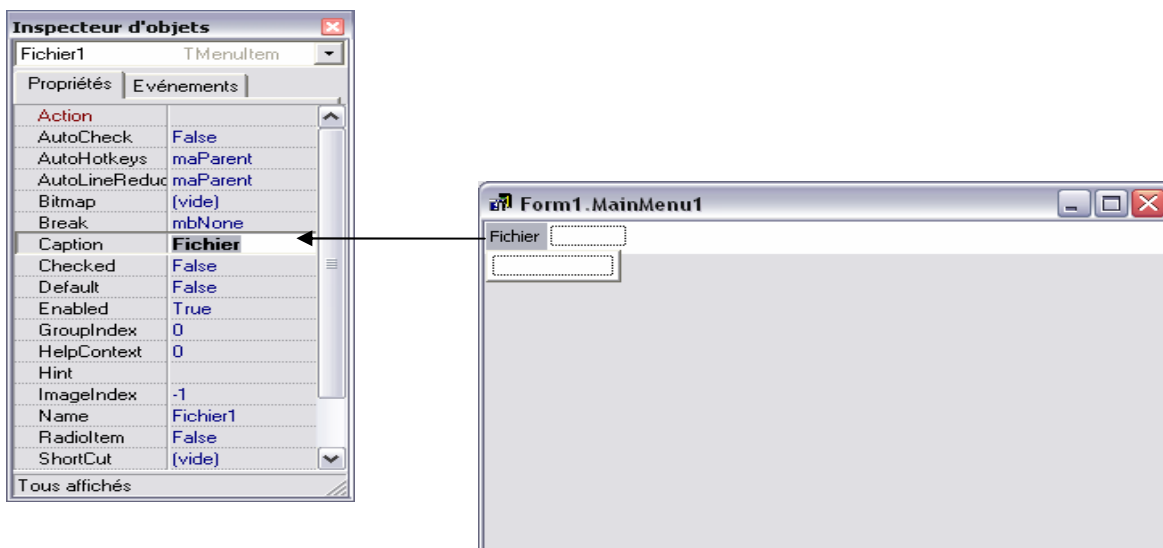


Étapes :

1- insérer de la palette standard le composant MainMenu1



2- double clique sur le composant MainMenu1 et écrire les éléments du menu dans leurs propriétés « caption »



3- pour faire une ligne de séparation on écrit dans la propriété "caption" le caractère "-"

4- pour chaque élément du menu on choisit une lettre et on écrit avant elle le symbole "&" pour assurer un raccourci clavier :

Exemple : &Edition ⇒ Edition

Donc : on tape le raccourci clavier Alt+E pour voir le menu Edition.

5- pour choisir un raccourci Clavier quelconque on sélectionne un élément du menu et dans la propriété



on choisit le raccourci adéquat (exemple : Ctrl + N pour l'élément « Nouveau » du menu).

6- pour insère un sous menu dans un élément du menu :

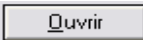
On clique sur cet élément et on tape les touches clavier : **CTRL** + **→** et on écrit les éléments du sous menu.

7- pour insère des icônes dans les éléments du menu ou insère le composant "imagelist1"  (palette "Win 32")

Et double clique sur ce composant "imagelist1"

Et clique sur le bouton 

Et choisir une image dans le chemin : « c:\program files\fichiers communs\borland shared\images\buttons »

Et clique sur 

Ensuite sur 

Et répéter **Ajouter** pour tous les autres images et enfin clique sur 

Ensuite clique sur le composant "MainMenu1" et choisit dans sa propriété :  **ImageList1**

Enfin : double clique sur " MainMenu1" et pour chaque élément du menu on choisit une image par sa propriété :



TP 2 : LES BARRES D'OUTILS

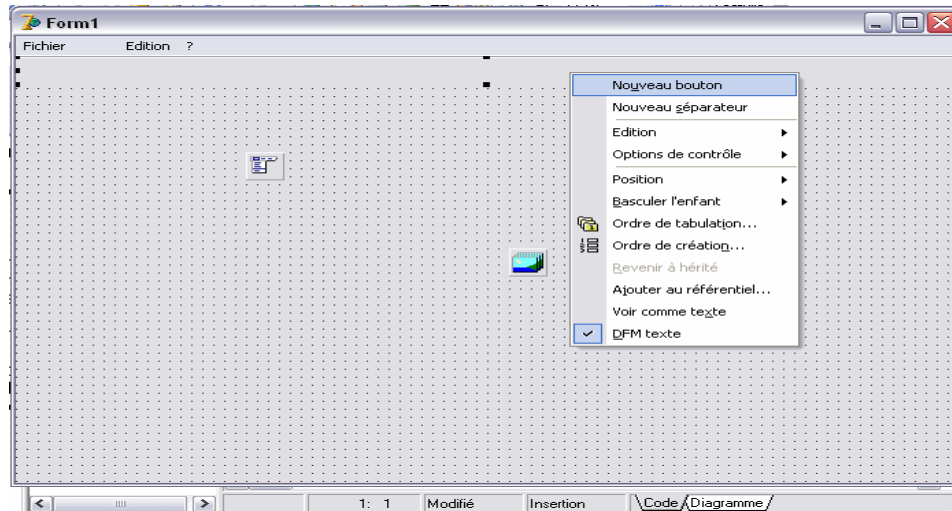
Interface :



La barre d'outils contient les éléments les plus utilisés du menu principal.

Étapes :

- 1- insérer de la palette "Win 32" le composant "Toolbar1"
- 2- clique droit sur ce composant et choisir : "Nouveau bouton"



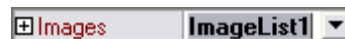
Pour chaque fois qu'on veut insérer un nouveau bouton, et on clic sur "Nouveau séparateur" séparer les groupes de boutons.

- 3- clique sur le composant "toolbar1" en évitant ses boutons et modifier sa propriété



Et cela pour que les boutons se gonflent lors du passage du pointeur de la souris au dessus et se dégonflent après.

- 4- Pour faire la liaison entre "imagelist1" et "toolbar1" on clique sur « toolbar1 »
- Et on choisit dans sa propriété



- 5- Pour modifier l'image de change bouton du Toolbar1 on clique sur ce bouton et on choisit son image par sa propriété :



- 6- pour expliquer le rôle de chaque bouton du Toolbar1 on clique sur ce bouton on écrit l'explication adéquate dans la propriété :



- 7- pour voir l'explication de ce bouton on choisit dans sa propriété :



LEÇON 6 : LA PROGRAMMATION DES BASES DE DONNEES

Définitions :

BD : Une base de données relationnelle est un ensemble de tables liées et gérées par un SGBD (Système de Gestion des Bases de Données).

Table : c'est une structuration de données sous forme de champs et d'enregistrements.

L'environnement de programmation Delphi donne la possibilité de se connecter à presque tous les SGBD disponibles sur le marché.

TP 1 : CRÉATION D'UNE ALIAS DE BASE DE DONNÉES

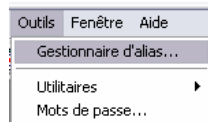
Delphi propose un module base de données pour simplifier la création des alias de BD.

- Pour accéder au module BD (nommé aussi BDE : **B**orland **D**atabase **E**ngine) on choisit dans le menu Delphi :

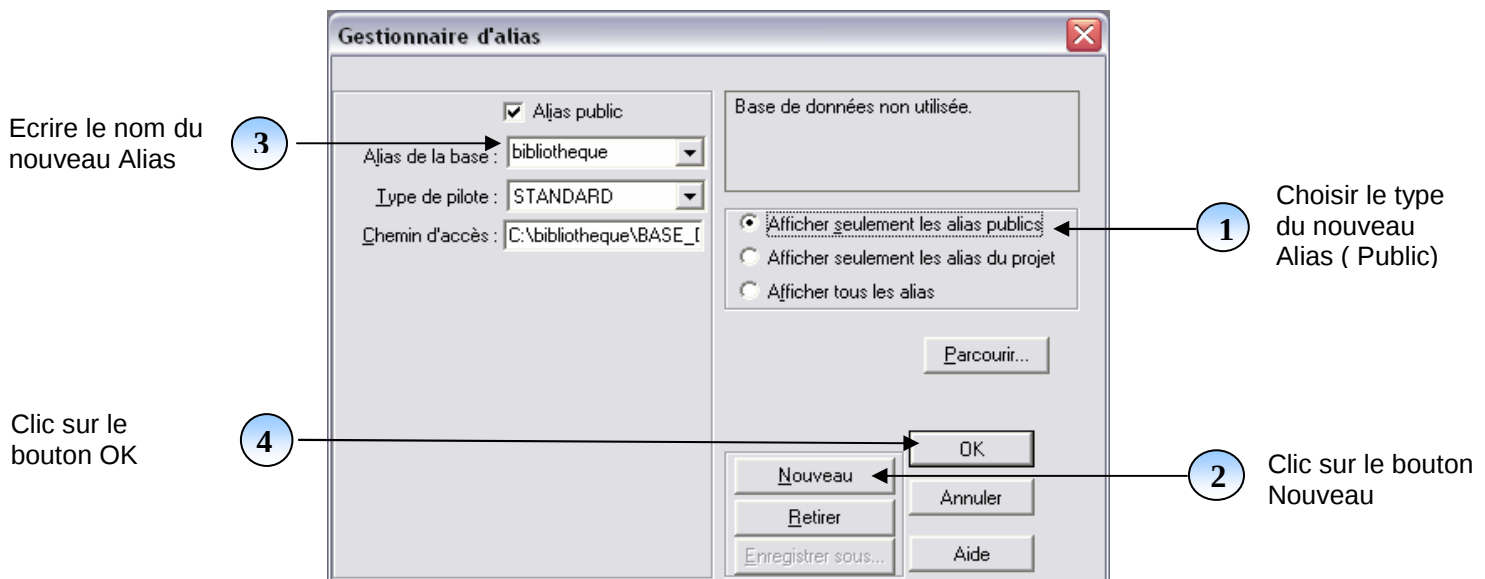
Outils ⇒ module base de données.

un alias est un chemin logique qui localise et donne la possibilité d'accès à la BD. Un alias représente une base de données ainsi que les informations de configuration de cette dernière.

- Pour créer un nouveau alias on procède comme suite :
 - Dans le menu du BDE on choisit : outils ⇒ gestionnaires d'alias.

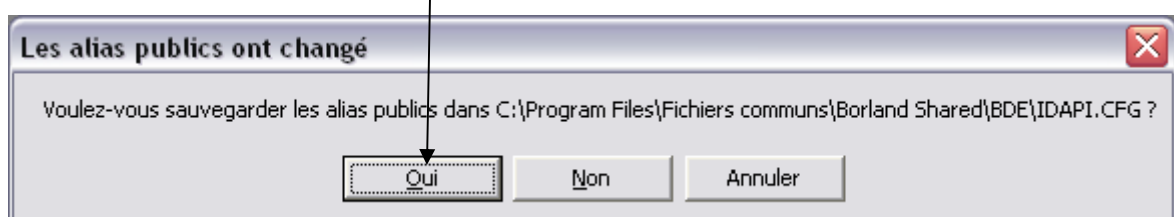


Ensuite dans la fenetre de gestionnaire des alias on procede comme suite :



Enfin, et dans la fenetre qui s'affiche :

Clique sur **Oui** pour enregistrer cet alias avec les alias publics.



TP 2 : CRÉATION D'UNE TABLE

1- Dans le menu BDE on choisit : fichier ⇒ nouveau ⇒ table.

Ensuite On choisit le SGBD par défaut « paradox 7 » et clique sur **OK**.



On écrit la structure de la table suivante :

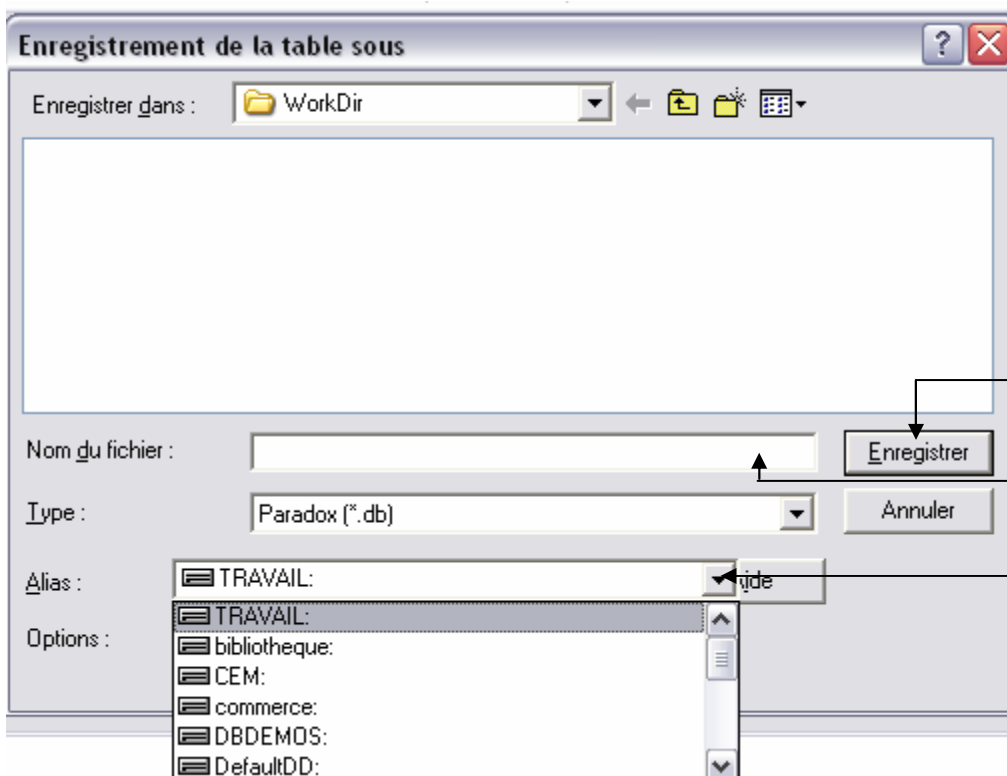


Double clic ici, ou tapez n'importe quelle touche clavier pour créer un index primaire

Pour enregistrer, on clique sur le bouton

1

Ensuite, et dans la fenêtre suivante on enregistre la table « produit » comme suite :



4

Clic sur enregistrer

3

Écrire le nom de la table (produit)

2

Choisir l'Alias

TP 3 : CONNEXION ET AFFICHAGE D'UNE BASE DE DONNÉES DANS UNE FICHE

- insérer de la palette BDE le composant « Table1 »  et manipuler ses propriétés :

Choisir notre alias DatabaseName **Test**

Choisir la table TableName **Produit.db**

Activer la connexion à la table Active **True**

- Pour afficher notre table on doit insérer de la palette Accesbd le composant « datasource1»  et choisir dans sa propriété :

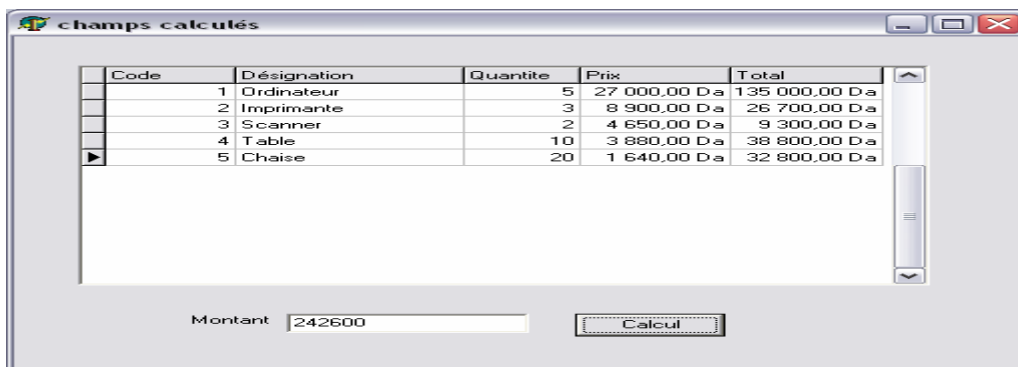
DataSet **Table1**

- Insérer de la palette Contrôlebd le composant « Dbgrid1 »  et choisir dans sa propriété :

DataSource **DataSource1**

TP 4 : LES CHAMPS CALCULÉES ET LES BOUCLES DANS UNE BD (TP FACTURE)

Interface :



Étapes :

- Dans le module base de données (BDE) :
 - On commence par la création d'un nouveau alias nommé "tp_facture" (Voir les étapes du TP1) :
 - Dans cet alias on créera la table suivante :

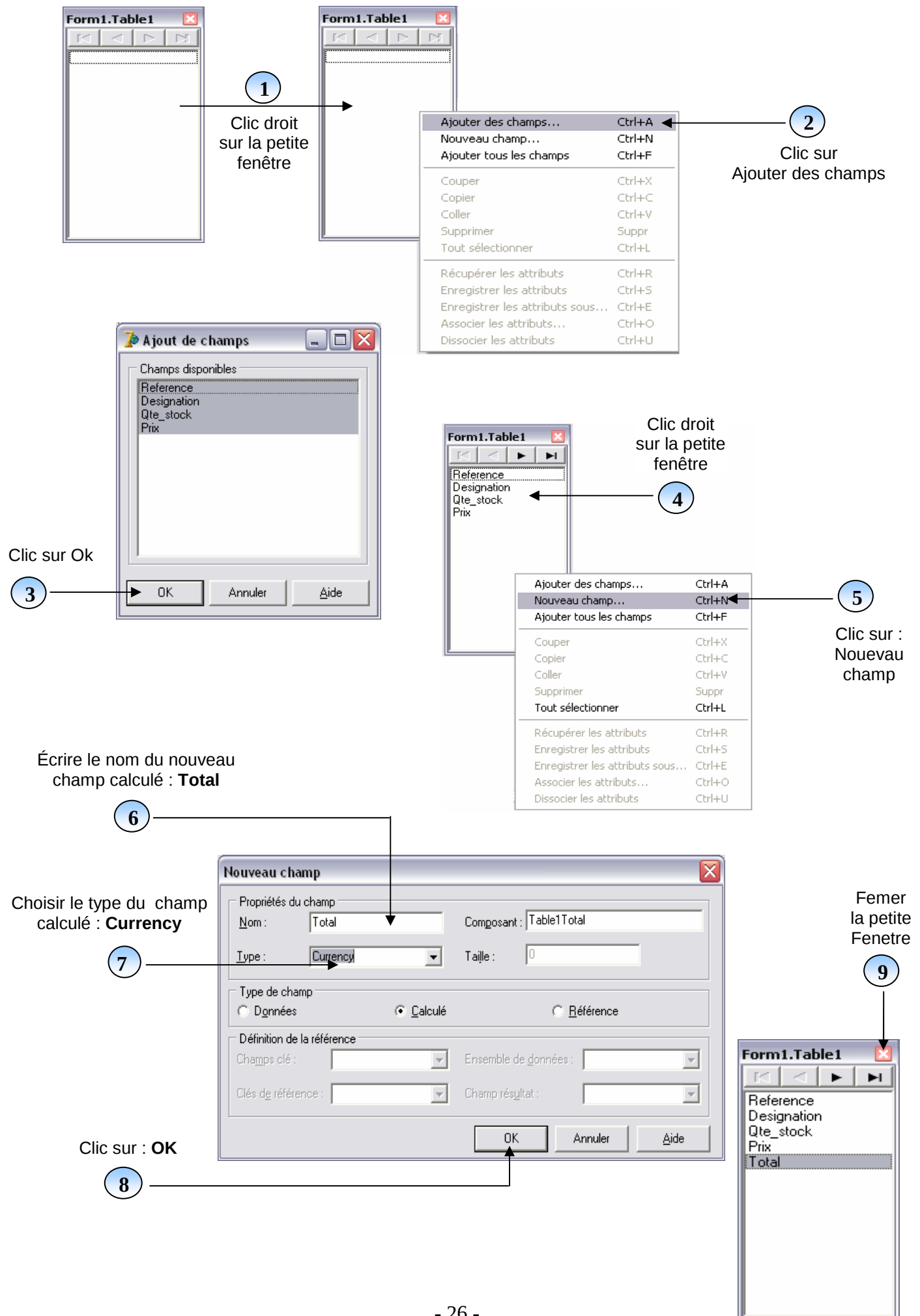
Champ	type	taille	index
Code	I		*
Désignation	A	20	
Quantite	N		
prix	\$		

- On enregistre cette table dans notre alias "tp_facture" sous le nom : « produit ».

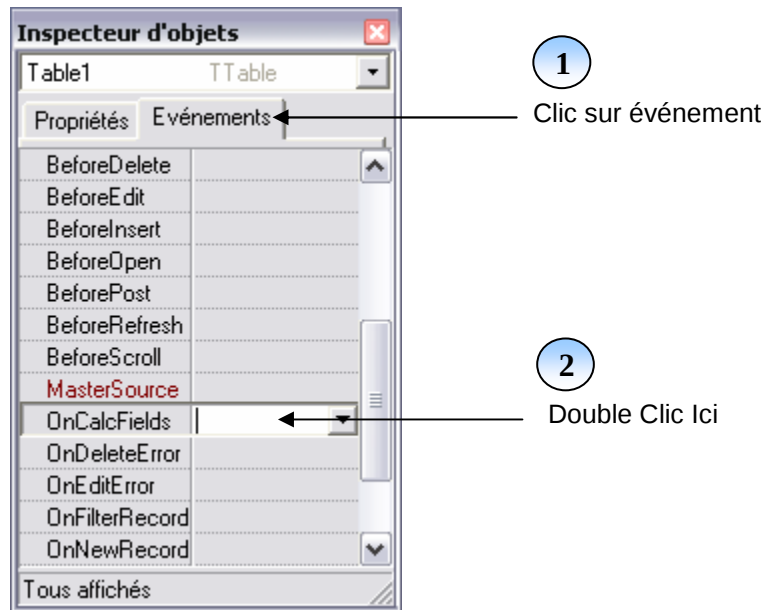
- Dans une fiche Delphi on insère les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Tp_facture
		Tablename	Produit
		Active	True
AccesBD	Datasource1	Dataset	Table1
ControleBD	Dbgrid1	Datasource	Datasource1
Standard	Label1	Caption	Montant
	Edit1	Text	
	Button1	Caption	Calcul

- Pour introduire le champ calculé « Total » dans la table produit on procède comme suite :
 - On double clique sur le composant table1, et dans la petite fenetre blanche qui s'affiche :



- 4- Dans l'inspecteur d'objet du table1 on lance la procédure de l'événement : **OnCalcField**



Et on écrit dans la procédure :

```
table1.fieldbyname('total').ascurrency := table1.fieldbyname('quantite').asfloat *
table1.fieldbyname('prix').ascurrency ;
```

- 5- Tapez **F12** et pour calculer le montant global de ces produits on double clique sur le Bouton1 et on écrit dans sa procédure : (Supprimer le **begin** et **end** de la procédure et coller tous le code source suivant)

```
var  somme : real;
begin
    somme := 0;
    table1.first; // se pointer vers le premier enregistrement de la table
    while not(table1.eof) do // tanque on a pas arriver à la fin de la table
    begin
        somme := somme + table1.fieldbyname('total').ascurrency; // cumul des totaux
        table1.next; // passer à l'enregistrement suivant.
    end;
    edit1.text := floattostr(somme); // afficher le cumul des totaux dans le comosant edit1
end ;
```

- 6- Lancez l'exécution.

Remarque :

La syntaxe d'écriture d'un champ d'une table est comme suite :

Source-du-champ.**fieldbyname**('nom-du-champ').**astype-du-champ**

Exemples :

table1.fieldbyname('total').ascurrency

table1.fieldbyname('nom').asstring

table2.fieldbyname('date_naissance').asdate

table4.fieldbyname('quantite').asfloat

TP 5 : LES CHAMPS CALCULÉES ET LES BOUCLES DANS UNE BD (TP BULLETIN)

Interface :

Numéro	Matiere	Devoir	Composition	moyenne_20	Coefficient	moy_coef
1	Mahes	15	33	16	5	80
2	Physique	12	19	10,33	4	41,33
3	Chemie	8	21	9,67	4	38,67
4	Sciences	19	39	19,33	3	58
5	Lettres	4	17	7	2	14

Moyenne générale

Resultat

Étapes :

- 1- Dans le module base de données (BDE) :
 - On commence par la création d'un nouveau alias nommé "scolarité" (voire Tp1)
 - Dans cet alias on créera la table "bulletin" suivante :

Champ	type	taille	index
Numéro	i		*
Matière	A	20	
Devoir	N		
Composition	N		
Coefficient	i		

- On enregistre cette table dans notre alias "scolarité" sous le nom : « bulletin ».

- 2- Dans une fiche Delphi insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	scolarité
		Tablename	bulletin
		Active	True
AccesBD	Datasource1	Dataset	Table1
ControleBD	Dbgrid1	Datasource	Datasource1
Standard	Label1	Caption	Moyenne générale
	Edit1	Text	
	Button1	Caption	Calcul
	Label2	Caption	Résultat
	Edit2	Text	

- 3- Pour introduire les champs calculés « Moyenne_20 » et « Moy_coef » dans la table bulletin on procède comme suite :
 - On double clique sur le composant table1
 - Ensuite : on clique droit sur la petite fenêtre et on clique sur « ajouter tous les champs ».
 - Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».
 - Ensuite : on écrit le nom du champ⇒Moyenne_20, on choisit le type du champ⇒float et on clique sur .
 - Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».
 - Ensuite : on écrit le nom du champ⇒Moy_coef, et on choisit le type du champ⇒float et on clique sur .
 - Ensuite : on ferme la petite fenêtre on clique sur le composant table1 et dans son inspecteur d'objet on clique sur l'onglet « événements ».
 - Ensuite : on double clique à droite de l'événement « oncalcfields » et on écrit dans la procédure :

```
table1.fieldbyname('moyenne_20').asfloat:= (table1.fieldbyname('devoir').asfloat+
table1.fieldbyname('composition').asfloat)/3;
table1.fieldbyname('moy_coef').asfloat:= table1.fieldbyname('moyenne_20').asfloat*
table1.fieldbyname('coefficient').asinteger;
```

- 4- Pour afficher la moyenne générale et le résultat de ce bulletin on double clique sur le bouton "calcul" et on écrit dans la procédure : (Supprimer le **begin** et **end** de la procédure et coller tous le code source suivant)

```

var  som_moy , mg : real;
     som_coef : integer;
begin
  som_moy := 0;
  som_coef := 0;
  table1.first;
  while not(table1.eof) do
  begin
    som_moy := som_moy + table1.fieldbyname('moy_coef').asfloat;
    som_coef := som_coef + table1.fieldbyname('coefficient').asinteger;
    table1.next
  end;
  mg := som_moy / som_coef;
  Mg := round(mg*100)/100 ;
  edit1.text := floattostr(mg);
  if mg <10 then
    edit2.text := 'étudiant ajourné'
  else
    edit2.text := 'étudiant admis';
end;

```

Remarque :

❖ Pour contrôler le format d'affichage des nombres réels (champs de type float) dans une BD (limiter le nombre de chiffres après la virgule) on procède comme suite :

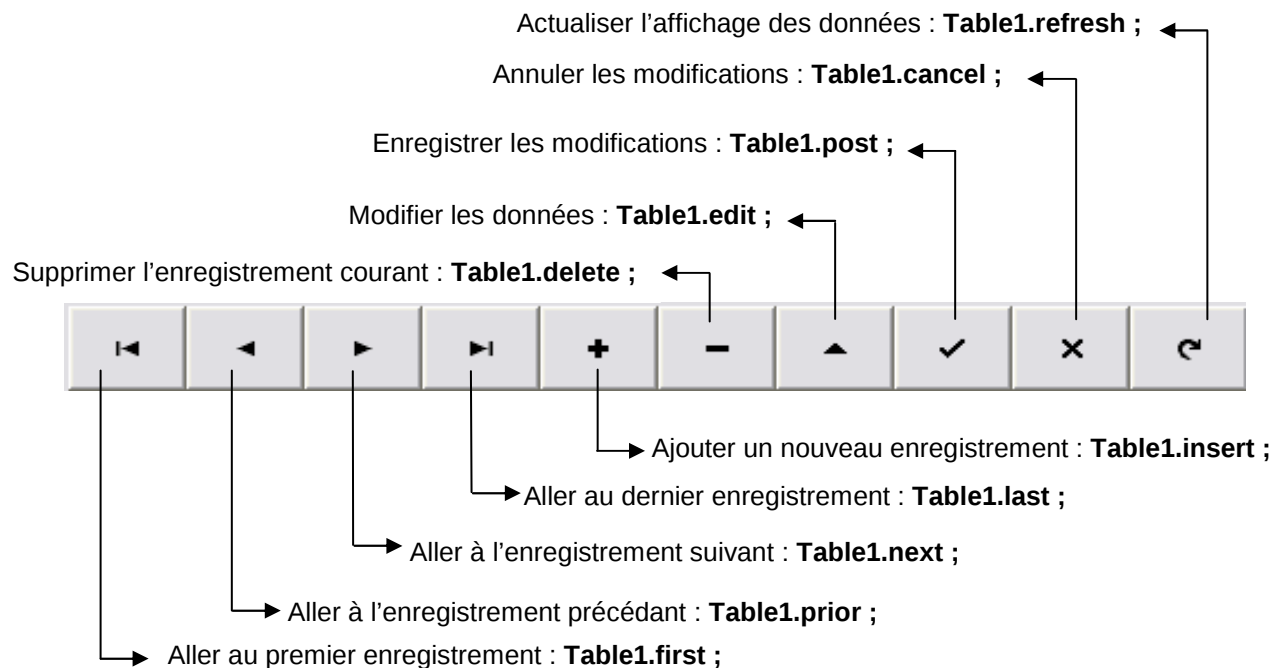
- double clique sur le composant « table1 ».
- clique sur le champ à formater (pour nous on va manipuler les deux champs «moyenne_20» et «moy_coef»)
- Écrire dans sa propriété :



(# Représente un chiffre, donc le format ##.## représente 2 chiffres avant et après la virgule)

TP6 : LE DBNAVIGATOR

Un DBNavigator Contient tous les opérations de base sur une source de données (table ou requête)



Étapes :

- On insère le composant DBNavigator1 de la palette contrôleBD.
- Pour le connecter à une source de données on choisit dans sa propriété :



Pour afficher l'explication (sous forme d'info-bulls) de chaque bouton du DBNavigator1 on choisit dans sa propriété :



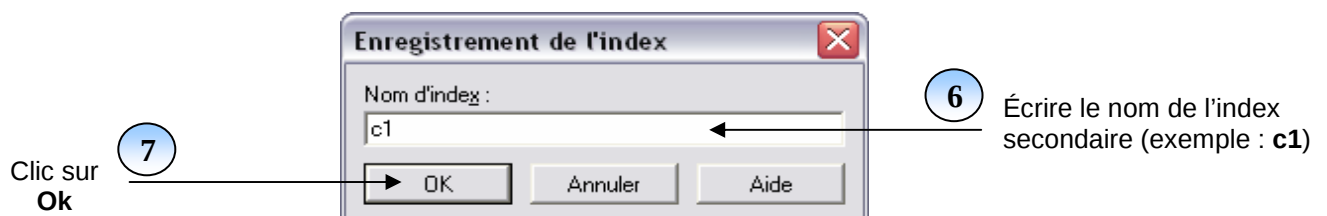
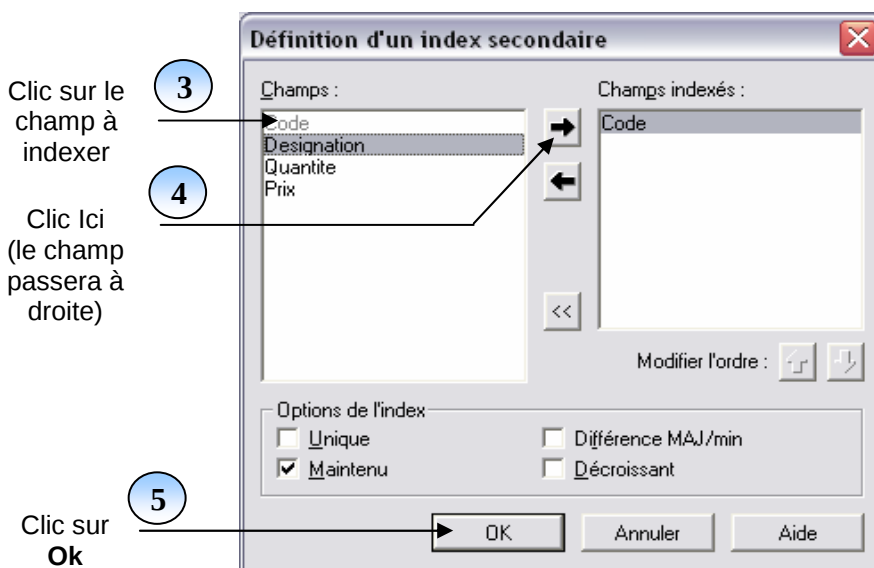
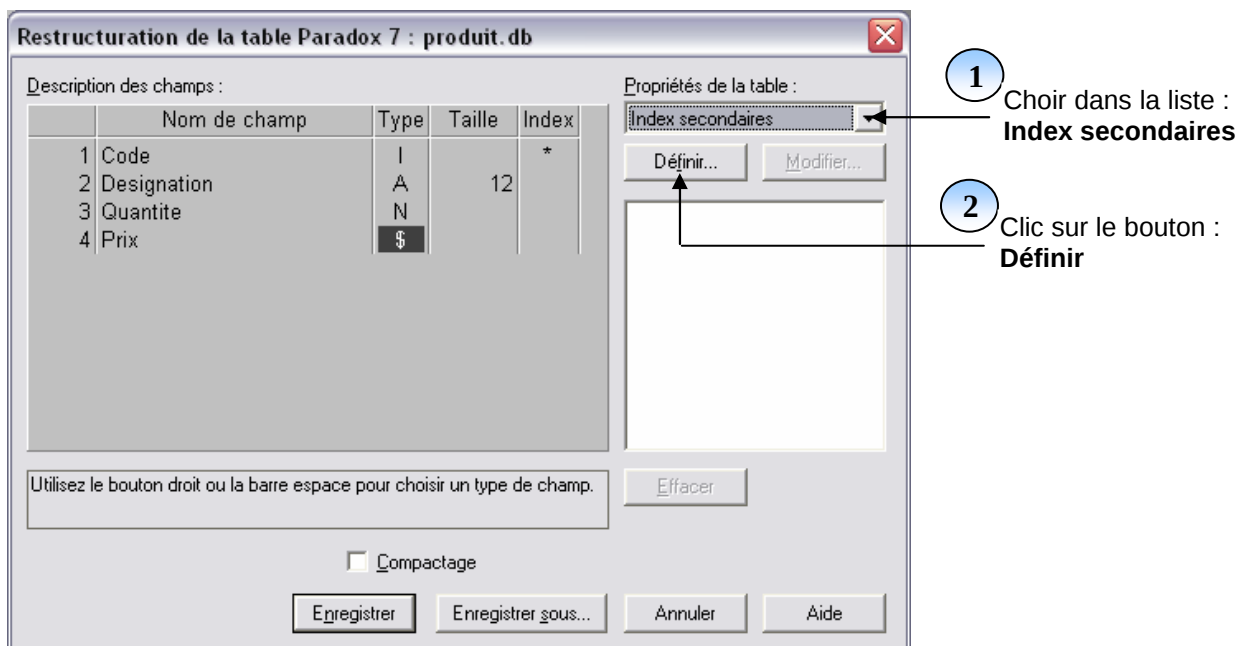
LEÇON 7 : LES INDEX SECONDAIRES ET LEURS UTILISATIONS

TP1 : LA CRÉATION DES INDEX SECONDAIRES

1- dans le Module base de données créez la table suivante :

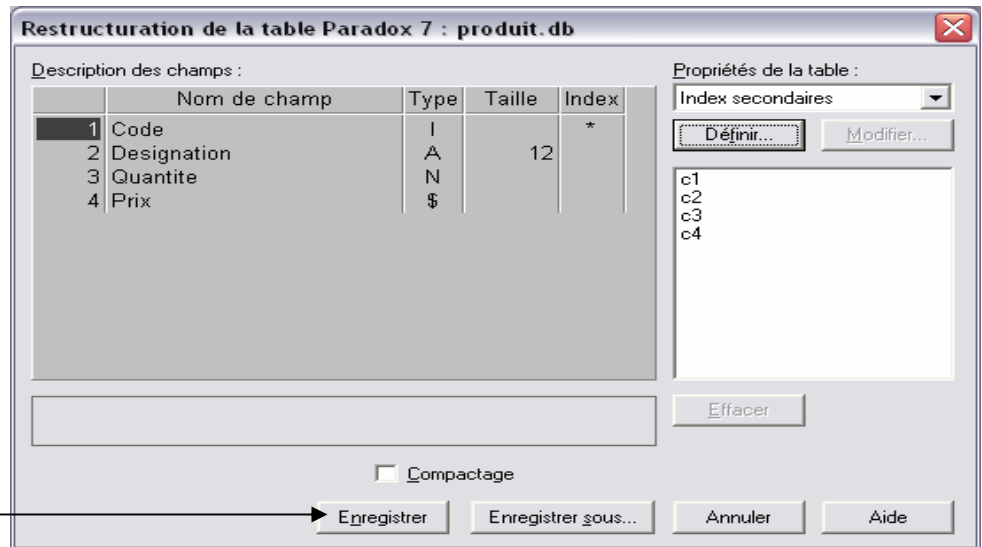
Champ	type	taille	index
code	I		*
designation	A	20	
Quantite	N		
prix	\$		

1- Avant d'enregistrer cette table et Pour insérer des index secondaires pour chaque champ on procède comme suite :



- Répéter les opérations : 2, 3, 4, 5, 6, 7 pour les autres champs de la table
On peut proposer par exemple les noms d'index secondaires comme suite :
Designation : c2
Quantite : c3
Prix : c4

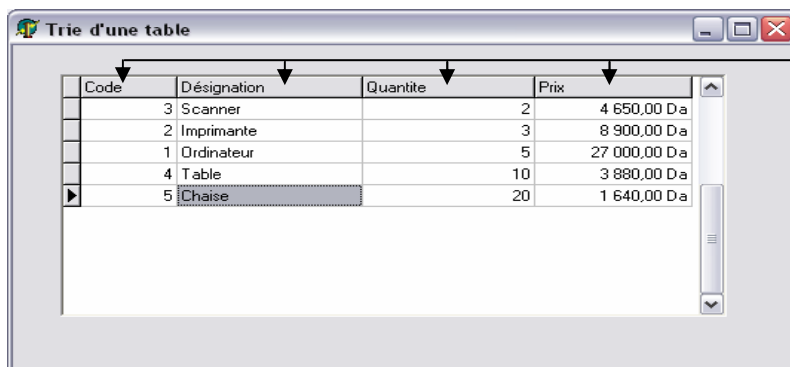
2- A la fin on aura l'affichage suivant :



Clic sur enregistrer, et choisir
da la fenetre d'enregistrement :

Alias : **DbDemos**
Table : **Produit**

TP2 : LE TRIE D'UNE TABLE



Clic ici pour trier la table par l'un des champs.
(Dans cet affichage la table est triée par le
champ quantité)

1- dans une fiche Delphi insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	dbdemos
		Tablename	produit
		Active	True
AccesBD	Datasource1	Dataset	Table1
ControleBD	Dbgrid1	Datasource	Datasource1

2- pour faire le trie de la table produit :

Clique sur le composant dbgrid1.

Ensuite : dans son inspecteur d'objet on clique sur l'onglet « événements »

Ensuite : on double clique à droite de son événement « onclick » et on introduit le code source suivant :

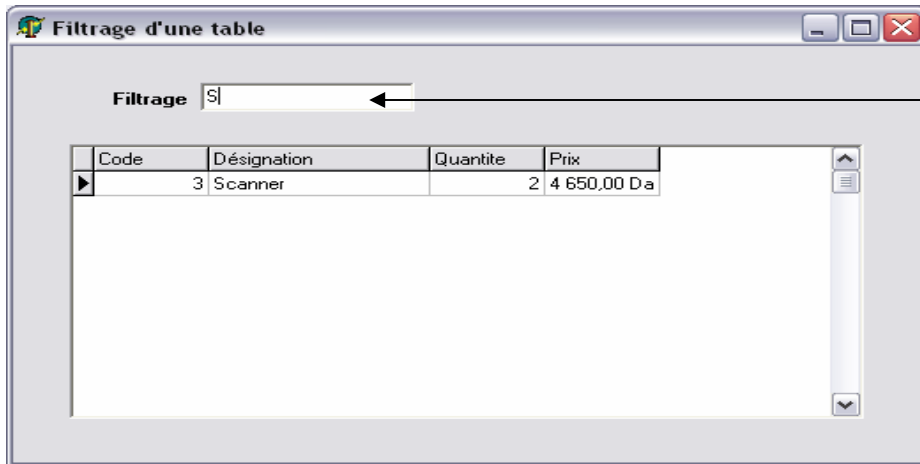
```

If column.index = 0 then
Table1.indexname := 'c1' ;
If column.index = 1 then
Table1.indexname := 'c2' ;
If column.index = 2 then
Table1.indexname := 'c3' ;
If column.index = 3 then
Table1.indexname := 'c4' ;

```

2- Lancer l'exécution par **F9** ; si on clique sur un entête de colonne (un champ) dans le DbGrid1, cette table sera triée automatiquement par ce champ.

TP3 : LE FILTRAGE D'UNE TABLE



Le filtrage est une technique très utile pour faire une recherche rapide dans les BD. À chaque fois qu'on écrit une (des) lettre (s), seulement les enregistrements contenant des désignations (noms) commençant par cette (ces) lettre (s). Si on supprime toutes les lettres, tous les enregistrements de la table s'afficheront.

Pour faire le filtrage :

- 1- Garder l'interface du Tp2 et insérer un composant Edit1 et effacer sa propriété « text »
- 2- Insérer un composant Label1 et écrire dans sa propriété : Caption ⇒ filtrage.
- 3- Pour faire le filtrage par le champ designation double clique sur ce Edit1 (génération de la procédure « onchange ») et écrire le code source suivant :

```
4-
if edit1.text <> '' then
begin
table1.close;
table1.filtered := true ;
table1.filter := '[designation]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;
```

Remarque :

Dans une base de données on peut voir des fichiers avec les extensions suivantes:

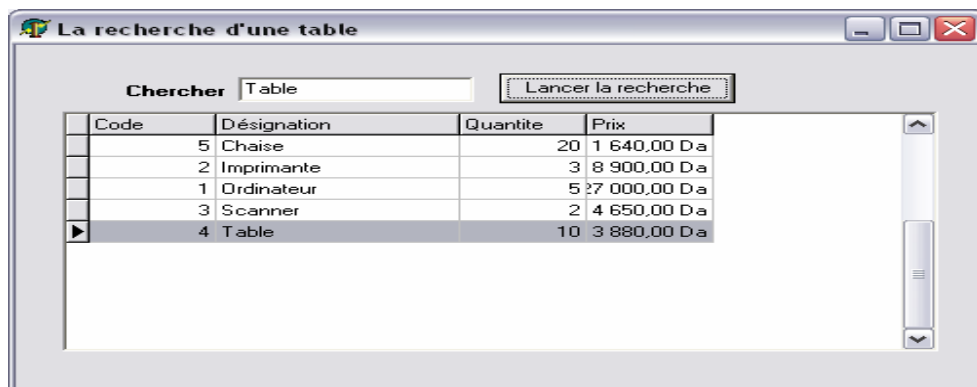
- *.db : c'est une table (sa structure (champs) et son contenu (enregistrements)).
- *.px : c'est un index primaire (clé d'une table)
- *.xg0,*.xg1,...,*.yg0,*.yg1,... : les index secondaires d'une table.

Rappel :

Dans une application delphi (logiciel) on peut voir des fichiers avec les extensions suivantes :

- *.dpr : **D**elphi **p**roject : c'est l'unité principale qui représente le projet (le logiciel).
- *.pas : **P**ascal : c'est une unité.
- *.dfm : **D**elphi **F**orm : c'est une fiche.
- *.dcu : **D**elphi **c**ompiled **u**nit : c'est une unité compilée
- *.exe : **E**xecutable : c'est le fichier exécutable du logiciel.

TP4 : LA RECHERCHE SIMPLE DANS UNE TABLE



- Pour faire la recherche simple:

- 1- garder l'interface du Tp3 et insérer un composant Edit2 et effacer le contenu de sa propriété « text »
- 2- insérer un composant Label2 et écrire dans sa propriété : caption ⇒ chercher.
- 3- insérer un composant Button1 et écrire dans sa propriété : caption ⇒ lancer la recherche.
- 4- double clique sur ce Button1 et écrire dans sa procédure le code source suivant :

Table1.indexname:='c2'; // l'index du champ objet de la recherche (designation).

Table1.setkey ; // préparation de la recherche dans la table produit.

Table1.fieldbyname('designation').asString :=edit2.text ; // la valeur à rechercher dans ce champ.

Table1.gotokey ; // lancer la recherche.

Remarque1 :

Delphi propose une autre instruction plus simple pour faire la recherche sur une base de données comme suite :

Table1.indexname := 'c2' ;

Table1.Findkey([Edit2.Text]); // Cette instruction remplace les trois instructions précédantes.

Remarque2 :

Pour avoir une ligne de sélection permanente dans le Dbgrid1 qui indique l'enregistrement résultat de la recherche :

- On clique à gauche de la propriété « Options » du dbgrid1 (exactement sur la case ☐).
- Choisir dans la sous propriété : DGRowSelect ⇒ true.
- Choisir dans la sous propriété : DGAlwaysShowSelection ⇒ true.

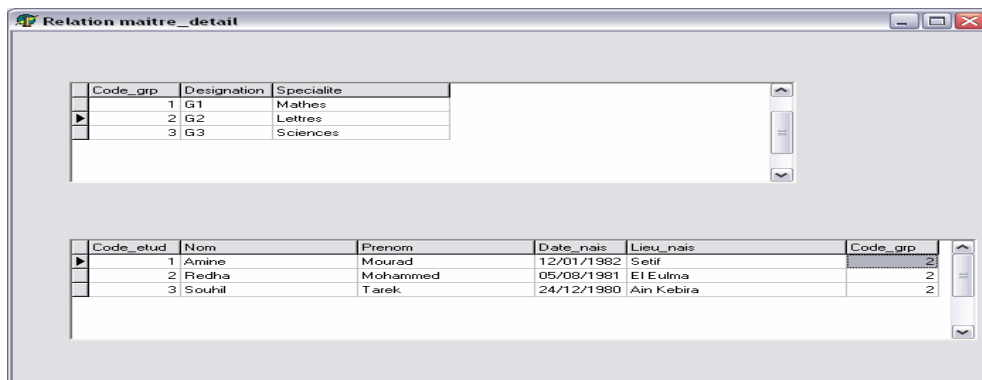
Quelques explications:

Eof : end of file : c'est la fin d'une table.

Bof : Begin of file : c'est le début d'une table.

#39 : c'est le code ASCII de l'apostrophe ' .

TP5 : LES RELATIONS MAITRE-DETAIL



1- Dans le BDE créer la table suivante :

Champ	type	taille	index
Code_grp	i		*
designation	A	10	
spécialité	A	20	

On enregistre cette table dans notre alias "dbdemos" sous le nom : « groupe ».

2- Dans le BDE (module base de données) créer la table suivante :

Champ	type	taille	index
Code_etud	I		*
Nom	A	20	
prénom	A	20	
Date_nais	D		
Lieu_nais	A	25	
Code_grp	i		

Avant d'enregistrer cette table insérer un index secondaire pour le champ code_grp (clé étrangère) comme suite :

- On choisit en haut et à droite de la fenêtre dans la liste des choix : propriétés de la table ⇒ index secondaires.
- On clique sur le bouton **Définir**.
- On clique sur le champ « code_grp » dans la liste à gauche.
- On fait passer ce champ à droite par le bouton ⇨.
- On clique sur le bouton **OK** et on propose un nom à l'index secondaire de ce champ, exemple : « c6 »
- On enregistre cette table dans notre alias "dbdemos" sous le nom : « étudiant ».

3- Dans une fiche Delphi insérer les composants suivants :

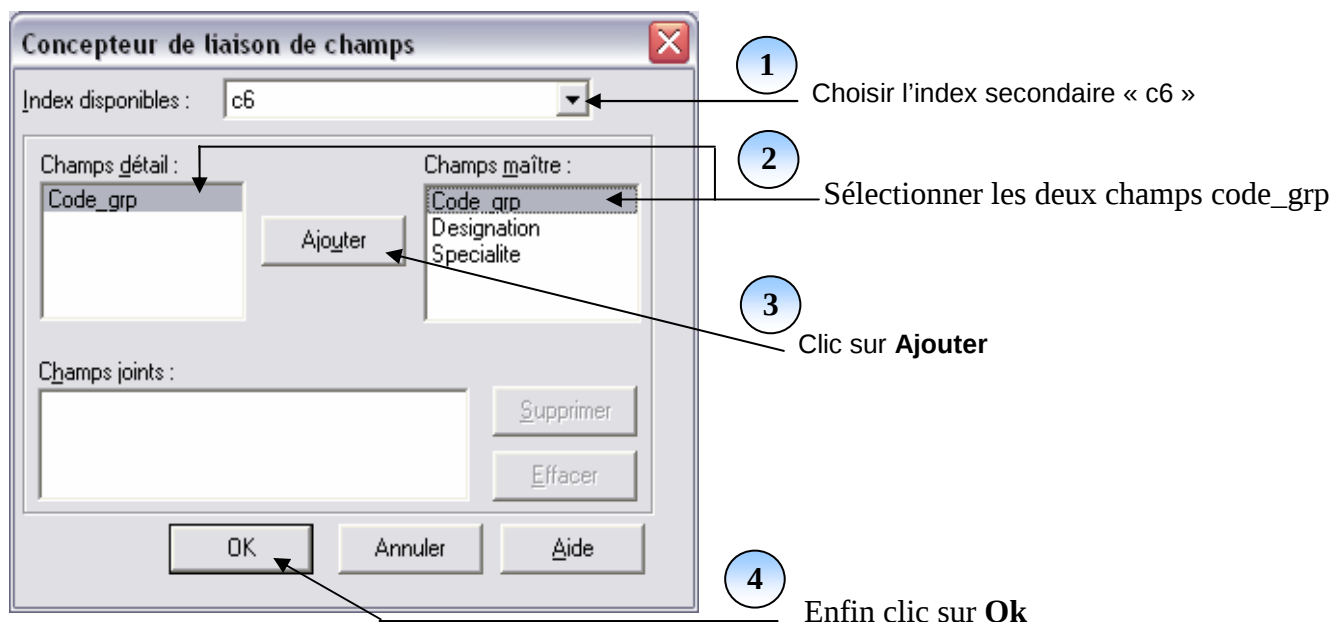
Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	dbdemos
		Tablename	groupe
		Active	True
AccesBD	Datasource1	Dataset	Table1
ContrôleBD	Dbgrid1	Datasource	Datasource1
BDE	Table2	Databasename	dbdemos
		Tablename	etudiant
		Active	True
AccesBD	Datasource2	Dataset	Table2
ContrôleBD	Dbgrid2	Datasource	Datasource2

3- Pour que la table etudiant soit liée par une relation maître-détail (père-fils) avec la table groupe

(La table groupe ⇒ maître (père), la table etudiant ⇒ détail (fils)) on procède comme suite :

On clique sur le composant table2 « etudiant», et on choisit dans sa propriété : mastersource ⇒ datasource1.

Ensuite : dans sa propriété « masterfield » on clique sur le petit bouton [...] et on suit les étapes suivantes :



Le résultat sera comme suite : si on choisit un groupe dans le dbgrid1 on aura seulement les étudiants de ce groupe qui s'affichent dans le dbgrid2.

LEÇON 8 : INTRODUCTION AUX REQUETES SQL

PRÉSENTATION GÉNÉRALE

1-Définition :

Le langage SQL (Structured Query Language) est un langage universel de création, de gestion et d'interrogation des bases de données. SQL est intégré dans plusieurs langages de programmation.

2- Les sous langages du SQL

Le SQL se subdivise en différents langages :

2.1- Langage de description des données (DDL : Data Definition Language) :

Permettant la définition des éléments d'une base de données, Celui-ci comprend trois commandes qui permettent l'administration d'une base de données, ces commandes de base sont :

CREATE	Création de tables ou d'index
ALTER	Modification de tables ou d'index
DROP	Suppression de tables ou d'index

2.2- Langage de manipulation des données (DML : Data Manipulation Language) :

Permettant la manipulation et l'extraction des données. Celui-ci comprend les commandes qui permettent les mises à jour des données d'une table, ces commandes de base sont :

INSERT	Insertion de nouvelles lignes dans une table
UPDATE	Mise à jour de certains champs d'une ligne dans une table
DELETE	Suppression des lignes inutiles dans une table

2.3- Langage de contrôle des données (DCL : Data Control Language) :

Permettant la gestion des droits d'accès aux données et des transactions (en cas de programmation sous réseau).

GRANT	Attribution de droits d'accès
REVOKE	Suppression de droits d'accès
COMMIT	Prise en compte des dernières transactions pour une mise à jour des données
ROLLBACK	Suppression des dernières transactions et restauration des anciennes données

2.4- Langage d'interrogation des données (DQL : Data Query Language).

Bien que ne comprenant qu'une seule commande SELECT, celle-ci étant la plus utilisée car si le peuplement d'une base de données est une étape importante, les différentes requêtes pour l'extraction des données représentent autant d'étapes importantes pour tous types d'utilisation :

SELECT	Extraction des données par projection, par sélection, avec restriction, par groupement
---------------	--

Exemple 1 :

```
CREATE DATABASE Commerce;
START DATABASE Commerce;
CREATE TABLE client (Numero SMALLINT, nom CHAR(30), prenom CHAR(30), adresse CHAR(50), telephone CHAR(15));
STOP DATABASE;
```

Ici nous venons de créer une base de données Commerce (ligne 1).

On ouvre ensuite cette base de données (ligne 2).

On y crée une table de nom Client, qui contient 5 colonnes.(ligne 3)

On ferme la base de donnée (ligne 4)

Exemple 2 :

```
INSERT INTO Client VALUES (1,'MOHAMED','Amine','Sétif','036-84-46-68');
```

Ici, on insère un nouveau enregistrement dans la table client ayant pour valeur de champs les données mentionnées.

```
UPDATE Client SET prenom='Redha', adresse='Alger' WHERE nom='MOHAMED';
```

Cette requête SQL modifie le prénom et l'adresse du client nommé MOHAMED par les valeurs mentionnées.

```
DELETE FROM Client WHERE numero=15;
```

Cette requête supprime l'enregistrement (le client) numero 15.

TP1 : LES REQUETES SQL LES PLUS UTILISÉES

1- Syntaxe simple :

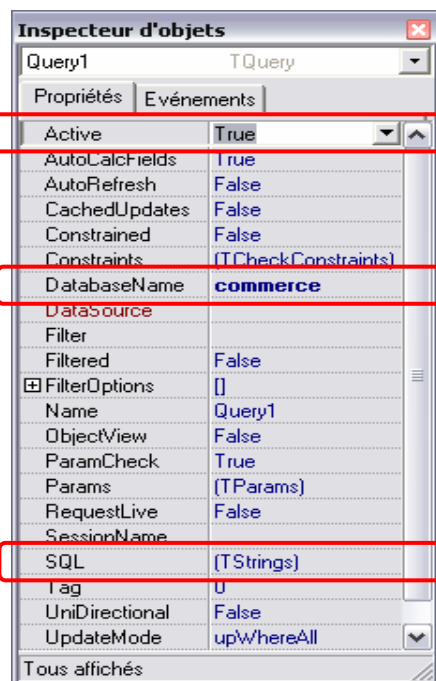
Soit la table produit (code, designation, quantite_stock, prix_unité, type_produit)

On veut sélectionner tous les champs et enregistrements de cette table

Syntaxe :

```
select *
from produit
```

Pour executer ce script on insere un composant Query  (palette accèsBD) et on manipule ses propriétés :



On test le fonctionnement de ce script SQL. Avant de rendre cette propriété **True**, ce composant compile ce script SQL et ne donne pas true que si la requete a une syntaxe juste.

1 On se connecte à notre Alias.

2 On ecrit le script SQL dans cette propriété.

Remarque :

Select : sélectionne un ou plusieurs champs en mentionnant leurs noms (* veut dire tous les champs)
On peut aussi Introduire les champs calculés en donnant leurs noms et la formule de calcul.
From : décide la source de données à sélectionner (la table source de données).
Where : donne un ou plusieurs critères de sélection des enregistrements.

2- requête avec critère de sélection :

Si on veut sélectionner que les produits qui ont une quantité de stock supérieur à 10, on écrit :

```
select *
from produit
where type_produit = 'consommables'
```

Donc : la clause where donne la possibilité d'introduire un critère de sélection qui sera généralement une comparaison d'un champ à une valeur de même type.

3- requête avec paramètre :(requête dynamique) :

Si on veut que la valeur du critère de sélection soit variable :

Exemple : à chaque fois on donne une valeur et on sélectionne les quantités supérieures à cette valeur donc on doit avoir une variable qui contiendra à chaque fois une valeur différente :

Syntaxe :

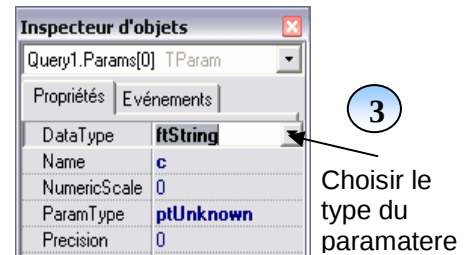
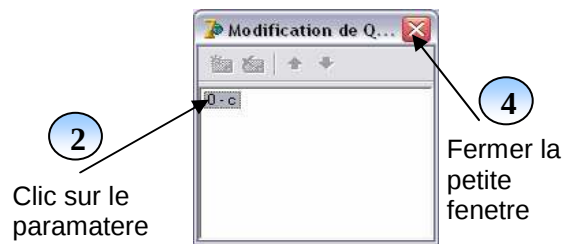
```
select *
from produit
where type_produit = :c
```

Remarque1:

- le paramètre ici est nommé **C** et les deux points qui le précède indiquent que c'est un paramètre.
- le paramètre en suite doit avoir un type adéquat (compatible au type du champ utilisé dans la comparaison). On donne un type au paramètre par la propriété :



Clic Ici

**Remarque2:**

Il faut choisir après un événement adéquat pour donner une valeur au paramètre par les instructions suivantes :

```
Query1.close ;  
Query1.parambyname('c').asstring:=combobox1.text;  
Query1.active:=true;
```

Pour cela on peut insérer un composant Combobox1 (palette standard) et écrire dans sa propriété Items les valeurs : matériel, consommables, meuble. Ensuite on double clic sur ce composant et on introduit le code source précédent.

4- les requêtes contenant des champs calculés :

On peut introduire un champs calculé dans une requête en le mentionnant dans la partie *select* avec les champs fixe. Et cela en donnant le nom du champ précédé de la formule qui le calcul.

Exemple : on veut calculer le prix total de chaque produit à savoir sa quantité_stock.

Syntaxe :

```
Select code, designation, quantite_stock, prix_unité, (quantite_stock * prix_unité)as Total  
From produit
```

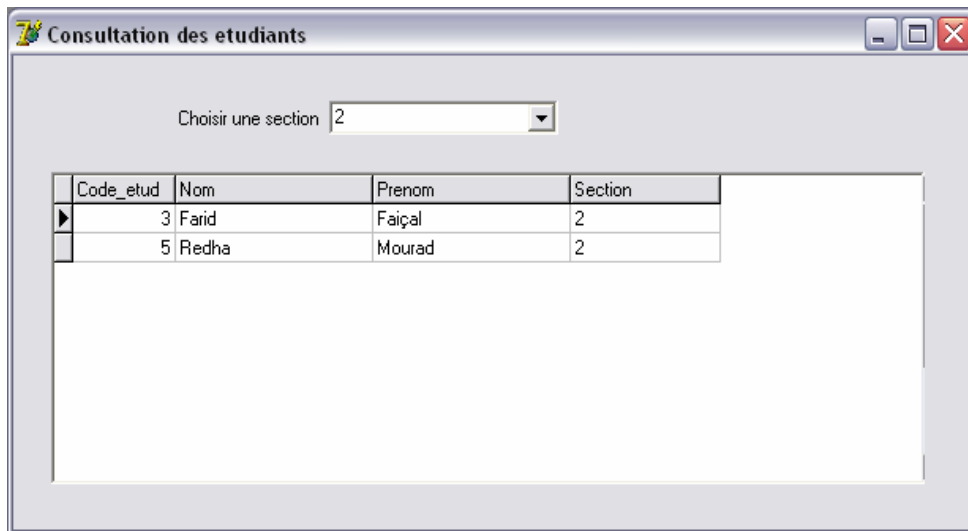
5- les requêtes des statistiques :

On peut préparer une table contenant par exemple les type des produits (exemple : matériel, consommables, meuble) et avoir comme résultat de la requête des données prêtes à être représentées par un graphe des statistiques de réussite.

Syntaxe:

```
Select type_produit, count(type_produit) as nombre  
From produit  
Group by type_produit
```

TP2 : LA CONSULTATION PAR DES REQUÊTES PARAMÉTRÉES



1- dans le module base de données créer la table suivante :

Champ	type	taille	index
Code_etud	I		*
Nom	A	20	
Prénom	A	20	
Section	A	15	

On enregistre cette table dans notre alias "dbdemos" sous le nom : « etudiant2007 ».

2- dans une fiche Delphi insérer les composants suivants :

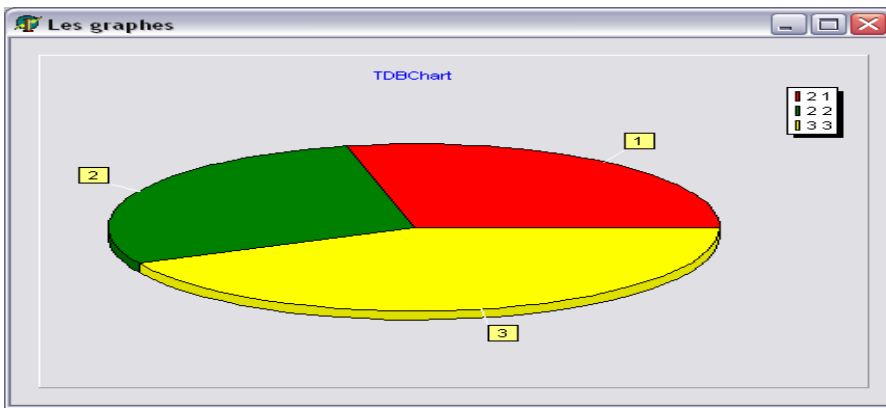
Palette	Composant	Propriété	Valeur
BDE	Query1 	Databasename	dbdemos
		SQL 	Select * From etudiant2007 Where section = :sec
		params	Sec ⇨ datatype → ftstring
		Active	True
AccesBD	Datasource1	Dataset	Query1
ControleBD	Dbgrid1	Datasource	Datasource1
Standard	Label1	caption	Choisir une section
	Combobox1	text	
		Items 	A B C

3- Double clique sur le Combobox1 et écrire dans sa procédure :

```
query1.close ;  
query1.parambyname('sec').asstring :=combobox1.text ;  
query1.active:=true;
```

4- lancer l'exécution et choisir dans le Combobox1 la section que vous voulez afficher ses étudiants.

TP3 : LES GRAPHES



1- Garder la table etudiant2007 du TP2 :

2- Dans une fiche Delphi insérer les composants suivants :

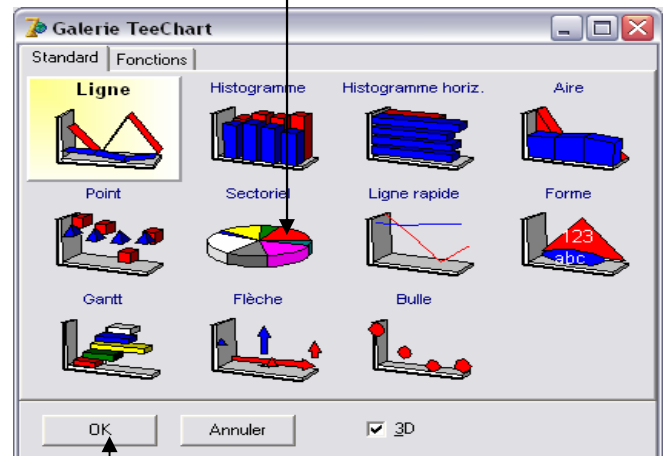
Palette	Composant	Propriété	Valeur
BDE	Query1	Databasename	dbdemos
		SQL	Select section , count(section) as nombre From etudiant2007 Group by section
		Active	True

3- insérer de la palette « controleBD » le composant Dbchart1  et procéder comme suite :

- Double clique sur ce composant

1 Clic sur le bouton **Ajouter**

2 Choisir le type du graphe **Sectoriel**



3 Clic sur **Ok**

4 Choisir L'onglet **Série**

5 Choisir l'onglet **Source de données**

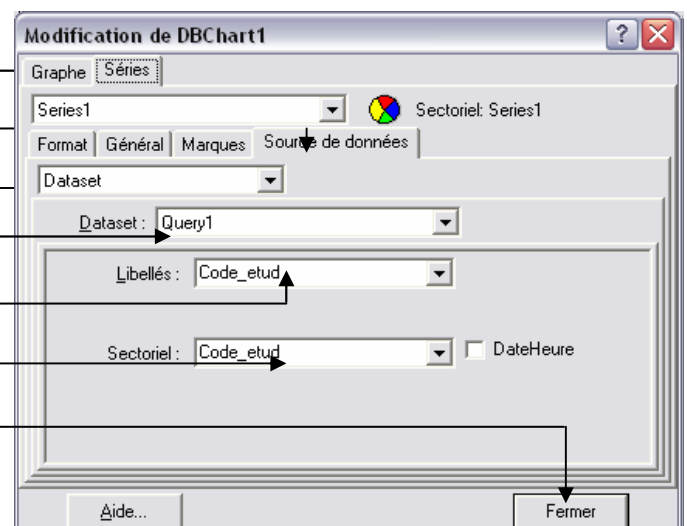
6 Choisir dans la liste **DataSet**

7 Choisir dans la liste **Query1**

8 Choisir le champ alphanumérique

9 Choisir le champ numérique

10 Enfin clic que **Fermer**



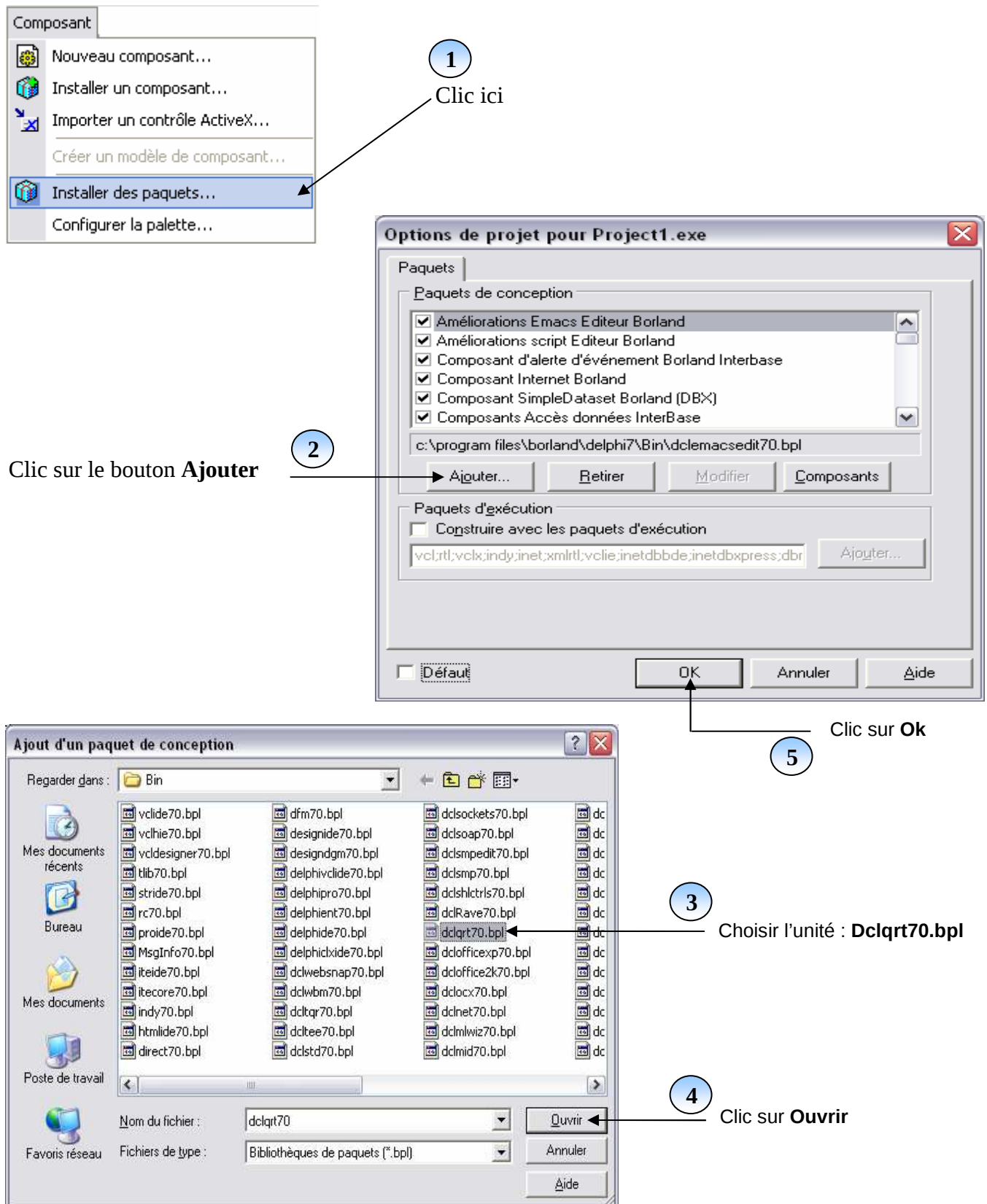
LEÇON 9 : LES ÉTATS D'IMPRESSION

INTRODUCTION

Les versions Delphi qui ont précédé Delphi7 ont proposés les composants Quickreport pour imprimer les données. On présentera maintenant les étapes d'installation des composants quickreport en Delphi7.

Installation des composants Quickreport en Delphi7 :

Choisir dans le menu Delphi : Composant ⇒ Installer des packets.



TP1 : Imprimer les enregistrements d'une Table (Liste des produits)

Procédés :

- insérer un composant Quickrep1 (de la palette Qreport) sur la fiche.
- La source de données à imprimer sera la table produit, Donc insérer de la palette BDE le composant « Table1 » et dans son inspecteur d'objet on manipule ses propriétés :

Choisir notre alias DatabaseName **Test**

Choisir la table TableName **Produit.db**

Activer la connexion à la table Active **True**

- on sélectionne le composant Quickrep1 posé sur la fiche pour avoir son inspecteur d'objet et on manipule dans ce dernier les propriétés comme suite :
 - pour modifier la taille d'affichage de la page d'impression « quickrep1 » on écrit dans la propriété :

Zoom **70**

- on ouvre la propriété Bands en cliquant sur l'icône  à son gauche

Clic Ici →  Bands (TQuickRepBands)

- on doit mettre à **true** les sous propriétés suivantes :

Bands	(TQuickRepBands)
HasColumnHeader	True
HasDetail	True
HasPageFooter	True
HasPageHeader	False
HasSummary	False
HasTitle	True

Hascolumnheader : bande d'entêtes de colonnes.

Hasdetail : pour avoir une bande de détail (cette bande a une caractéristique spéciale de se répéter suivant le nombre d'enregistrements).

Hastitle : pour avoir une bande de titre d'état.

- on élargi la hauteur de la bande de titre d'état.
- dans la palette des composants du Qreport on insère autant de composants Qrlabel nécessaire dans la bande titre pour bien représenter l'organisme propriétaire de l'état (société, commerçant ...); ces composants contiendront des textes explicatives dans leurs propriétés Caption.
 - Dans la bande entête des colonnes on insère des composants Qrlabel représentant par le texte de leur propriété Caption le nom des colonnes à imprimer (code, designation, quantité...).
 - Dans la bande détail on insère toujours de la palette Qreport des composants QrDbText qui seront positionnées respectivement juste en dessous des composants Qrlabel de la bande entête de colonnes(un par un).
 - chaque composant Qrdbtext sera lié à un champ de la table produit à imprimer et suivant le Caption du Qrlabel au dessus. La liaison du Qrdbtext avec le champ se fait :

- on se pointant par sa propriété DataSet à la table produit.

DataSet **Table1**

- on se pointant par sa propriété DataField au champ adéquat.

DataField **Code**

- en fin n'oubliez pas de lier le composant Quickrep1 lui même à la table produit. Cela se fait en cliquant sur ce composant en évitant les bandes et dans son inspecteur d'objet on se pointe par la propriété dataset à la table produit.
- Pour previsualiser notre état on écrit dans la procédure d'un bouton (son caption sera : « previsualiser ») l'instruction :
quickrep1.preview ;
- Pour imprimer notre état on écrit dans la procédure d'un bouton (son caption sera : « imprimer ») l'instruction :
quickrep1.print ;

- Pour encadrer les champs de l'état d'impression on utilise le composant Qrshape et on le contrôle par :

Ctrl + **←** : déplacer le composant sélectionné à gauche.
Ctrl + **→** : déplacer le composant sélectionné à droite.
Ctrl + **↑** : déplacer le composant sélectionné vers le haut.
Ctrl + **↓** : déplacer le composant sélectionné vers le bas.
Shift + **←** : réduire le composant sélectionné à gauche.
Shift + **→** : agrandir le composant sélectionné à droite.
Shift + **↑** : réduire le composant sélectionné vers le haut.
Shift + **↓** : agrandir le composant sélectionné vers le bas.

Remarque : pour mettre le cadre (QRShape) en arrière plan et voir les autres composants (QRLabel, QREdit) on doit cliquer par le bouton droit de la souris sur ce cadre et choisir dans le menu contextuel :

Options de contrôle ⇒ mettre en arrière plan.

Clic par le bouton Droit de la souris sur l'état d'impression en évitant les bandes et les composants

Clic sur **Prévisualiser**

L'état sous sa forme d'impression

Prévisualisation d'impression

SOCIETE COMMERCE MULTIPLE

LISTE DES PRODUITS

Reference	Designation	Qte_stock	Prix	TVA
1	Ordinateur	85	20	Fonction
2	Imprimante	32	45	Fonction
3	Scanneur	3560	20	Fonction
4	Table	31	84	Fonction
5	Timbre	11	30	Fonction

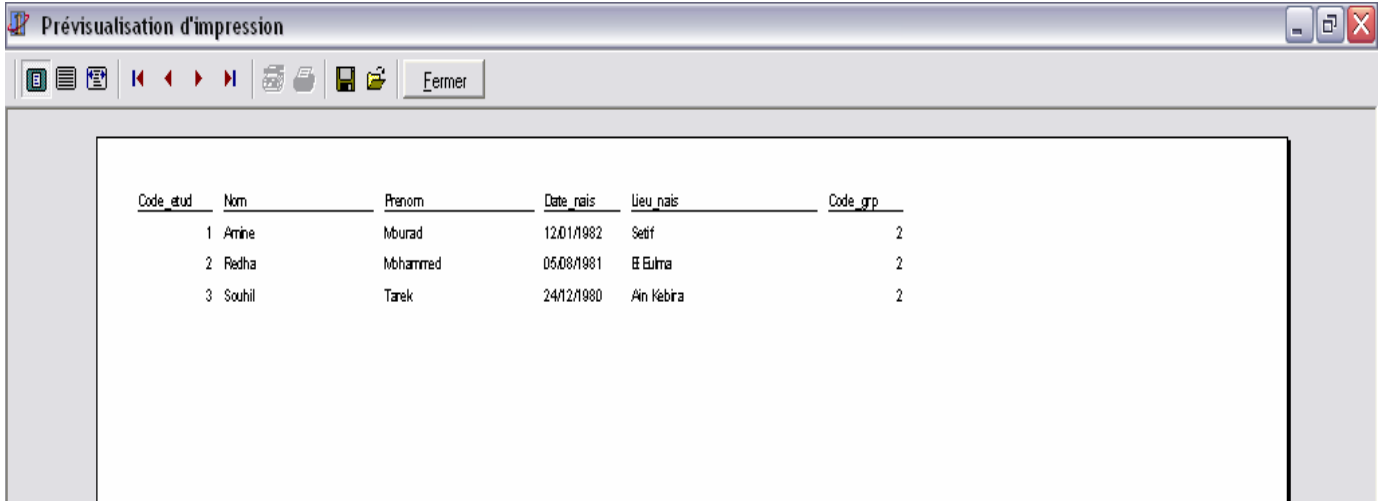
Page 1 sur 1

LEÇON 10 : DELPHI EST UN RAD

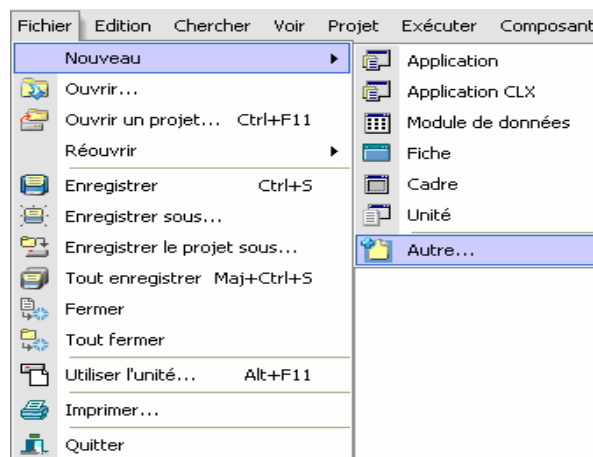
TP1 : UNE FICHE SIMPLE D'IMPRESSION D'UNE TABLE

Delphi est un RAD (**R**apide **A**pplication **D**éveloppement) et propose des assistants qui aident à la création d'une grande variété de fiches.

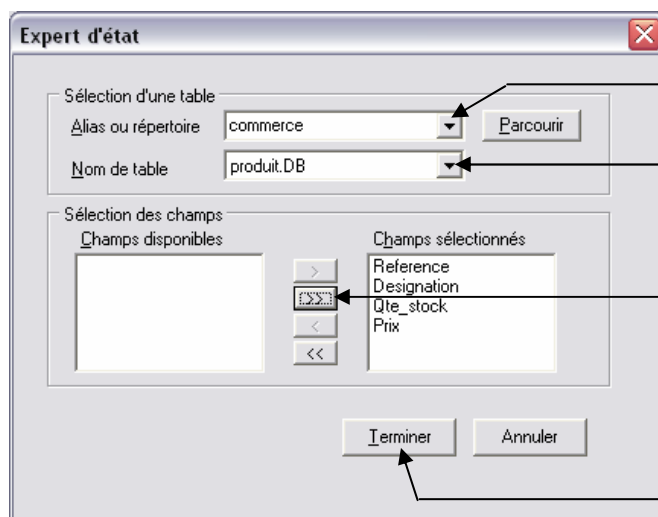
Pour la création d'une fiche d'impression (image suivante) on procède comme suite :



1. Dans le menu Delphi on choisit : Fichier ⇒ Nouveau ⇒ Autre ..

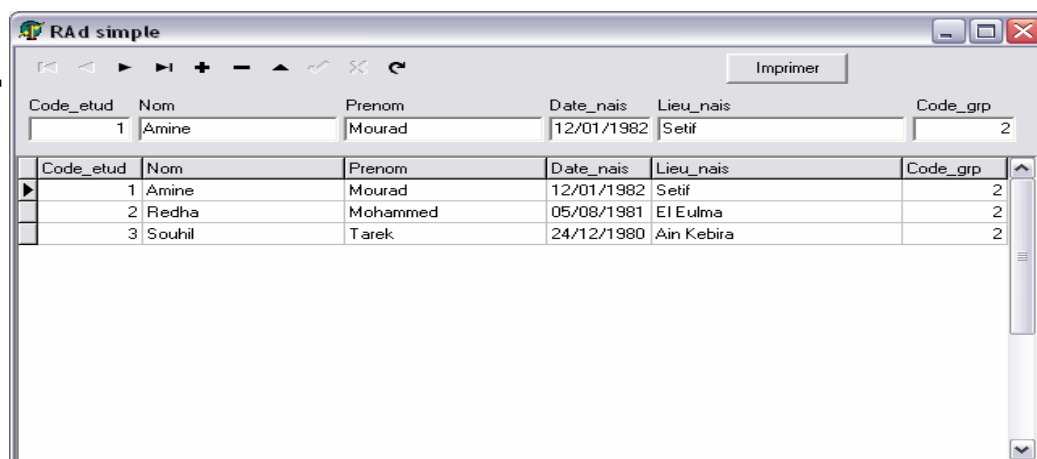


2. Dans la boîte de dialogue qui s'affiche on choisit l'onglet « affaires » et dans ce dernier on choisit l'icône « expert quick report » et on clique sur le bouton **Ok**.
3. Dans l'assistant qui s'affiche on clique en bas sur le bouton **lancer l'expert**.



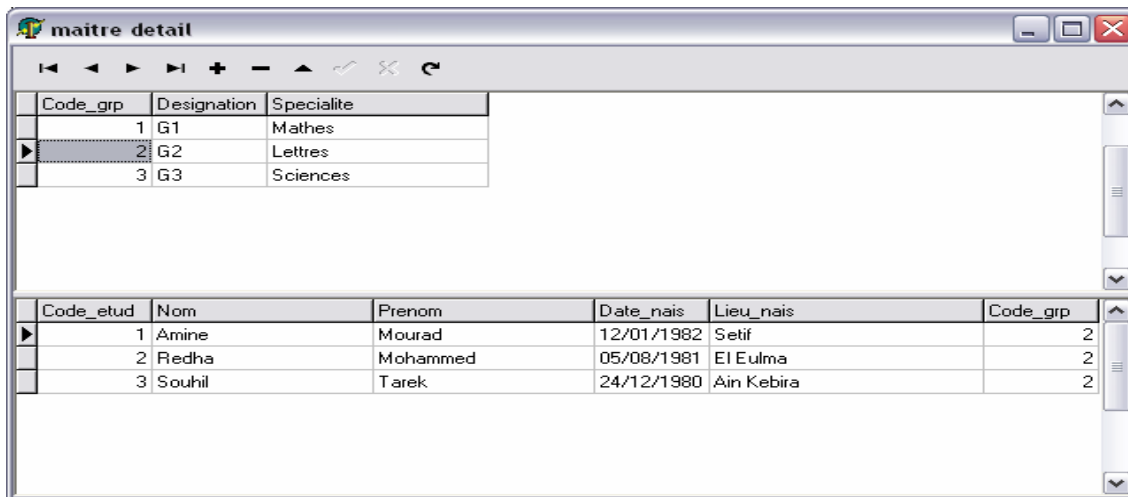
- 1 Choisir l'alias de la base de données
- 2 Choisir la table à imprimer
- 3 Faire passer tous les champs à droite pour les imprimer tous (si on veut imprimer quelques champs seulement, on les fait passer un par un par le bouton plus haut **>**)
- 4 À la fin on ferme la petite fenêtre

TP2 : UNE FICHE SIMPLE DE CONNEXION ET D'AFFICHAGE ET D'IMPRESSION D'UNE TABLE



- 1- Pour créer cette fiche on procède comme suite :
 - Dans le menu Delphi on choisit : fichier ⇒ nouveau
 - Dans la boîte de dialogue qui s'affiche on choisit l'onglet « affaires » et dans ce dernier on choisit l'icône « expert fiche base de données » et clique sur le bouton **OK**.
 - Dans l'assistant qui s'affiche on choisit dans la première étape le type de la fiche :
 - Créer une fiche simple
 - Clique sur le bouton **Suivre**.
 - Choisir dans la liste « lecteur ou nom de l'alias » notre alias « scolarité »
 - Clique sur la table « étudiant » et clique sur **Suivant** et clique sur le bouton **>>** et clique sur **Suivant**.
 - Clique sur l'option « ○ dans une grille » et clique sur **Suivant**.
 - Enfin clique sur l'option « ○ fiche seulement » et clique sur **Terminer**
- 2- Clique sur le composant « dbgrid1 » et choisir dans sa propriété « align » ⇒ none, et réduire la taille de ce dbgrid vers le bas; et clique sur le composant table1 et modifier sa propriété : « Active » ⇒ true.
- 3- Double clique sur le composant table1 et clique droit sur la petite fenêtre et clique sur « ajouter tous les champs »
- 4- Glisser les champs un par un vers l'espace libre en haut de notre fiche.
- 5- Clique sur la nouvelle fiche en évitant le clic sur l'un de ses composants et écrire dans sa propriété : name ⇒ fiche_etudiant.
Ensuite, clique sur (menu Delphi) : fichier ⇒ enregistrer, et nommer l'unité : etudiant.
- 6- On doit maintenant préparer l'impression de la liste des étudiants comme suite :
 - Choisir dans le menu Delphi : fichier ⇒ nouveau et clique dans la fenêtre qui s'affiche sur l'onglet « affaires ».
 - Clique sur l'élément : « expert quickreport » et clique sur **OK**.
 - Clique sur le bouton « lancer l'expert », et choisir l'alias « scolarité » et la table « etudiant ».
 - Clique sur le bouton **>>** qui fait passer tous les champs vers la droite pour les imprimer tous et clique sur **OK**.
 - Clique sur la nouvelle fiche en évitant le clic sur le composant « quickrep1 » et écrire dans sa propriété : name ⇒ fiche_etat_etudiant.
Ensuite, clique sur (menu Delphi) : fichier ⇒ enregistrer, et nommer l'unité : etat_etudiant.
- 7- On tape **Shift+F12** pour afficher la liste des fiches et on choisit la fiche « fiche_etudiant » et on clique sur **OK**.
Ensuite, et pour une impression bien présentée des étudiants on insère un composant popupmenu (palette standard).
Ensuite, on double clique sur ce composant et on introduit les deux éléments « previsualiser » et « imprimer ».
Ensuite, on ferme la petite fenêtre du « popupmenu » et on insère de la palette « supplément » le composant « speedubutton1 » et on clique sur ce composant qui représente l'impression et dans son inspecteur d'objet on choisit dans sa propriété « popupmenu » l'élément « popupmenu1 ».
Ensuite : pour programmer le composant popupmenu1 on double clique sur ce dernier et on double clique sur l'élément « previsualiser » et on écrit dans sa procédure :
fiche_etat_etudiant.quickrep1.preview ;
 Ensuite : on tape **F12** pour retourner à la fiche et on double clique sur le composant « popupmenu1 » et on double clique sur l'élément « imprimer » et on écrit dans sa procédure :
fiche_etat_etudiant.quickrep1.print ;
 Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_etudiant et la fiche_etat_etudiant et on doit cliquer sur **Oui** pour chaque demande de confirmation.

TP3 : UNE FICHE MAITRE-DETAIL ENTRE DEUX TABLES



Pour la création de cette fiche on procède comme suite :

1. dans le menu Delphi on choisit : fichier ⇒ nouveau
2. dans la boîte de dialogue qui s'affiche on choisit l'onglet « affaires » et dans ce dernier on choisit l'icône « expert fiche base de données » et clique sur le bouton **Ok**.
3. dans l'assistant qui s'affiche on choisit dans la première étape le type de la fiche :
☒ Créer une fiche maître/détail, ensuite clique sur le bouton **Suite**.
4. choisir dans la liste « lecteur ou nom d'alias » l'alias de notre table maître à afficher (exemple : dbdemos).
5. On choisit la table dans la liste (exemple : groupe), ensuite clique sur le bouton **Suite**.
6. On clique sur le bouton **>>** pour faire passer tous les champs à droite (pour les afficher tous) et clique sur **Suite**
7. Clique sur l'option : « ☒ dans une grille » et clique sur **Suite**
8. Choisir dans la liste « lecteur ou nom d'alias » l'alias de notre table détail à afficher (exemple : dbdemos).
9. On choisit la table dans la liste (exemple : etudiant), ensuite clique sur le bouton **Suite**.
10. On clique sur le bouton **>>** pour faire passer tous les champs à droite (pour les afficher tous) et clique sur **Suite**
11. Clique sur l'option : « ☒ dans une grille » et clique sur **Suite**
12. Ensuite : on choisit dans la liste en haut l'index disponible « c6 »
13. Ensuite : on choisit à gauche le champ « code_grp » et à droite le champ « code_grp ».
14. Ensuite : clique sur le bouton **ajouter** ensuite sur le bouton **Suite**.
15. On fin sur le bouton **Terminer**

TP4 : TP EXEMPLAIRE

- 1- choisir dans le menu Delphi : fichier ⇒ nouvelle application.
(Si Delphi demande l'enregistrement de l'ancienne application on clique sur **Non**)
- 2-supprimer la fiche vierge proposée par défaut dans cette nouvelle application par :
Le menu Delphi : projet ⇒ supprimer une fiche, et clique sur **Ok**
- 3- préparer une fiche BD simple en utilisant les étapes du TP2 et nommer cette fiche « fiche_liste ».
- 4- préparer une fiche d'impression en utilisant les étapes du TP1 pour imprimer la même table choisie dans la fiche_liste et nommer cette fiche « fiche_etat_liste ».
- 5- insérer en haut de la fiche_liste deux boutons de la palette standard.
 Button1 ⇒ Caption ⇒ Previsualiser.
 Button2 ⇒ Caption ⇒ Imprimer.
- 6- double clique sur le button1 et écrire dans sa procédure :
fiche_etat_liste.quickrep1.preview ;
- 7- double clique sur le button2 et écrire dans sa procédure :
fiche_etat_liste.quickrep1.print ;
- 8- compiler par **Ctrl + F9** et clique sur **Oui** pour faire la liaison entre les deux fiches et lancer l'exécution par **F9**.

CHAPITRE 2

LOGICIEL

GESTION

COMMERCIALE

**PREPARATION DU LOGICIEL :**

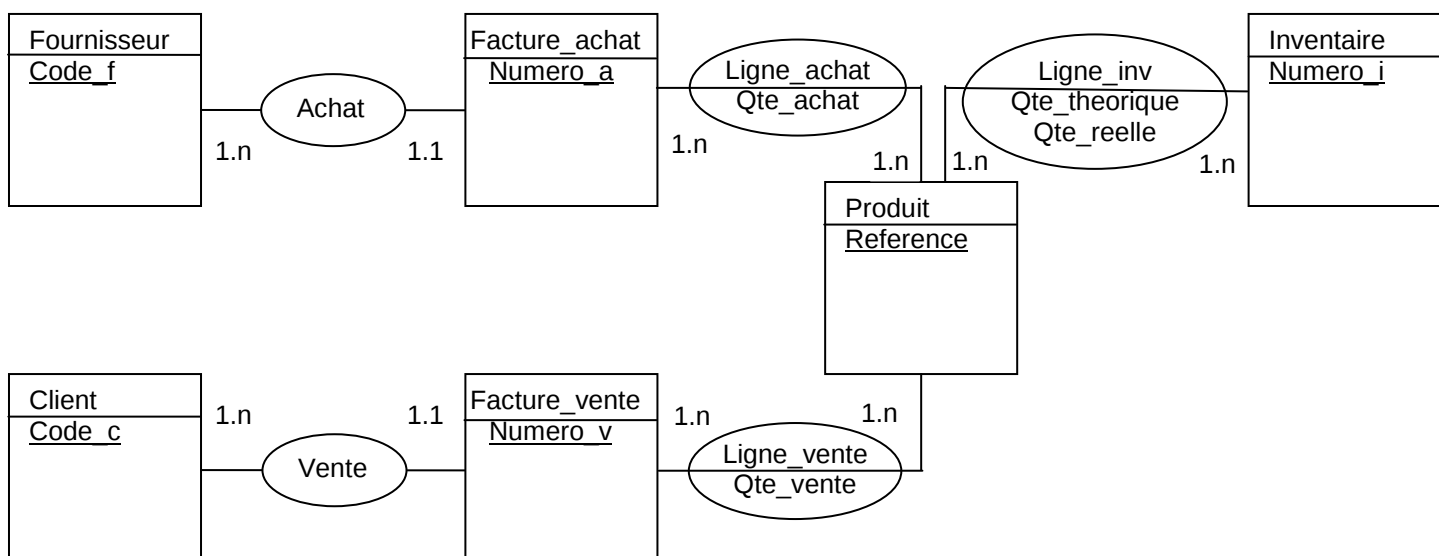
- 1- créer dans le lecteur C du disque dur un nouveau dossier nommé : commerce (pour stocker le logiciel).
 - 2- créer dans le dossier commerce 2 autres dossiers nommés : base (pour stocker la BD) et source (pour stocker le code source)
 - 3- lancer le BDE de Delphi par : (le menu Delphi) outils ⇒ module base de données.
 - 4- créer un nouveau alias pour notre logiciel par : (le menu BDE) outils ⇒ gestionnaire d'alias
- Ensuite : cliquer sur l'option : « ☒ afficher seulement les alias publics »
- Ensuite : cliquer sur le bouton : nouveau
- Ensuite : écrire dans : Alias de la base : commerce
Chemin d'accès : c:\commerce\base
- Ensuite cliquer sur Ok
- En fin cliquer sur Oui pour que cet alias soit un alias public.

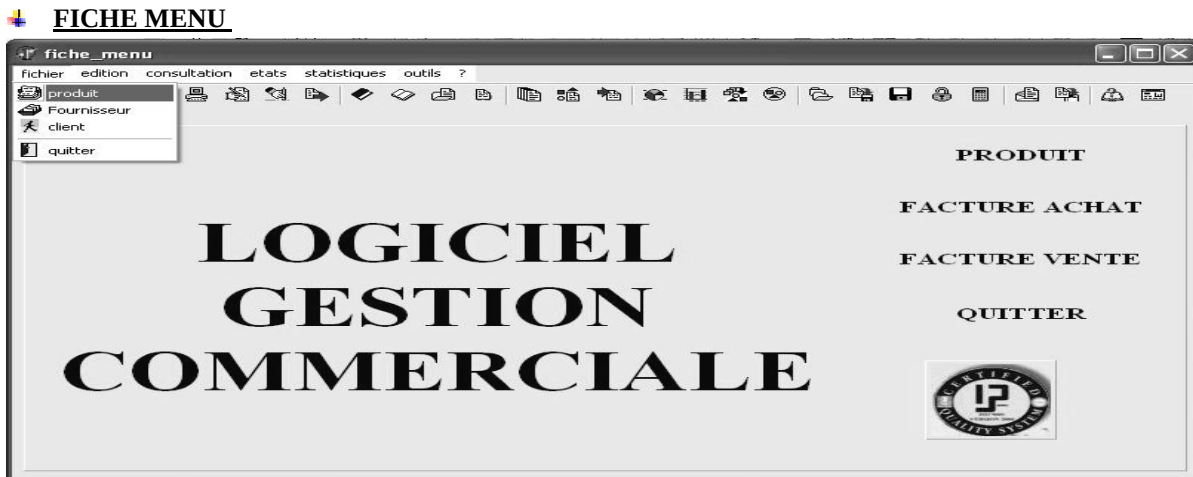
Remarque

On peut fixer cet alias comme alias de travail par défaut au cour du développement de notre logiciel et cela par le choix de cet alias dans le menu du module base de données : fichier ⇒ répertoire du travail.
(Le BDE se pointera automatiquement sur cet alias à chaque accès au module BD)

- 5- créer dans cet alias les tables suivantes, n'oubliez pas de les enregistrer dans l'alias « commerce »

Table: client				Table : facture_achat				Table : produit			
Champ	Type	Taille	index	Champ	Type	Taille	index	Champ	Type	Taille	index
Code_c	i		*	Numero_a	i		*	Reference	i		*
Nom_c	A	20		Date_achat	d			Designation	A	20	
Prenom_c	A	20		Reglement_a	A	10		Qte_stock	n		
Adresse_c	A	30		code_four	i			Prix_achat	\$		
Telephone_c	A	15		Table : facture_vente				Prix_vente	\$		
RC_c	A	15		Champ	Type	Taille	index	Tva	n		
AI_c	A	15		Numero_v	i		*	Table : inventaire			
IF_c	A	15		Date_vente	d			Champ	Type	Taille	Index
Table: fournisseur				Reglement_v	A	10		Numero_i	i		*
Champ	Type	Taille	index	code_cli	i			Date_inventaire	d		
Code_f	i		*	Table : ligne_vente				durée	A	10	
Nom_f	A	20		Champ	Type	Taille	Index	responsable	A	20	
Prenom_f	A	20		Num_vente	i		*	Table : ligne_inventaire			
Adresse_f	A	30		Ref_prod	i		*	Champ	Type	Taille	Index
Telephone_f	A	15		Qte_vente	n			Num_inv	i		*
RC_f	A	15		Table : ligne_achat				Ref_prod	i		*
AI_f	A	15		Champ	Type	Taille	index	Qte_theorique	n		
IF_f	A	15		Num_achat	i		*	Qte_reelle	n		
				Ref_prod	i		*	Table protection			
				Qte_achat	n			Champ	Type	Taille	index
								Password	A	20	*

**MCD CORRESPONDANT (c'est seulement une partie du MCD réel et complet de la gestion commerciale) :**



1- Retournez à l'environnement Delphi et choisir dans le menu : Fichier ⇒ nouvelle application.
 Ensuite : insérer dans la fiche de ce nouveau projet (ce nouveau logiciel) un composant « mainmenu1 »
 Ensuite : double clique sur ce composant et insérer le menu principal suivant :

Fichier	Edition	Consultation	Statistiques	Outils	?
Produit	Facture achat	Achats	Règlements	Inventaire	Aide
Client	Facture vente	Ventes		Archivage	A propos
Fournisseur				Protection	
				Cloture	
Quitter					

- ❖ Chaque fiche (ainsi que son unité) dans le logiciel doit être nommée par un nom significatif. pour notre fiche actuelle (le menu général du projet) on procède comme suite :

Cliquer sur cette fiche (éviter le composant mainmenu1) et dans son inspecteur d'objet écrire dans la propriété : name ⇒ fiche_menu.

Cliquer dans le menu Delphi sur : fichier ⇒ enregistrer sous, et dans le dossier « c:\commerce\source » on nomme l'unité : « menu ».

Cliquer dans le menu Delphi sur : fichier ⇒ enregistrer le projet sous, et nommer le projet : commerce (toujours dans « c:\commerce\source »)

2- insérer un composant ToolBar1 (palette win32).

Ensuite : clique droit sur ce composant et choisir « nouveau bouton » ou « nouveau séparateur »

Ensuite : répéter le clic droit pour insérer 15 boutons qui représentent les éléments du menu principal du logiciel;

Et 6 séparateurs pour les grands éléments du menu (fichier, édition, consultation, ...)

Ensuite : insérer un composant imagelist1 (palette win32)

Ensuite : double clique sur ce composant et clique sur le bouton **Ajouter** pour choisir la gamme des images de notre logiciel dans le chemin : « C:\program files\fichiers communs\borland shared\images\buttons »

Ensuite : après le choix de 15 images adéquates aux 15 boutons on clique sur **OK**

Ensuite : clique sur le « mainmenu1 » et dans son inspecteur d'objet choisir dans la propriété :

Images ⇒ imagelist1

Ensuite : double clique sur le « mainmenu1 » et chaque élément de ce menu sera lié à une image adéquate par sa propriété : « imageindex »

Enfin : fermer la petite fenêtre du menu et clique sur le composant « toolbar1 » et pour affecter des images à ses boutons choisir dans sa propriété : Images ⇒ imagelist1.

(Chaque bouton du « toolbar1 » peut avoir une image adéquate par le choix de cette image dans la propriété « imageindex » de ce bouton).

3- pour expliquer le rôle de chaque bouton on clique sur ce dernier on écrit dans sa propriété : « Hint » l'explication voulue, ensuite et pour voir cette explication on choisit dans la propriété : showHint ⇒ true

Programmation du sous menu : Quitter

Double clique sur le composant « mainmenu1 » et double clique sur son élément quitter (du sous menu fichier)

Ensuite : écrire dans la procédure : **close;**

Ensuite : dans la fiche_menu, double clique à droite de son événement « onclosequery » et écrire dans sa procédure : **if messagedlg('voulez vous quitter le logiciel?',mtconfirmation,[mbyes,mbno],0)=mryes then**

begin

canclose:=true;

application.terminate;

end

else

canclose:=false;

Résultat:

Cette technique assure l'affichage de la fenêtre de confirmation de sortie du logiciel pour n'importe quel type de tentative de fermeture du logiciel :

- le menu : fichier ⇒ quitter.
- Par clavier en utilisant les touches : **Alt + F4**
- En cliquant sur la case fermer ☐ en haut et à droite de la fenêtre.

4- Pour que la fiche menu occupe tous l'espace de l'écran lors du lancement de l'exécution du logiciel :

- Cliquez sur cette fiche_menu (éviter de cliquer sur l'un de ses composants).
- Choisir dans sa propriété : « Window state ⇒ wsmaximized »

5- Enregistrer notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

Remarques :

1- Pour lier chaque bouton de la barre d'outils « Toolbar1 » à une fiche on clique sur ce bouton et on choisit dans son événement « onclick » la procédure qui appelle la fiche correspondante (faites ce travail à la fin du logiciel).

2- Lorsqu'on lance l'exécution de notre logiciel la fiche menu apparaît automatiquement la première puisque elle est la fiche principale dans notre logiciel, mais si cette fiche n'a pas été lancée la première donc elle n'est pas considérée comme fiche principale et on doit corriger cette faute par le choix dans le menu Delphi : projet ⇒ options, et on choisit dans la liste : « Fiche principale » ⇒ « fiche_menu » et clique sur .

3- dans le menu principal :

✓ La partie fichier: sert à représenter les tables permanentes (produit, clients, fournisseur) qui se ressemblent côté programmation.

✓ Une table permanente est une table qui lors de l'archivage de notre BD pour le passage à une nouvelle année d'activité, ses données ne seront pas vidées (conservent les mêmes données pour le nouvel an)

✓ La partie édition : sert à représenter les mouvements entre les tables permanentes. (bien sur les mouvements sont représentés sous formes de tables de mouvement (facture_achat et ligne_achat par exemple)

Une table mouvement après son archivage (après copiage de son contenu dans un autre emplacement mémoire comme réserve) doit être vidée pour commencer une nouvelle année d'activité.

L'image de l'arrière plan :

1- insérer un composant « image1 » de la palette « supplément ».

2- dans sa propriété picture on clique sur le bouton et chercher l'image voulue.

(Exemple du chemin de l'image : C:\Program Files\Fichiers communs\Borland Shared\Images\Splash\256Color)

3- Enfin clique sur ensuite sur

4- pour que l'image occupe toute la surface de son cadre on modifie sa propriété : Autosize ⇒ true.

5- pour que l'image s'agrandisse si elle a une taille plus petite que la surface de son cadre on modifie sa propriété : Stretch ⇒ true. Aussi modifier sa propriété : align ⇒ alClient.

Les abréviations du domaine commercial :

HT : prix Hors Taxe.

TVA : la Taxe sur la Valeur Ajoutée.

TTC : le montant Tous Taxes Comprise.

RC : le numéro du Registre de Commerce

AI : le numéro de l'Article d'Imposition

IF : le numéro de l'Identification Fiscale.

LA FICHE PRODUIT

Reference	Designation	Qte_stock	Prix_achat	Prix_vente	Tva	total
1	Ordinateur	85	23 000,00 Da	24 000,00 Da	14	1 955 000,00 Da
2	Imprimante	32	7 500,00 Da	7 900,00 Da	17	240 000,00 Da
3	Scanneur	30	4 600,00 Da	5 000,00 Da	14	138 000,00 Da
4	Table	31	3 900,00 Da	4 300,00 Da	17	120 900,00 Da
5	Chaise	210	1 250,00 Da	1 700,00 Da	14	262 500,00 Da

montant_stock
2716400 Da

1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom de la fiche : « fiche_produit ».
Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité : « produit »

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Commerce
		Tablename	Produit
		Active	True
Accesbd	Datasource1	Dataset	Table1
Contrôlebd	Dbgrid1	Datasource	Datasource1
Supplement	Speedbutton1	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose
BDE	Query1	databasename	Commerce
		Sql	Select sum(qte_stock*prix_achat)as montant_stock from produit
		Active	True
Standard	Label1	Caption	Filtrage
	Edit1	Text	

3- Double clique sur le query1 et clique droit sur la petite fenêtre et clique sur «ajouter tous les champs».

Ensuite : clique sur le champ montant_stock et glisser ce dernier vers le bas de notre fiche.

4- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange):

```

if edit1.text <> '' then
begin
table1.close ;
table1.filtered :=true ;
table1.filter :='[designation]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;

```

5- On tape la touche **F12** (retour à la fiche), et pour faire le trie de la table produit : clique sur le composant dbgrid1.

Ensuite : dans son inspecteur d'objet on clique sur l'onglés « événements »

Ensuite : on double clique à droite de l'événement « onclick » et on introduit le code source suivant :

// Une table est triée automatiquement selon son index actuel (primaire ou secondaire)

```

If column.index = 0 then
Table1.indexname := 'c1' ;
If column.index = 1 then
Table1.indexname := 'c2' ;
If column.index = 2 then
Table1.indexname := 'c3' ;
If column.index = 3 then
Table1.indexname := 'c4' ;
If column.index = 4 then
Table1.indexname := 'c5' ;
If column.index = 5 then
Table1.indexname := 'c6' ;

```

Remarque importante :

Le trie et le filtrage dans une table ne réussissent que si on a défini des index secondaires sur chaque champ trié ou filtré.

Pour insérer des index secondaires dans les champs de la table produit il faut que cette table ne soit pas en cours d'utilisation donc sa propriété : active ⇒ false.

Aussi la requête query1 doit être : active ⇒ false, puisque elle aussi utilise la table produit.

Ensuite : on lance le BDE Delphi par : (le menu Delphi) : outils ⇒ module base de données.

Ensuite : on ouvre la table par : (le menu) fichier ⇒ ouvrir ⇒ table et on choisit l'alias « commerce » et la table « produit ».

Ensuite : on clique dans le menu du BDE sur table ⇒ restructurer.

Ensuite : on choisit dans la liste des choix « propriétés de la table » l'option « index secondaires »

Ensuite : on clique pour chaque champ sur le bouton **Définir** et on fait passer ce champ à droite, on clique sur le bouton **OK** et on propose un nom symbolique à cet index secondaire (c1 pour le champ reference, c2 pour le champ designation,...etc.)

Ensuite : on clique sur le bouton **enregistrer** et on retourne à l'environnement Delphi.

Enfin : on réactive les deux composants table1 et query1 par leurs propriétés : active ⇒ true.

6- on renforce cette fiche par des champs calculés pour chaque enregistrement de la table produit et cela comme suite :

On double clique sur le composant table1

Ensuite : on clique droit sur la petite fenêtre et on clique sur « ajouter tous les champs ».

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « total » et on choisit le type du champ : « currency » et on clique sur **ok**.

Ensuite : on clique sur le composant table1 et dans son inspecteur d'objet on clique sur l'onglé « événements ».

Ensuite : on double clique à droite de l'événement « oncalcfields » et on écrit dans la procédure :

table1.fieldbyname('total').ascurrency := table1.fieldbyname('qte_stock').asfloat *

table1.fieldbyname('prix_achat').ascurrency ;

query1.close;

query1.active:=true;

// Pour que la requête prend en considération les dernières données saisies (rafraîchir les calculs de la requête).

7- on va maintenant préparer l'impression de la liste des produits comme suite :

On choisit dans le menu Delphi : fichier ⇒ nouveau, et on clique dans la fenêtre qui s'affiche sur l'onglé « affaires ».

Ensuite : on clique sur l'élément : « expert quickreport » et on clique sur **ok**.

Ensuite : on clique sur le bouton **lancer l'expert** et on choisit l'alias « commerce » et la table « produit ».

Ensuite : clique sur le bouton **>>** qui fait passer tous les champs à droite pour les imprimer tous et clique sur **Terminer**.

Ensuite : on clique sur cette nouvelle fiche en évitant le clic sur le composant « quickrep1 » et dans sa propriété « name » on donne le nom de cette fiche qui sera « fiche_etat_produit ».

Ensuite : on clique sur : (menu Delphi) fichier ⇒ enregistrer, et dans le dossier : « c:\commerce\source » on nomme l'unité : etat_produit.

8- on tape **Shift + F12** pour afficher la liste des fiches du logiciel et on choisit la fiche « fiche_produit » et on clique sur **ok**.

Ensuite : et pour une impression bien présentée des produits on insère un composant popupmenu1 (palette standard).

Ensuite : on double clique sur ce composant et on introduit les deux éléments «previsualiser» et «imprimer» dans leurs propriétés «caption».

Ensuite : on ferme la petite fenêtre du « popupmenu1 » et on clique sur le composant « speedbutton1 » qui représente l'impression et dans son inspecteur d'objet on choisit dans sa propriété « popupmenu » l'élément « popupmenu1 ».

Ensuite : pour programmer le composant popupmenu1 on double clique sur ce dernier et on double clique sur l'élément « previsualiser » et on écrit dans sa procédure : **fiche_etat_produit.quickrep1.preview ;**

Ensuite : on tape **F12** pour retourner à la fiche et on double clique sur le composant « popupmenu1 » et on double clique sur l'élément « imprimer » et on écrit dans sa procédure : **fiche_etat_produit.quickrep1.print ;**

9- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément produit (du sous menu fichier)

Ensuite : on écrit dans la procédure : **fiche_produit.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_produit et on doit cliquer sur **oui** pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_produit et la fiche_etat_produit et on doit cliquer sur **oui** aussi pour cette demande de confirmation.

Remarques :

1- Pour que la fiche produit se place au milieu de l'ecran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_produit » et on clique sur **Ok**.

- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

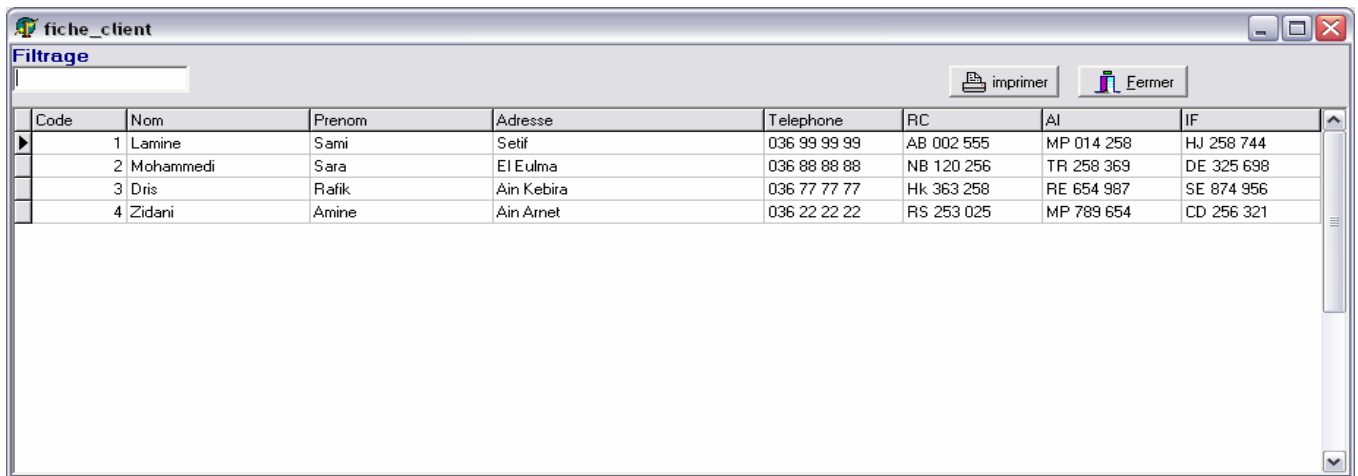
Remarque importante :

Dans le cas des champs de type monétaire (prix_achat, prix_vente, total ...), on peut modifier le symbole monétaire (La monnaie : Da, \$, £, € ...) et cela en suivant les étapes :

- On lance la fenêtre « panneau de configuration » de Windows par :
Le bouton « démarrer » ⇒ paramètre ⇒ « panneau de configuration »
- dans cette fenêtre on double clique sur : « options régionales et linguistiques »
- clique sur le bouton « personnaliser »
- clique en haut sur la page : « symbole monétaire »
- choisir ou écrire dans la liste « symbole monétaire » le symbole voulu (généralement Da)
- (Si le symbole Da ne figure pas dans la liste, donc il faut l'écrire)
- Enfin clique sur **Ok** ensuite **Ok**

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

LA FICHE CLIENT



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_client ».

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité : « client »

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Commerce
		Tablename	client
		Active	True
Accesbdb	Datasource1	Dataset	Table1
Contrôlebdb	Dbgrid1	Datasource	Datasource1
Supplement	Speedbutton1	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose
Standard	Label1	Caption	Filtrage
	Edit1	text	

3- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange):

```

if edit1.text <> '' then
begin
table1.close ;
table1.filtered :=true ;
table1.filter :='[nom_c]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;

```

4- On tape la touche **F12**, et pour faire le trie de la table client : clique sur le composant dbgrid1.
 Ensuite : dans son inspecteur d'objet on clique sur l'onglet « événements »
 Ensuite : on double clique à droite de l'événement « onclick » et on introduit le code source suivant

```

If column.index = 0 then
Table1.indexname := 'c1' ;
If column.index = 1 then
Table1.indexname := 'c2' ;
If column.index = 2 then
Table1.indexname := 'c3' ;
If column.index = 3 then
Table1.indexname := 'c4' ;
If column.index = 4 then
Table1.indexname := 'c5' ;
If column.index = 5 then
Table1.indexname := 'c6' ;

```

Remarque:

Pour insérer des index secondaires dans les champs de la table client il faut que cette table ne soit pas en cours d'utilisation donc sa propriété : active ⇒ false.

Ensuite : on lance le BDE Delphi par : (le menu Delphi) : outils ⇒ module base de données.

Ensuite : on ouvre la table par : (le menu) fichier ⇒ ouvrir ⇒ table et on choisit l'alias « commerce » et la table « client ».

Ensuite : on clique dans le menu du BDE sur table ⇒ restructurer.

Ensuite : on choisit dans la liste des choix « propriétés de la table » l'option « index secondaires »

Ensuite : on clique pour chaque champ sur le bouton **Définir** et on fait passer ce champ à droite, on clique sur le bouton **OK** et on propose un nom symbolique à cet index secondaire (c1 pour le champ code_c, c2 pour le champ nom_c,...etc.)

Ensuite : on clique sur le bouton **enregistrer** et on retourne à l'environnement Delphi.

Enfin : on réactive le composant table1 par sa propriété : active ⇒ true.

5- On doit maintenant préparer l'impression de la liste des clients comme suite :

On choisit dans le menu Delphi : fichier ⇒ nouveau, et on clique dans la fenêtre qui s'affiche sur l'onglet « affaires ».

Ensuite : on clique sur l'élément : « expert quickreport » et on clique sur « ok ».

Ensuite : on clique sur le bouton **lancer l'expert**, et on choisit l'alias « commerce » et la table « client ».

Ensuite : on clique sur le bouton **>>** qui fait passer tous les champs à droite et on clique sur **Terminer**.

Ensuite : on clique sur cette nouvelle fiche en évitant le clic sur le composant « quickrep1 » et dans sa propriété « name » on donne le nom « fiche_etat_client » à cette fiche.

Ensuite : on clique sur : (menu Delphi) fichier ⇒ enregistrer, et dans le dossier : c:\commerce\source on nomme l'unité : etat_client.

6- On tape **Shift + F12** pour afficher la liste des fiches et on choisit la fiche « fiche_client » et on clique sur **OK**.

Ensuite : et pour une impression bien présentée des clients on insère un composant popupmenu (palette standard).

Ensuite : on double clique sur ce composant et on introduit les deux éléments «previsualiser» et «imprimer» dans leurs propriétés «caption».

Ensuite : on ferme la petite fenêtre du « popupmenu » et on clique sur le composant « speedbutton1 » qui représente l'impression et dans son inspecteur d'objet on choisit dans sa propriété « popupmenu » l'élément « popupmenu1 ».

Ensuite : pour programmer le composant popupmenu1 on double clique sur ce dernier et on double clique sur l'élément « previsualiser » et on écrit dans sa procédure : **fiche_etat_client.quickrep1.preview ;**

Ensuite : on tape **F12** pour retourner à la fiche et on double clique sur le composant « popupmenu1 » et on double clique sur l'élément « imprimer » et on écrit dans sa procédure : **fiche_etat_client.quickrep1.print ;**

7- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément client (du sous menu fichier)

Ensuite : on écrit dans la procédure : **fiche_client.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_client et on doit cliquer sur **Oui** pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_client et la fiche_etat_client et on doit cliquer sur **Oui** aussi pour cette demande de confirmation.

Remarques :

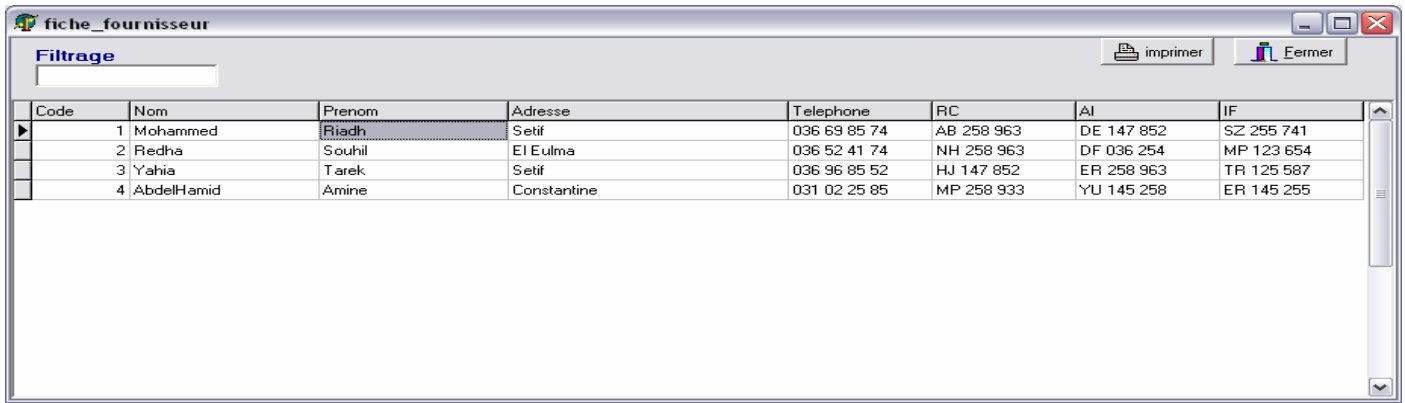
1- Pour que la fiche client se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_client » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

LA FICHE FOURNISSEUR



1- Choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_fournisseur ».

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité : « fournisseur »

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Commerce
		Tablename	fournisseur
		Active	True
Accesbd	Datasource1	Dataset	Table1
Contrôlebd	Dbgrid1	Datasource	Datasource1
Supplement	Speedbutton1	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
Standard	Bitbtn1	Kind	Bkclose
	Label1	Caption	Filtrage
	Edit1	text	

3- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange):

```

if edit1.text <> '' then
begin
table1.close ;
table1.filtered :=true ;
table1.filter :='[nom_f]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;

```

4- On tape la touche **F12** (retour à la fiche), et pour faire le trie de la table fournisseur : clique sur le composant dbgrid1.

Ensuite : dans son inspecteur d'objet on clique sur l'onglés « événements »

Ensuite : on double clique à droite de l'événement « ontitleclick » et on introduit le code source suivant (Une table est triée automatiquement selon sont index actuel (primaire ou secondaire) et on suppose que les index secondaires existes déjà pour chaque champ) :

```

If column.index = 0 then
Table1.indexname := 'c1' ;
If column.index = 1 then
Table1.indexname := 'c2' ;
If column.index = 2 then
Table1.indexname := 'c3' ;
If column.index = 3 then
Table1.indexname := 'c4' ;
If column.index = 4 then
Table1.indexname := 'c5' ;
If column.index = 5 then
Table1.indexname := 'c6' ;

```

Remarque:

Pour insérer des index secondaires dans les champs de la table fournisseur il faut que cette table ne soit pas en cours d'utilisation donc sa propriété : active ⇒ false.

Ensuite : on lance le BDE Delphi par : (le menu Delphi) : outils ⇒ module base de données.

Ensuite : on ouvre la table par : (le menu) fichier ⇒ ouvrir ⇒ table et on choisit l'alias « commerce » et la table « fournisseur ».

Ensuite : on clique dans le menu du BDE sur table ⇒ restructurer.

Ensuite : on choisit dans la liste des choix « propriétés de la table » l'option « index secondaires »

Ensuite : on clique pour chaque champ sur le bouton **Définir** et on fait passer ce champ à droite, on clique sur le bouton **OK** et on propose un nom symbolique à cet index secondaire (c1 pour le champ code_f, c2 pour le champ nom_f,...etc.)

Ensuite : on clique sur le bouton **enregistrer** et on retourne à l'environnement Delphi.

Enfin : on réactive le composant table1 par sa propriété : active ⇒ true.

5- On doit maintenant préparer l'impression de la liste des fournisseurs comme suite :

On choisit dans le menu Delphi : fichier ⇒ nouveau, et on clique dans la fenêtre qui s'affiche sur l'onglet « affaires ».

Ensuite : on clique sur l'élément : « expert quickreport » et on clique sur **OK**.

Ensuite : on clique sur le bouton **lancer l'expert**, et on choisit l'alias « commerce » et la table « fournisseur ».

Ensuite : on clique sur le bouton **>>** qui fait passer tous les champs à droite et on clique sur **Terminer**.

Ensuite : on clique sur cette nouvelle fiche en évitant le clic sur le composant « quickrep1 » et dans sa propriété « name » on donne le nom « fiche_etat_fournisseur » à cette fiche.

Ensuite : on clique sur : (menu Delphi) fichier ⇒ enregistrer, et dans le dossier : c:\commerce\source on nomme l'unité : etat_fournisseur.

6- on tape **Shift+F12** pour afficher la liste des fiches et on choisit la fiche « fiche_fournisseur » et on clique sur **OK**.

Ensuite : et pour une impression bien présentée des fournisseurs on insère un composant popupmenu (palette standard).

Ensuite : on double clique sur ce composant et on introduit les deux éléments « previsualiser » et « imprimer » dans leurs propriétés « caption ».

Ensuite : on ferme la petite fenêtre du « popupmenu » et on clique sur le composant « speedbutton1 » qui représente l'impression et dans son inspecteur d'objet on choisit dans sa propriété « popupmenu » l'élément « popupmenu1 ».

Ensuite : pour programmer le composant popupmenu1 on double clique sur ce dernier et on double clique sur l'élément « previsualiser » et on écrit dans sa procédure : fiche_etat_fournisseur.quickrep1.preview ;

Ensuite : on tape **F12** pour retourner à la fiche et on double clique sur le composant « popupmenu1 » et on double clique sur l'élément « imprimer » et on écrit dans sa procédure : fiche_etat_fournisseur.quickrep1.print ;

7- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : Shift + **F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément fournisseur (du sous menu fichier)

Ensuite : on écrit dans la procédure : fiche_fournisseur.showmodal ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_fournisseur et on doit cliquer sur **Oui** pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_fournisseur et la fiche_etat_fournisseur et on doit cliquer sur **Oui** aussi pour cette demande de confirmation.

Remarques :

1- Pour que la fiche fournisseur se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_fournisseur » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

FICHE FACTURE ACHAT

1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_facture_achat ».

Ensuite : choisir dans le menu Delphi: fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité: «facture_achat»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Commerce
		Tablename	Facture_achat
		Active	True
	Table2	Databasename	Commerce
		Tablename	Fournisseur
		Active	True
	Table3	Databasename	Commerce
		Tablename	ligne_achat
		Active	True
	Table4	Databasename	Commerce
		Tablename	Produit
		Active	True
Accesbd	Datasource1	Dataset	Table1
	Datasource2	Dataset	Table2
	Datasource3	Dataset	Table3
controlebd	Dbgrid1	Datasource	Datasource3
Supplement	Speedbutton1	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose
BDE	Query1	databasename	Commerce
		Sql	Select sum(qte_achat*prix_achat)as HT,Sum(qte_achat*prix_achat*tva)as total_tva, Sum (qte_achat*prix_achat*(1+tva)) as TTC from produit, ligne_achat Where (reference = ref_prod) and (num_achat = :p)
		Params	Clique sur « p » et choisir dans la propriété datatype : « finteger »
		Active	True
controlebd	dbnavigator	datasource	Datasource1

3- Double clique sur le composant table1.

Ensuite : clique droit sur la petite fenêtre et clique dans le menu contextuel sur: «ajouter tous les champs».

Ensuite : glisser les champs de la table facture_achat un par un vers le haut de la fiche, à la fin fermer la petite fenêtre.

Ensuite : supprimer le composant « dbedit3 » qui représente le champ « règlement » et le remplacer par le composant « dbcombobox1 » de la palette « controlebd » et modifier les propriétés de ce « dbcombobox1 » comme suite :

Datasource ⇒ datasource1

Datafield ⇒ règlement

Items ⇒ clique sur le petit bouton ⇒ écrire : cheque, espece, credit (chaque élément dans une ligne) ⇒ fermer la fenêtre.

Ensuite : supprimer le composant « dbedit4 » qui représente le champ « code_four » et le remplacer par le composant « dblookupcombobox1 » (palette « controlebd ») et modifier les propriétés de ce « dblookupcombobox1 » comme suite :

Datasource ⇒ datasource1

Datafield ⇒ code_four

Listsourc ⇒ datasource2

Listfield ⇒ code_f; nom_f

Keyfield ⇒ code_f

4- Pour mettre le curseur du clavier automatiquement dans le « dbedit1 » pour faciliter la saisie du numéro de la facture on clique sur le composant « table1 » et dans son inspecteur d'objet double clique à droite de son événement « afterinsert » on écrit dans sa procédure :

```
dbedit1.setfocus ;
- ensuite on peut contrôler la manipulation du clavier par :
- clique sur le composant dbedit1.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
Dbedit2.setfocus ;
- tapez F12 et clique sur le composant dbedit2.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
DbComboBox1.setfocus ;
- tapez F12 et clique sur le composant dbcombobox1.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
DblookupComboBox1.setfocus ;
```

5- Pour masquer le format d'affichage de la « date_achat » :

- on double clique sur le composant « table1 »
- dans la petite fenêtre on clique sur le champ « date_achat »
- on clique sur le petit bouton [...] de sa propriété « editmask »
- on clique dans la liste à droite de la fenêtre sur l'élément « date longue1 » et on clique sur
- on ferme la petite fenêtre et la date_achat sera masquée par le format « __/__/__ »

6- pour faire une liaison maître-detail entre la table maître « facture_achat », et la table détaillée « ligne_achat » : On clique sur le composant table3 « ligne_achat », et choisir dans sa propriété : mastersource → datasource1.

Ensuite : dans sa propriété « masterfield » on clique sur petit bouton.

Ensuite : on choisit dans la liste à gauche le champ « num_achat » et à droite le champ « numero_a ».

Ensuite : clique sur le bouton « ajouter » ensuite sur le bouton .

7- on peut renforcer cette fenêtre par les champs calculés comme suite :

On double clique sur le composant table3 « ligne_achat ».

Ensuite : on clique droit sur la petite fenêtre et on clique sur « ajouter tous les champs ».

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « designation » et on choisit le type du champ : « string » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « prix_achat » et on choisit le type du champ : « currency » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « tva » et on choisit le type du champ : « float » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « total » et on choisit le type du champ : « currency » et on clique sur .

A la fin : on ferme la petite fenêtre.

Ensuite : on clique sur le composant « table3 » (ligne_achat) et dans son inspecteur d'objet on clique sur l'onglet événement

Ensuite : on double clique à droite de l'événement « oncalcfield » et on écrit dans la procédure :

```
Table4.open ;
Table4.setkey ; // préparation de la recherche du produit qui a la reference tapée
Table4.fieldbyname('reference').asinteger := Table3.fieldbyname('ref_prod').asinteger ;
If table4.gotokey then // si le produit existe, on introduit sa designation, prix, tva, et on calcul le total
Begin
Table3.fieldbyname('designation').asString:= Table4.fieldbyname('designation').asString;
Table3.fieldbyname('prix_achat').ascurrency:= Table4.fieldbyname('prix_achat').ascurrency ;
Table3.fieldbyname('tva').asfloat:= Table4.fieldbyname('tva').asfloat;
Table3.fieldbyname('total').ascurrency:=Table3.fieldbyname('qte_achat').asfloat*
Table4.fieldbyname('prix_achat').ascurrency;
End;
// On profite l'occasion pour passer une valeur au paramètre à notre requête « query1 »
Query1.close;
Query1.parambyname('p').asinteger :=table1.fieldbyname('numero_a').asinteger ;
Query1.active:=true;
```

8- on tape **F12** pour retourner à la fiche et on double clique sur le composant « query1 ».

Ensuite : clique droit sur la petite fenêtre et clique dans le menu contextuel sur : « ajouter tous les champs ».

Ensuite : glisser les champs de la requête facture un par un vers le bas de la fiche, à la fin fermer la petite fenêtre.

9- pour que la facture affiche son montant ttc en lettres on procède comme suite :

On insère un composant « Button1 » (palette standard) et dans sa propriété : « caption » on écrit : conversion.
 Ensuite : on insère un composant Label (palette standard) et supprimer le contenu de sa propriété « caption »
 Ensuite : on double clique sur le composant « Bouton1 » et on écrit dans sa procédure : (Supprimer son **begin** et **end** et coller tous le code source suivant)

```
var chaine1,chaine:string;
p,nombre,q,DEC:integer;
N,n1:real;
procedure decimale(var p3:integer);
begin
if p3=10 then chaine:=chaine+ ' Dix';
if p3=11 then chaine:=chaine+ ' Onze';
if p3=12 then chaine:=chaine+ ' Douze';
if p3=13 then chaine:=chaine+ ' Treize';
if p3=14 then chaine:=chaine+ ' Quatorze';
if p3=15 then chaine:=chaine+ ' Quinze';
if p3=16 then chaine:=chaine+ ' Seize';
if p3=17 then chaine:=chaine+ ' Dix Sept';
if p3=18 then chaine:=chaine+ ' Dix Huit';
if p3=19 then chaine:=chaine+ ' Dix Neuf';
end;
procedure chiffre(var p2:integer);
begin
if p2=1 then chaine:=chaine+ ' Un';
if p2=2 then chaine:=chaine+ ' Deux';
if p2=3 then chaine:=chaine+ ' Trois';
if p2=4 then chaine:=chaine+ ' Quatre';
if p2=5 then chaine:=chaine+ ' Cinq';
if p2=6 then chaine:=chaine+ ' Six';
if p2=7 then chaine:=chaine+ ' Sept';
if p2=8 then chaine:=chaine+ ' Huit';
if p2=9 then chaine:=chaine+ ' Neuf';
end;
procedure mille(var p1:integer);
var c:integer;
begin
c:=p1 div 100;
if c=1 then chaine:=chaine+ ' Cent';
if c=2 then chaine:=chaine+ ' Deux Cent';
if c=3 then chaine:=chaine+ ' Trois Cent';
if c=4 then chaine:=chaine+ ' Quatre Cent';
if c=5 then chaine:=chaine+ ' Cinq Cent';
if c=6 then chaine:=chaine+ ' Six Cent';
if c=7 then chaine:=chaine+ ' Sept Cent';
if c=8 then chaine:=chaine+ ' Huit Cent';
if c=9 then chaine:=chaine+ ' Neuf Cent';
p1:=p1-c*100; c:=p1 div 10;
if c=2 then begin chaine:=chaine+ ' Vingt';
p1:=p1-c*10; chiffre(p1);
end;
if c=3 then begin chaine:=chaine+ ' Trente';
p1:=p1-c*10;
chiffre(p1);end;
if c=4 then begin chaine:=chaine+ ' Quarante';
p1:=p1-c*10;
chiffre(p1);end;
if c=5 then begin chaine:=chaine+ ' Cinquante';
p1:=p1-c*10;
chiffre(p1);end;
if c=6 then begin chaine:=chaine+ ' Soixante';
p1:=p1-c*10;
chiffre(p1);end;
if c=8 then begin chaine:=chaine+ ' Quatre Vingt';
p1:=p1-c*10;
chiffre(p1);end;
```

```
if c=0 then chiffre(p1);
if c=1 then decimale(p1);
if c=7 then
begin
chaine:=chaine+ ' Soixante'; p1:=p1-60;
decimale(p1);
end;
if c=9 then
begin
chaine:=chaine+ ' Quatre Vingt'; p1:=p1-80;
decimale(p1);
end; end;

{Programme principal}
var abc:real;
begin
chaine:=' ';
n:=query1.fieldbyname('ttc').asfloat;
nombre:=trunc(n);
abc:=int(n*100);
dec:=trunc(abc) mod 100;
if nombre>=1000000000 then
begin
p:=nombre div 1000000000;
nombre:=nombre-p*1000000000;
if p<>0 then
begin
mille(p); chaine:=chaine+ ' Milliard';
end; end;
{milliards}
if nombre>=1000000 then
begin
p:=nombre div 1000000;
nombre:=nombre-p*1000000;
if p<>0 then
begin
mille(p); chaine:=chaine+ ' Million';
end; end;
{milliers}
if nombre>=1000 then
begin
p:=nombre div 1000; nombre:=nombre-p*1000;
if p<>1 then
begin
mille(p); chaine:=chaine+ ' Mille';
end else
chaine:=chaine+ 'mille';
end;
{simple}
if (0<=nombre) and (nombre<1000) then
begin
p:=nombre; mille(p);
end;
chaine:=chaine+ ' dinars' + ' et';
begin q:=dec; mille(q);
IF q=0 THEN chaine:=chaine+ ' Zero CTS'
Else chaine:=chaine+ ' CTS'
end;
{chaine:=chaine+ ' dinars' + ' et'+chaine1+' CTS';}
label8.caption:=chaine; end;
```

10- le « dbgrid1 » affiche le champ « num_achat » (de la table ligne_achat) qui contient bien sûr le même numéro de la facture_achat actuelle (puisque il y'a une liaison maître_détail entre la table ligne_achat et la table facture_achat par leurs champs communs num_achat et numero_a), donc ce champ répètera pour tous les enregistrements du « dbgrid1 » le même numéro de facture, il vaut mieux alors de ne pas afficher ce dernier dans le « dbgrid1 ». pour supprimer ce champ :

- double clic sur le « dbgrid1 ».
- clic en haut de la petite fenêtre sur le bouton « ajouter tous les champs »
- supprimer le champ « num_achat » et fermer la petite fenêtre.

(Le champ « num_achat » ne sera pas supprimé de la table « ligne_achat » mais seulement il ne sera pas affiché)

11- pour préparer l'impression de notre facture_achat on procède comme suite :

On insère une nouvelle fiche à notre logiciel par le choix dans le menu Delphi de : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom: « fiche_etat_facture_achat ».

Ensuite : (le menu Delphi): fichier⇒enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité: «etat_facture_achat».

Ensuite : on insère dans cette fiche un composant « quickrep1 » de la palette « qreport ».

Ensuite : on tape **F12** et en haut de l'unité on écrit dans la liste «uses» le nom «facture_achat» et on compile **Ctrl+F9** pour assurer la liaison entre la fiche_facture_achat et la fiche_etat_facture_achat.

Nb : par fois Delphi répète quelques unités de la partie uses si on écrit une nouvelle unité dans cette partie, donc on doit supprimer les unités dupliquées si Delphi signal ce problème de répétition après la compilation du logiciel.

Ensuite : dans l'inspecteur d'objet du quickrep1 on écrit dans la propriété : zoom ⇒ 70

Ensuite : dans l'inspecteur d'objet du quickrep1 on choisit dans la propriété : dataset ⇒ « fiche_facture_achat.table3 »

Ensuite: choisir dans sa propriété Bands: hastitle⇒true; hascolumnheader⇒true; hasdetail⇒true; hassummary⇒ true.

Ensuite : on insère les composants suivants de la palette « qreport » sur les bandes de ce quickrep1 comme suite:

bande	Composant	Propriété	valeur
Titre (agrandir la hauteur de cette bande)	qrlabel1	caption	société de commerce multiple
	qrlabel2	caption	Facture achat
	qrlabel3	caption	numero
	qrdtext1	dataset	fiche_facture_achat.table1
		datafield	Numero_a
	Qrlabel4	caption	Date achat
	Qrdtext2	dataset	fiche_facture_achat.table1
		datafield	Date_achat
	Qrlabel5	caption	reglement
	Qrdtext3	dataset	fiche_facture_achat.table1
		datafield	Reglement_a
	Qrlabel6	caption	fournisseur
	Qrdtext4	dataset	fiche_facture_achat.table2
		datafield	Nom_f
Entête de colonnes	Qrlabel7	caption	Reference
	Qrlabel8	caption	Designation
	Qrlabel9	caption	Qte achat
	Qrlabel910	caption	Prix unite
	Qrlabel11	caption	Tva
	Qrlabel12	caption	total

Detail	Qrdbtext5	dataset	fiche_facture_achat.table3
		datafield	reference
	Qrdbtext6	dataset	fiche_facture_achat.table3
		datafield	designation
	Qrdbtext7	dataset	fiche_facture_achat.table3
		datafield	Qte_achat
	Qrdbtext8	dataset	fiche_facture_achat.table3
		datafield	Prix_achat
	Qrdbtext9	dataset	fiche_facture_achat.table3
		datafield	tva
Resumé (agrandir la hauteur de cette bande)	Qrdbtext10	dataset	fiche_facture_achat.table3
		datafield	total
	Qrlabel13	caption	HT
	Qrdbtext11	dataset	fiche_facture_achat.query1
		datafield	HT
	Qrlabel14	caption	TOTAL TVA
	Qrdbtext12	dataset	fiche_facture_achat.query1
		datafield	TOTAL TVA
	Qrlabel15	caption	TTC
	Qrdbtext13	dataset	fiche_facture_achat.query1
		datafield	TTC
	Qrlabel16	caption	Arrêtez la présente facture à la somme de
	Qrlabel17	caption	

Remarque :

On peut utiliser les composants « Qrshape » de la palette « Qreport » pour encadrer les champs des tables (composants qrdbtext) et leurs entêtes de colonnes (composants « qrlabel ») pour avoir ensuite des tables encadrées dans notre état d'impression. Les étapes sont :

- insérer dans la bande entête de colonnes des composants « Qrshape » pour chaque « qrlabel » de cette bande pour les encadrer.
- Le « qrshape » au début cache le « qrlabel » qui va l'encadrer, donc on clique sur chaque « qrshape » par le bouton droit de la souris et on choisit dans le menu contextuel « mettre en arrière plan ».
- Pour bien contrôler la position des « qrshape » on utilise les touches : **Ctrl** +  ou les autre touches de flèches.
- Pour bien contrôler la taille des « qrshape » on utilise les touches : **Shift** +  ou les autre touches de flèches.
- on presse sur la touche **Shift** et par souris on clique sur tous les « qrshape » de la bande « entête de colonnes » et en fin on lache la touche **Shift** et on tape les touches **Ctrl** + **C** pour copier ces cadres.
- On clique sur la bande « detail » et on tape **Ctrl** + **V** pour coller les cadres dans cette bande.
- On déplace ces cadres par les touches **Ctrl** +  et on clique sur ces cadres par le bouton droit de la souris et on clique sur « mettre en arrière plan ».

Ensuite : on tape **Shift** + **F12** et on affiche la fiche « fiche_facture_achat »

Ensuite : et pour une impression bien présentée de la facture on insère un composant popupmenu (palette standard).

Ensuite : on double clique sur ce composant et on introduit les deux éléments «previsualiser» et «imprimer» dans leurs propriétés «caption».

Ensuite : on ferme la petite fenêtre du « popupmenu » et on clique sur le composant « speedbutton1 » qui représente l'impression et dans son inspecteur d'objet on choisit dans sa propriété « popupmenu » l'élément « popupmenu1 ».

Ensuite : pour programmer le composant popupmenu1 on double clique sur ce dernier et on double clique sur l'élément « previsualiser » et on écrit dans sa procédure :

Fiche_etat_facture_achat.qrlabel17.caption := label8.caption ; // qrlabel17 reçoit le TTC en lettres.

Fiche_etat_facture_achat.quickrep1.preview ;

Ensuit : on tape **F12** pour retourner à la fiche et on double clique sur le composant « popupmenu1 » et on double clique sur l'élément « imprimer » et on écrit dans sa procédure :

Fiche_etat_facture_achat.qrlabel17.caption := label8.caption ;

Fiche_etat_facture_achat.quickrep1.print ;

12- lorsqu'on enregistre les modifications dans notre table1 (facture_achat) par le bouton « enregistrer » du dbnavigator1, on doit ajouter les quantités_achat à la quantité du stock. On programme cela par le choix du composant table1 et dans la procédure de son événement « afterpost » on écrit :

```

table3.first ;
while not(table3.eof) do
begin
Table4.setkey ; // préparation de la recherche
Table4.fieldbyname('reference').asinteger := Table3.fieldbyname('ref_prod').asinteger ;
If table4.gotokey then
    Begin
    Table4.edit; //préparation de la modification
    Table4.fieldbyname('qte_stock').asfloat:=Table4.fieldbyname('qte_stock').asfloat+
    Table3.fieldbyname('qte_achat').asfloat;
    Table4.post;
    End;
Table3.next;
End;

```

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

13- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément facture achat (du sous menu Edition)

Ensuite : on écrit dans la procédure : fiche_facture_achat.showmodal ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_facture_achat et on doit cliquer sur **oui** pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_facture_achat et la fiche_etat_facture_achat et on doit cliquer sur **oui** aussi pour cette demande de confirmation.

Remarques :

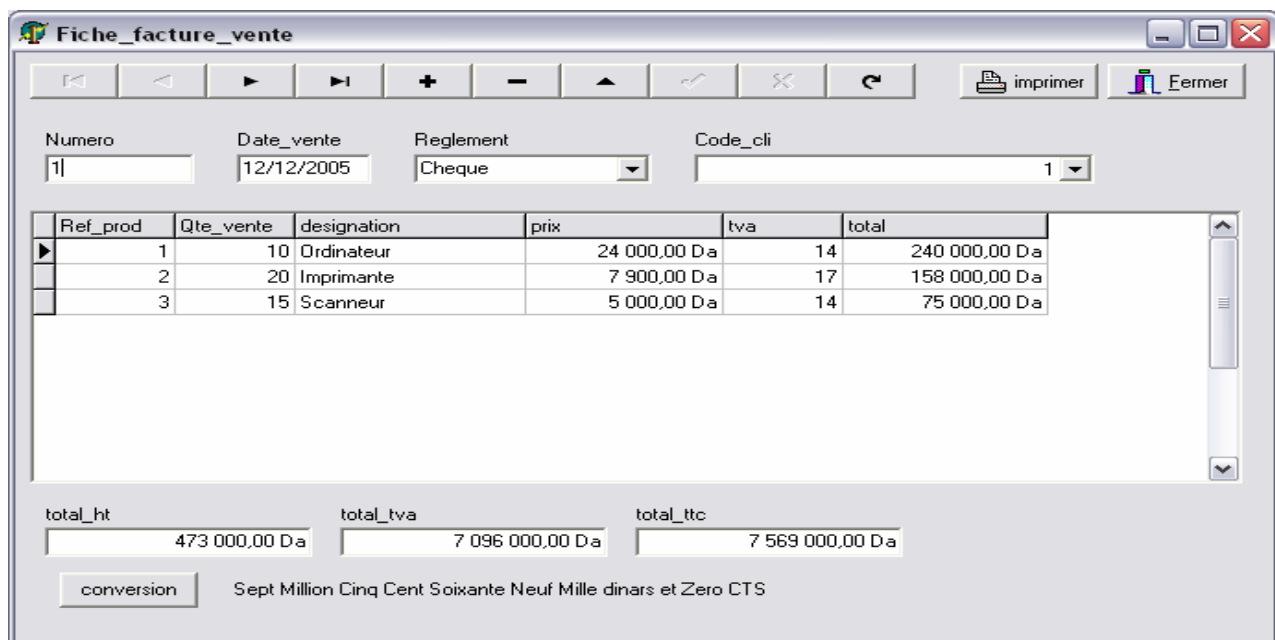
1- Pour que la fiche facture achat se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_facture_achat » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

- ❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

FICHE FACTURE VENTE



Ref_prod	Qte_vente	designation	prix	tva	total
1	10	Ordinateur	24 000,00 Da	14	240 000,00 Da
2	20	Imprimante	7 900,00 Da	17	158 000,00 Da
3	15	Scanneur	5 000,00 Da	14	75 000,00 Da

total_ht: 473 000,00 Da total_tva: 7 096 000,00 Da total_ttc: 7 569 000,00 Da

conversion Sept Million Cinq Cent Soixante Neuf Mille dinars et Zero CTS

1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_facture_vente».

Ensuite : choisir dans le menu Delphi: fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité: «facture_vente»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Commerce
		Tablename	Facture_vente
		Active	True
	Table2	Databasename	Commerce
		Tablename	client
		Active	True
	Table3	Databasename	Commerce
		Tablename	Ligne_vente
		Active	True
	Table4	Databasename	Commerce
		Tablename	Produit
		Active	True
Accesbd	Datasource1	Dataset	Table1
	Datasource2	Dataset	Table2
	Datasource3	Dataset	Table3
controlebd	Dbgrid1	Datasource	Datasource3
Supplement	Speedbutton1	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose
BDE	Query1	databasename	Commerce
		Sql	Select sum(qte_vente*prix_vente)as HT,Sum(qte_vente *prix_vente*tva)as total_tva, Sum (qte_vente*prix_vente*(1+tva)) as TTC from produit, ligne_vente Where (reference = ref_prod) and (num_vente = :p)
		Params	Clique sur « p » et choisir dans la propriété datatype : « ftinteger »
		Active	True
controlebd	dbnavigator	datasource	Datasource1

3- Double clique sur le composant table1.

Ensuite : clique droit sur la petite fenêtre et clique dans le menu contextuel sur: « ajouter tous les champs ».

Ensuite : glisser les champs de la table facture_vente un par vers le haut de la fiche, à la fin fermer la petite fenêtre.

Ensuite : supprimer le composant « dbedit3 » qui représente le champ « règlement » et le remplacer par le composant « dbcombobox1 » de la palette « controlebd » et modifier les propriétés de ce « dbcombobox1 » comme suite :

Datasource ⇒ datasource1

Datafield ⇒ règlement

Items ⇒ clique sur le petit bouton [...] et écrire : cheque, espece, credit (chacun dans une ligne) et fermer la fenêtre.

Ensuite : supprimer le composant « dbedit4 » qui représente le champ « code_cli » et le remplacer par le composant « dblookupcombobox1 » (palette « controlebd ») et modifier les propriétés de ce « dblookupcombobox1 » comme suite :

Datasource ⇒ datasource1

Datafield ⇒ code_cli

Listsourc ⇒ datasource2

Listfield ⇒ code_c; nom_c

Keyfield ⇒ code_c

4- Pour mettre le curseur du clavier automatiquement dans le « dbedit1 » pour faciliter la saisie du numéro de la facture on clique sur le composant « table1 » et dans son inspecteur d'objet double clique à droite de son événement : « afterinsert » on écrit dans sa procédure :

dbedit1.setfocus ;

- ensuite on peut contrôler la manipulation du clavier par :

- clique sur le composant dbedit1.

- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :

If key = vk_return then

Dbedit2.setfocus ;

- tapez **F12** et clique sur le composant dbedit2.

- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :

If key = vk_return then

DbCombobox1.setfocus ;

- tapez **F12** et clique sur le composant dbcombobox1.

- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :

If key = vk_return then

DblookupCombobox1.setfocus ;

- 5- Pour masquer le format d'affichage de la « date_vente » :
- on double clique sur le composant « table1 »
 - dans la petite fenêtre on clique sur le champ « date_vente »
 - on clique sur le petit bouton [...] de sa propriété « editmask »
 - on clique dans la liste à droite de la fenêtre sur l'élément « date longue1 » et on clique sur
 - on ferme la petite fenêtre et la date_vente sera masquée par le format « __/__/__ »

6- pour faire une liaison maître-detail entre la table maître « facture_vente », et la table détaillée « ligne_vente » :
 On clique sur le composant table3 « ligne_vente », et choisir dans sa propriété : mastersource → datasource1.
 Ensuite : dans sa propriété « masterfield » on clique sur petit bouton.
 Ensuite : on choisit dans la liste à gauche le champ « num_vente » et à droite le champ « numero_v ».
 Ensuite : clique sur le bouton « ajouter » ensuite sur le bouton .

7- on peut renforcer cette fenêtre par les champs calculés comme suite :

On double clique sur le composant table3 « ligne_vente ».

Ensuite : on clique droit sur la petite fenêtre et on clique sur « ajouter tous les champs ».

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « designation » et on choisit le type du champ : « string » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « prix_vente » et on choisit le type du champ : « currency » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « tva » et on choisit le type du champ : « float » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « total » et on choisit le type du champ : « currency » et on clique sur .

A la fin : on ferme la petite fenêtre.

Ensuite : on clique sur le composant « table3 » (ligne_vente) et dans son inspecteur d'objet on clique sur l'onglet événement

Ensuite : on double clique à droite de l'événement « oncalcfld » et on écrit dans la procédure :

```
Table4.open ;
Table4.setkey ;
Table4.fieldbyname('reference').asinteger := Table3.fieldbyname('ref_prod').asinteger ;
If table4.gotokey then
Begin
Table3.fieldbyname('designation').asString:= Table4.fieldbyname('designation').asString;
Table3.fieldbyname('prix_vente').ascurrency:= Table4.fieldbyname('prix_vente').ascurrency ;
Table3.fieldbyname('tva').asfloat:= Table4.fieldbyname('tva').asfloat;
Table3.fieldbyname('total').ascurrency:=Table3.fieldbyname('qte_vente').asfloat*
Table4.fieldbyname('prix_vente').ascurrency;
End;
// On profite l'occasion pour passer une valeur au paramètre à notre requête « query1 »
Query1.close;
Query1.parambyname('p').asinteger :=table1.fieldbyname('numero_v').asinteger ;
Query1.active:=true;
```

8- on tape **F12** pour retourner à la fiche et on double clique sur le composant « query1 ».

Ensuite : clique droit sur la petite fenêtre et clique dans le menu contextuel sur : « ajouter tous les champs ».

Ensuite : glisser les champs de la requête facture un par un vers le bas de la fiche, à la fin fermer la petite fenêtre.

9- pour que la facture affiche son montant ttc en lettres on procède comme suite :

On insère un composant « Button1 » (palette standard) et dans sa propriété : « caption » on écrit : conversion.

Ensuite : on insère un composant Label (palette standard) et supprimer le contenu de sa propriété « caption »

Ensuite : on double clique sur le composant « Bouton1 » et on écrit dans sa procédure : (Supprimer son **begin** et **end** et coller tous le code source suivant)

```

var chaine1,chaine:string;
p,nombre,q,DEC:integer;
N,n1:real;
procedure decimale(var p3:integer);
begin
if p3=10 then chaine:=chaine+ ' Dix';
if p3=11 then chaine:=chaine+ ' Onze';
if p3=12 then chaine:=chaine+ ' Douze';
if p3=13 then chaine:=chaine+ ' Treize';
if p3=14 then chaine:=chaine+ ' Quatorze';
if p3=15 then chaine:=chaine+ ' Quinze';
if p3=16 then chaine:=chaine+ ' Seize';
if p3=17 then chaine:=chaine+ ' Dix Sept';
if p3=18 then chaine:=chaine+ ' Dix Huit';
if p3=19 then chaine:=chaine+ ' Dix Neuf';
end;
procedure chiffre(var p2:integer);
begin
if p2=1 then chaine:=chaine+ ' Un';
if p2=2 then chaine:=chaine+ ' Deux';
if p2=3 then chaine:=chaine+ ' Trois';
if p2=4 then chaine:=chaine+ ' Quatre';
if p2=5 then chaine:=chaine+ ' Cinq';
if p2=6 then chaine:=chaine+ ' Six';
if p2=7 then chaine:=chaine+ ' Sept';
if p2=8 then chaine:=chaine+ ' Huit';
if p2=9 then chaine:=chaine+ ' Neuf';
end;
procedure mille(var p1:integer);
var c:integer;
begin
c:=p1 div 100;
if c=1 then chaine:=chaine+ ' Cent';
if c=2 then chaine:=chaine+ ' Deux Cent';
if c=3 then chaine:=chaine+ ' Trois Cent';
if c=4 then chaine:=chaine+ ' Quatre Cent';
if c=5 then chaine:=chaine+ ' Cinq Cent';
if c=6 then chaine:=chaine+ ' Six Cent';
if c=7 then chaine:=chaine+ ' Sept Cent';
if c=8 then chaine:=chaine+ ' Huit Cent';
if c=9 then chaine:=chaine+ ' Neuf Cent';
p1:=p1-c*100; c:=p1 div 10;
if c=2 then begin
chaine:=chaine+ ' Vingt';
p1:=p1-c*10; chiffre(p1); end;
if c=3 then begin
chaine:=chaine+ ' Trente';
p1:=p1-c*10; chiffre(p1); end;
if c=4 then begin
chaine:=chaine+ ' Quarante';
p1:=p1-c*10; chiffre(p1); end;
if c=5 then begin
chaine:=chaine+ ' Cinquante';
p1:=p1-c*10; chiffre(p1); end;
if c=6 then begin
chaine:=chaine+ ' Soixante';
p1:=p1-c*10; chiffre(p1); end;
if c=8 then begin
chaine:=chaine+ ' Quatre Vingt';
p1:=p1-c*10; chiffre(p1); end;

```

```

if c=0 then chiffre(p1);
if c=1 then decimale(p1);
if c=7 then
begin
chaine:=chaine+ ' Soixante';
p1:=p-60; decimale(p1); end;
if c=9 then
begin
chaine:=chaine+ ' Quatre Vingt';
p1:=p1-80; decimale(p1);
end; end;
{Programme principal}
var abc:real;
begin
chaine:=' ';
n:=query1.fieldbyname('ttc').asfloat;
nombre:=trunc(n);
abc:=int(n*100);
dec:=trunc(abc) mod 100;
if nombre>=1000000000 then
begin
p:=nombre div 1000000000;
nombre:=nombre-p*1000000000;
if p<>0 then
begin
mille(p); chaine:=chaine+ ' Milliard';
end; end;
{milliards}
if nombre>=1000000 then
begin
p:=nombre div 1000000;
nombre:=nombre-p*1000000;
if p<>0 then
begin
mille(p); chaine:=chaine+ ' Million';
end; end;
{milliers}
if nombre>=1000 then
begin
p:=nombre div 1000;
nombre:=nombre-p*1000;
if p<>1 then
begin
mille(p); chaine:=chaine+ ' Mille';
end else
chaine:=chaine+ 'mille';
end;
{simple}
if (0<=nombre) and (nombre<1000) then
begin
p:=nombre; mille(p);
end;
chaine:=chaine+ ' dinars' + ' et';
begin q:=dec; mille(q);
IF q=0 THEN
chaine:=chaine+ ' Zero CTS'
Else
chaine:=chaine+ ' CTS'
end;
{chaine:=chaine+ ' dinars' + ' et'+chaine1+' CTS';}
label8.caption:=chaine;
end;

```

10- le « dbgrid1 » affiche le champ « num_vente » (de la table ligne_vente) qui contient bien sûr le même numéro de la facture_vente actuelle (puisque il y'a une liaison maître_détail entre la table ligne_vente et la table facture_vente par leurs champs communs num_vente et numero_v), donc ce champ répétera pour tous les enregistrements du « dbgrid1 » le même numéro de facture, il vaut mieux alors de ne pas afficher ce dernier dans le « dbgrid1 ». pour supprimer ce champ :

- double clic sur le « dbgrid1 ».
- clic en haut de la petite fenêtre sur le bouton « ajouter tous les champs »
- supprimer le champ « num_vente » et fermer la petite fenêtre.

(Le champ « num_vente » ne sera pas supprimé de la table « ligne_vente » mais seulement il ne sera pas affiché)

11- pour préparer l'impression de notre facture_vente on procède comme suite :

On insère une nouvelle fiche à notre logiciel par le choix dans le menu Delphi de : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom: « fiche_etat_facture_vente ».

Ensuite : (le menu Delphi): fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité: «etat_facture_vente».

Ensuite : on insère dans cette fiche un composant « quickrep1 » de la palette « qreport ».

Ensuite : on tape **F12** et en haut de l'unité on écrit dans la liste «uses» le nom «facture_vente» et on compile **Ctrl+F9** pour assurer la liaison entre la fiche_facture_vente et la fiche_etat_facture_vente.

Nb : par fois Delphi répète quelques unités de la partie uses si on écrit une nouvelle unité dans cette partie, donc on doit supprimer les unités dupliquées si Delphi signal ce problème de répétition après la compilation du logiciel.

Ensuite : dans l'inspecteur d'objet du quickrep1 on écrit dans la propriété : zoom ⇒ 70

Ensuite : dans l'inspecteur d'objet du quickrep1 on choisit dans la propriété : dataset ⇒ « fiche_facture_vente.table3 »

Ensuite :choisir dans sa propriété Bands: hastitle⇒true; hascolumnheader⇒true; hasdetail⇒true; hassummary⇒ true.

Ensuite : on insère les composants suivants de la palette « qreport » sur les bandes de ce quickrep1 comme suite:

bande	Composant	Propriété	valeur
Titre (agrandir la hauteur de cette bande)	qrlabel1	caption	société de commerce multiple
	qrlabel2	caption	Facture vente
	qrlabel3	caption	numero
	qrdbtext1	dataset	fiche_facture_vente.table1
		datafield	Numero_v
	Qrlabel4	caption	Date vente
	Qrdbtext2	dataset	fiche_facture_vente.table1
		datafield	Date_vente
	Qrlabel5	caption	reglement
	Qrdbtext3	dataset	fiche_facture_vente.table1
		datafield	Reglement_v
	Qrlabel6	caption	client
	Qrdbtext4	dataset	fiche_facture_vente.table2
		datafield	Nom_c

Entête de colonnes	Qrlabel7	caption	Reference
	Qrlabel8	caption	Designation
	Qrlabel9	caption	Qte vente
	Qrlabel910	caption	Prix unite
	Qrlabel11	caption	Tva
	Qrlabel12	caption	total
Detail	Qrdbtext5	dataset	fiche_facture_vente.table3
		datafield	reference
	Qrdbtext6	dataset	fiche_facture_vente.table3
		datafield	designation
	Qrdbtext7	dataset	fiche_facture_vente.table3
		datafield	Qte vente
	Qrdbtext8	dataset	fiche_facture_vente.table3
		datafield	Prix vente
	Qrdbtext9	dataset	fiche_facture_vente.table3
		datafield	tva
Resumé (agrandir la hauteur de cette bande)	Qrdbtext10	dataset	fiche_facture_vente.table3
		datafield	total
	Qrlabel13	caption	HT
	Qrdbtext11	dataset	fiche_facture_vente.query1
		datafield	HT
	Qrlabel14	caption	TOTAL TVA
	Qrdbtext12	dataset	fiche_facture_vente.query1
		datafield	TOTAL TVA
	Qrlabel15	caption	TTC
	Qrdbtext13	dataset	fiche_facture_vente.query1
		datafield	TTC
	Qrlabel16	caption	Arrêtez la présente facture à la somme de
	Qrlabel17	caption	

Remarque :

On peut utiliser les composants « Qrshape » de la palette « Qreport » pour encadrer les champs des tables (composants qrdbtext) et leurs entêtes de colonnes (composants « qrlabel ») pour avoir ensuite des tables encadrées dans notre état d'impression.

Les étapes sont :

- insérer dans la bande entête de colonnes des composants « Qrshape » pour chaque « qrlabel » de cette bande pour les encadrer.
- Le « qrshape » au début cache le « qrlabel » qui va l'encadrer, donc on clique sur chaque « qrshape » par le bouton droit de la souris et on choisit dans le menu contextuel « mettre en arrière plan ».
- Pour bien contrôler la position des « qrshape » on utilise les touches : **Ctrl** +  ou les autre touches de flèches.
- Pour bien contrôler la taille des « qrshape » on utilise les touches : **Shift** +  ou les autre touches de flèches.
- on presse sur la touche **Shift** et par souris on clique sur tous les « qrshape » de la bande « entête de colonnes » et en fin on lache la touche **Shift** et on tape les touches **Ctrl** + **C** pour copier ces cadres.
- On clique sur la bande « détail » et on tape **Ctrl** + **V** pour coller les cadres dans cette bande.
- On déplace ces cadres par les touches **Ctrl** +  et on clique sur ces cadres par le bouton droit de la souris et on clique sur « mettre en arrière plan ».

Ensuite : on tape **Shift** + **F12** et on affiche la fiche « fiche_facture_vente »

Ensuite : et pour une impression bien présentée de la facture on insère un composant popupmenu (palette standard).

Ensuite : on double clique sur ce composant et on introduit les deux éléments «previsualiser» et «imprimer» dans leurs propriétés «caption».

Ensuite : on ferme la petite fenêtre du « popupmenu » et on clique sur le composant « speedbutton1 » qui représente l'impression et dans son inspecteur d'objet on choisit dans sa propriété « popupmenu » l'élément « popupmenu1 ».

Ensuite : pour programmer le composant popupmenu1 on double clique sur ce dernier et on double clique sur l'élément « previsualiser » et on écrit dans sa procédure :

Fiche_etat_facture_vente.qrlabel17.caption := label8.caption ;

Fiche_etat_facture_vente.quickrep1.preview ;

Ensuite : on tape **F12** pour retourner à la fiche et on double clique sur le composant « popupmenu1 » et on double clique sur l'élément « imprimer » et on écrit dans sa procédure :

Fiche_etat_facture_vente.qrlabel17.caption := label8.caption ;

Fiche_etat_facture_vente.quickrep1.print ;

12- lorsqu'on enregistre les modifications dans notre table1 (facture_vente) par le bouton « enregistrer » du dbnavigator1, on doit supprimer les quantités_vente des quantité_stock. On programme cela par le choix du composant table1 et dans la procédure de son événement « afterpost » on écrit :

table3.first ;

while not(table3.eof) do

begin

Table4.setkey ; // préparation de la recherche

Table4.fieldbyname('reference').asinteger := Table3.fieldbyname('ref_prod').asinteger ;

If table4.gotokey then

Begin

Table4.edit; //préparation de la modification

Table4.fieldbyname('qte_stock').asfloat:=Table4.fieldbyname('qte_stock').asfloat-

Table3.fieldbyname('qte_vente').asfloat;

Table4.post;

End;

Table3.next;

End;

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

13- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément facture vente (du sous menu Edition)

Ensuite : on écrit dans la procédure : **fiche_facture_vente.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_facture_vente et on doit cliquer sur **oui** pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_facture_vente et la fiche_etat_facture_vente et on doit cliquer sur **oui** aussi pour cette demande de confirmation.

Remarques :

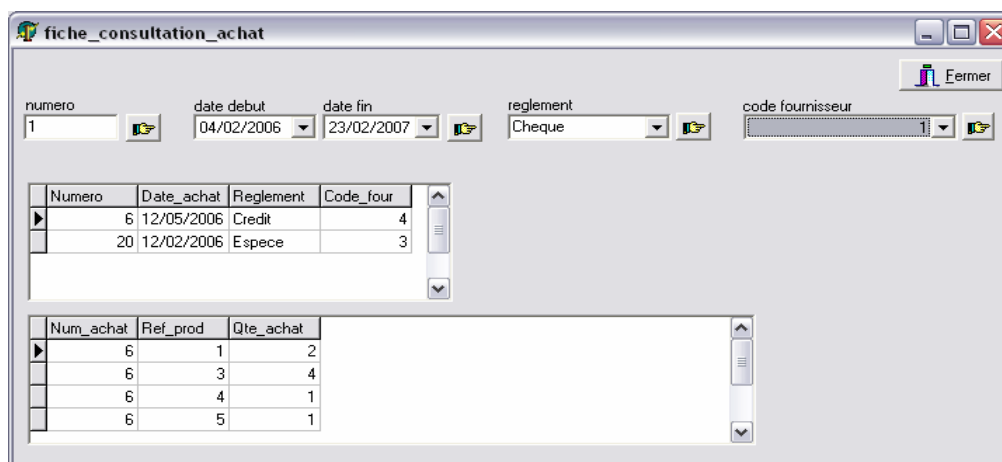
1- Pour que la fiche facture vente se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_facture_vente » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position → poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier → tout enregistrer.

- ❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

LA FICHE CONSULTATION ACHATS



Numero	Date_achat	Reglement	Code_four
6	12/05/2006	Credit	4
20	12/02/2006	Especie	3

Num_achat	Ref_prod	Qte_achat
6	1	2
6	3	4
6	4	1
6	5	1

1. choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_consultation_achat ».

Ensuite : (le menu Delphi) : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité : « consultation_achat »

2. insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	bitbtn	kind	bkclos
standard	label1	Caption	numéro facture
	edit1	text	
BDE	Query1	databasename	Commerce
		Sql	Select * from facture_achat Where numero_a = :num
		Params	Clique sur « num » et choisir dans la propriété datatype : « finteger »
		Active	True
Accèsbd	Datasource1	dataset	Query1
Supplement	Speedbutton1	glyph	(choisir une image)
standard	label2	Caption	Date début
	label3	Caption	Date fin
win32	datetimepicker1		
	datetimepicker2		
supplement	Speedbutton2	glyph	(choisir une image)
BDE	Query2	databasename	commerce
		Sql	Select * from facture_achat Where (date_achat >=:debut) and (date_achat <=:fin)
		Params	Clique sur « debut » et choisir dans la propriété datatype : « fdate » Clique sur « fin » et choisir dans la propriété datatype : « fdate »
		Active	True
standard	label4	Caption	Règlement
	combobox1	text	
		items	cheque espece credit
Supplement	Speedbutton3	glyph	(choisir une image)
BDE	Query3	databasename	commerce
		Sql	Select * from facture_achat Where reglement_a =:reg
		Params	Clique sur « reg » et choisir dans la propriété datatype : « fstring »
		Active	True
	Table1	databasename	Commerce
		tablename	Fournisseur
		Active	True
Accèsbd	Datasource2	dataset	Table1
Standard	Label1	caption	Fournisseur
controlebd	oboxdblookupcomb1	listsource	datasource2
		listfield	code_f ; nom_f
		keyfield	Code_f
BDE	Query4	databasename	Commerce
		Sql	Select * from facture_achat Where code_four =:cod
		Params	Clique sur « cod » et choisir dans la propriété datatype : « finteger »
		Active	True
Supplement	Speedbutton4	glyph	(choisir une image)
BDE	Table2	databasename	commerce
		tablename	Ligne_achat
		active	true
		mastersource	datasource1
		masterfield	Clique sur « num_achat » et clique sur « numero_a » et clique sur Ajouter et clique sur Ok
Accesbd	datasource3	dataset	table2
controlebd	dbgrid1	datasource	datasource1
	dbgrid2	datasource	datasource3

3. double clique sur le composant « speedbutton1 » et écrire :

```

Query1.close ;
Query1.params[0].asinteger :=strtoint(edit1.text);
Query1.active :=true ;
Datasource1.dataset :=Query1 ;

```

4. tapez **F12** et double clique sur le composant « speedbutton2 » et écrire:

```

Query2.close ;
Query2.params[0].asdate :=datetimepicker1.date;
Query2.params[1].asdate :=datetimepicker2.date;
Query2.active :=true ;
Datasource1.dataset :=Query2 ;

```

5. tapez **F12** et double clique sur le composant « speedbutton3 » et écrire:

```
Query3.close ;
Query3.params[0].asstring :=combobox1.text;
Query3.active :=true ;
Datasource1.dataset :=Query3 ;
```

6. tapez **F12** et double clique sur le composant « speedbutton4 » et écrire:

```
Query4.close ;
Query4.params[0].asinteger :=strtoint(dblookupcombobox1.text);
Query4.active :=true ;
Datasource1.dataset :=Query4 ;
```

7. cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément Achats (du sous menu Consultation)

Ensuite : on écrit dans la procédure : fiche_consultation_achat.showmodal ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_consultation_achat et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

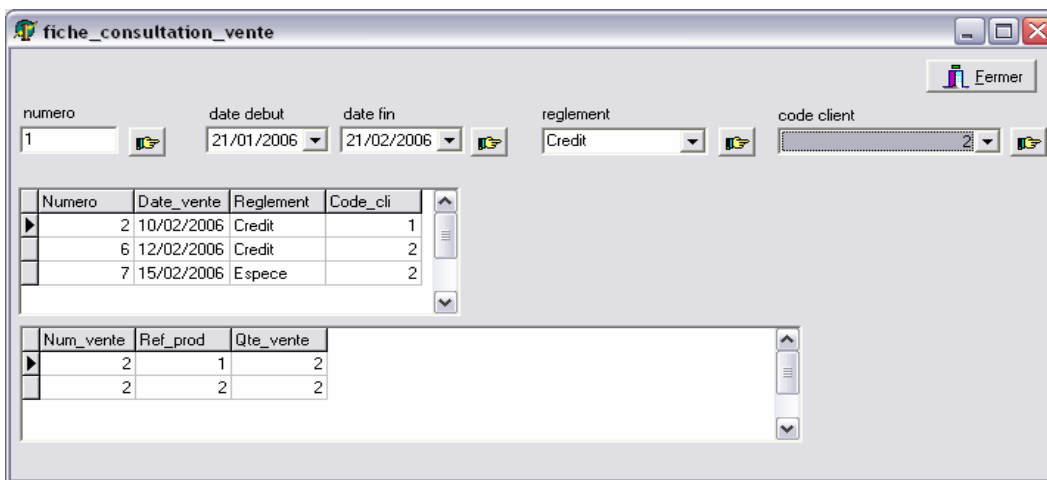
1- Pour que la fiche consultation achat se place au milieu de l'écran si on l'appel lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_consultation_achat » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

FICHE CONSULTATION VENTES



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_consultation_vente».

Ensuite : (le menu Delphi): fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité: «consultation_vente»

2- insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	bitbtn	kind	bkclos
standard	label1	Caption	numéro facture
	edit1	text	
BDE	Query1	databasename	Commerce
		Sql	Select * from facture_vente Where numero_v = :num
		Params	Clique sur « num » et choisir dans la propriété datatype : « ftinteger »
		Active	True
Accéesbd	Datasource1	dataset	Query1
Supplement	Speedbutton1	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
standard	label2	Caption	Date début
	label3	Caption	Date fin
win32	datetimepicker1		

	datetimepicker2		
supplement	speedddbutton2	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
BDE	Query2	databasename	commerce
		Sql	Select * from facture_vente Where (date_vente >=:debut) and (date_vente <=:fin)
		Params	Clique sur « debut » et choisir dans la propriété datatype : « ftdate » Clique sur « fin » et choisir dans la propriété datatype : « ftdate »
		Active	True
standard	label4	Caption	Règlement
	combobox1	text	
		items	cheque espece credit
Supplement	Speedbutton3	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
BDE	Query3	databasename	commerce
		Sql	Select * from facture_vente Where reglement_v=:reg
		Params	Clique sur « reg » et choisir dans la propriété datatype : « ftstring »
		Active	True
	Table1	databasename	Commerce
		tablename	client
		Active	True
Accèsbd	Datasource2	dataset	Table1
Standard	Label1	caption	client
controlebd	dblookupcombobox1	listesource	datasource2
		listefield	code_c;nom_c
		keyfield	code_c
BDE	Query4	databasename	Commerce
		Sql	Select * from facture_vente Where code_cli=:cod
		Params	Clique sur « cod » et choisir dans la propriété datatype : « ftinteger »
		Active	True
Supplement	Speedbutton4	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
BDE	Table2	databasename	commerce
		tablename	ligne_vente
		active	true
		mastersource	datasource1
		masterfield	Clique sur « num_vente » et clique sur « numero_v » et clique sur Ajouter et clique sur Ok
AccesBd	datasource3	dataset	table2
controlebd	dbgrid1	datasource	cedatasour1
	dbgrid2	datasource	datasource3

8. double clique sur le composant « speedbutton1 » et écrire :

```
Query1.close ;
Query1.params[0].asinteger :=strtoint(edit1.text);
Query1.active :=true ;
Datasource1.dataset :=Query1 ;
```

9. tapez **F12** et double clique sur le composant « speedbutton2 » et écrire:

```
Query2.close ;
Query2.params[0].asdate :=datetimepicker1.date;
Query2.params[1].asdate :=datetimepicker2.date;
Query2.active :=true ;
Datasource1.dataset :=Query2 ;
```

10. tapez **F12** et double clique sur le composant « speedbutton3 » et écrire:

```
Query3.close ;
Query3.params[0].asstring :=combobox1.text;
Query3.active :=true ;
Datasource1.dataset :=Query3 ;
```

11. tapez **F12** et double clique sur le composant « speedbutton4 » et écrire:

```
Query4.close ;
Query4.params[0].asinteger :=strtoint(dblookupcombobox1.text);
Query4.active :=true ;
Datasource1.dataset :=Query4 ;
```

12. cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : double clique sur le « mainmenu1 » et double clique sur son élément Ventes (sous menu Consultation)

Ensuite : on écrit dans la procédure : `fiche_consultation_vente.showmodal` ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_consultation_vente et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

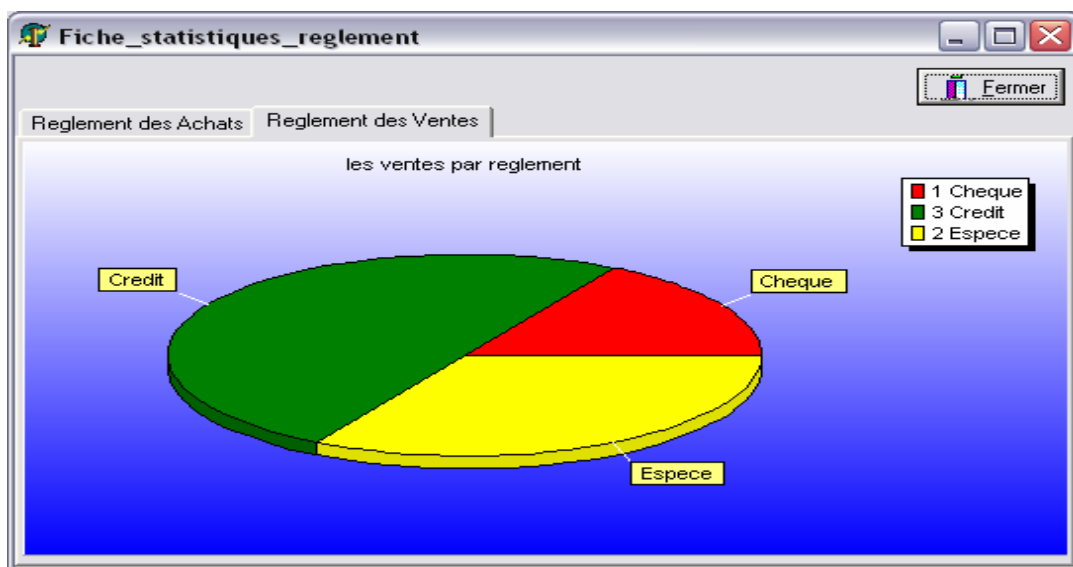
1- Pour que la fiche consultation vente se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_consultation_vente » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ `poScreenCenter` »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

FICHE STATISTIQUES REGLEMENT



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name»le nom:«fiche_statistiques_reglement».

Ensuite:(le menu Delphi): fichier ⇒enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité:«statistiques_reglement»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Query1	databasename	commerce
		Sql	Select reglement_a, count(reglement_a) as nombre from facture_achat group by reglement_a
		Active	True
	Query2	databasename	commerce
		Sql	Select reglement_v, count(reglement_v) as nombre from facture_vente group by reglement_v
		Active	True
supplement	bitbtn	kind	bkclose
win32	Pagecontrol1		
	Clique droit sur ce composant et clique sur « nouvelle page »	Caption	atsReglement des ach
	Clique droit sur ce composant et clique sur « nouvelle page »	Caption	Reglement des ventes

controlebd	DBchart1 (Insérer ce composant sur le composant « pagecontrole1 » dans la page « règlement des achats ») (il faut cliquer sur l'onglés « lément des achatsreg » et cliquer ensuite dans le cadre intérieur de cette page) <u>E</u>t double clique sur ce composant « dbchart1 » <u>E</u>t clique sur l'onglé inférieur « série » <u>E</u>t clique sur le bouton <u>A</u>jouter <u>E</u>t clique sur le graphe de la forme « sectoriel » <u>E</u>t clique sur le bouton <u>O</u>k <u>E</u>t clique sur l'onglé supérieur « série » <u>E</u>t clique sur l'onglé « source de données » <u>E</u>t choisir dans la liste l'élément « dataset » <u>E</u>t choisir dans la liste dataset l'élément « query1 » <u>E</u>t choisir dans la liste « libellé » l'élément « règlement » <u>E</u>t choisir dans la liste « sectoriel » l'élément « nombre » <u>A la fin</u> clique sur <u>F</u>ermer
controlebd	DBchart2 (Insérer ce composant sur le composant « pagecontrole1 » dans la page « règlement des ventes ») (il faut cliquer sur l'onglés « règlement des ventes » et cliquer ensuite dans le cadre intérieur de cette page) <u>E</u>t double clique sur ce composant « dbchart2 » <u>E</u>t clique sur l'onglé inférieur « série » <u>E</u>t clique sur le bouton <u>A</u>jouter <u>E</u>t clique sur le graphe de la forme « sectoriel » <u>E</u>t clique sur le bouton <u>O</u>k <u>E</u>t clique sur l'onglé supérieur « série » <u>E</u>t clique sur l'onglé « source de données » <u>E</u>t choisir dans la liste l'élément « dataset » <u>E</u>t choisir dans la liste dataset l'élément « query2 » <u>E</u>t choisir dans la liste « libellé » l'élément « règlement » <u>E</u>t choisir dans la liste « sectoriel » l'élément « nombre » <u>A la fin</u> clique sur <u>F</u>ermer

Remarque :

Pour que la fiche statistiques achats affiche toujours les dernières modifications automatiquement (rafraîchissement automatique des données du query1 et query2) : Cliquez sur la fiche_statistiques_reglement

Dans son inspecteur d'objet double cliquez à droite de l'événement « onactivate »

Ensuite : écrire le code source suivant dans la procédure correspondante :

Query1.close ;

Query1.active :=true ;

Query2.close ;

Query2.active :=true ;

3- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur Ok.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément Reglements (du sous menu statistiques)

Ensuite : on écrit dans la procédure : fiche_statistiques_reglement.showmodal ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_statistiques_reglement et on doit cliquer sur Oui pour cette demande de confirmation.

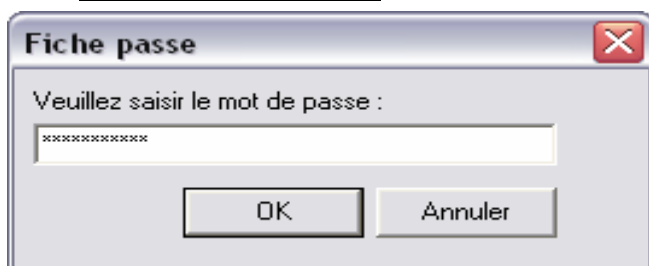
Remarques :

1- Pour que la fiche statistique règlement se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_statistique_reglement » et on clique sur Ok.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- On enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

**FICHE MOT DE PASSE**

1-choisir dans le menu Delphi : fichier ⇒ nouveau ⇒ dialogue ⇒ dialogue de mot de passe.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom: « fiche_passe».

Ensuite : (le menu Delphi) : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité : « passe »
 2- insérer un composant « table1 » (palette BDE) et manipuler ses propriétés :
 Databasename ⇒ commerce table name ⇒ protection active ⇒ true.

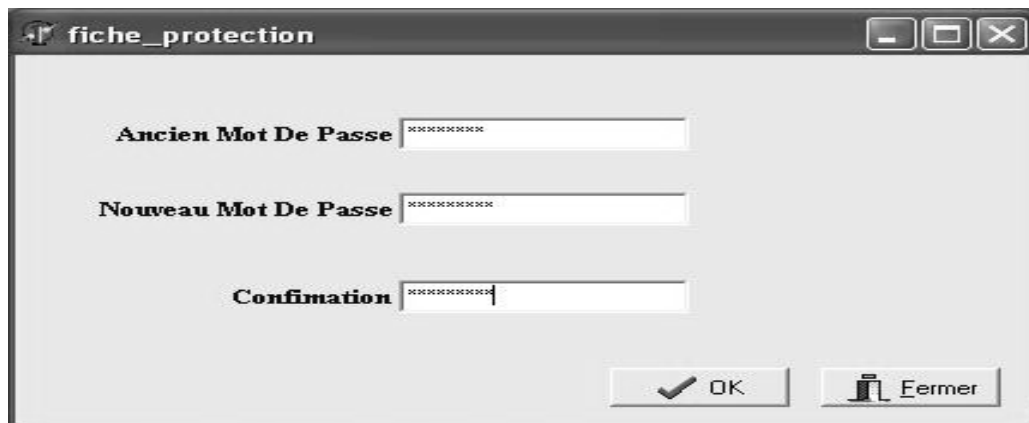
3- Double clique sur le bouton **Ok** et écrire dans sa procédure :
if password.text=table1.fieldbyname('password').asstring then
fiche_menu.showmodal
else
begin
showmessage('mot de passe incorrecte');
password.clear;
password.setfocus;
end;

4- double clique sur le Bouton **Annuler** et écrire dans sa procédure : application.terminate ;

remarques importantes:

- a- Pour que la fiche_passe se lance la première lors de chaque exécution du logiciel on choisit dans le menu Delphi :
 Projet ⇒ options ⇒ dans l'onglet « fiche » ⇒ dans la liste « fiche principale » ⇒ choisir : « fiche_passe » ⇒ **Ok**
 b- Il faut ajouter les deux unités : messages, dialogs, dans la liste uses de l'unité « passe »
Nb : par fois Delphi répète quelques unités de la partie uses si on écrit une nouvelle unité dans cette partie, donc on doit supprimer les unités dupliquées si Delphi signal ce problème de répétition après la compilation du logiciel.
 ❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

FICHE PROTECTION



- 1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.
 Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_protection ».
 Ensuite : (le menu Delphi) : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité : « protection »

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	bitbtn1	kind	bkclosé
	bitbtn2	kind	bkok
standard	label1	caption	eancien mot de pass
	edit1	text	
		passwordchar	*
	label2	caption	nouveau mot de passe
	edit2	text	
		passwordchar	*
	label3	caption	confirmation
	edit3	text	
		passwordchar	*
BDE	table1	databasename	commerce
		tablename	protection
		Active	True

3- double clique sur le composant « BitBtn2 » et écrire dans sa procédure:

```

If edit1.text = table1.fieldbyname('password').asstring then
If edit2.text = edit3.text then
Begin
Table1.edit;
Table1.fieldbyname('password').asstring:=edit2.text;
Table1.post;
Showmessage(' la modification du mot de passe est effectuée ')
End
Else Showmessage(' la confirmation du mot de passe est incorrecte ')
Else Showmessage(' l'ancien mot de passe est incorrecte ');

```

4- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément protection (du sous menu outils)

Ensuite : on écrit dans la procédure : fiche_protection.showmodal ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_protection et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

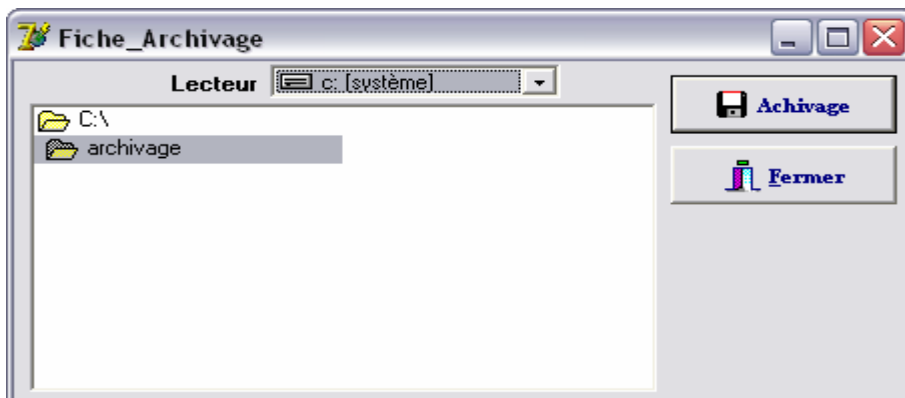
1- Pour que la fiche protection se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_protection » et on clique sur **Ok**.
 - On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
 - On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche.

FICHE ARCHIVAGE



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom:« fiche_archivage ».

Ensuite : (le menu Delphi) : fichier ⇒enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité:« archivage»

2- tapez **F12** et dans la liste uses de l'unité archivage Il faut ajouter l'unité : ShellAPI, .

3- tapez **F12** pour revenir à la fiche et Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
win3.1	FileListBox1	name	fichier
		Mask	*.db
	DirectoryListBox1	name	driverlist
	DriveComboBox1	name	driver
		Dirlist	driverlist
	FilterComboBox1	Filelist	fichier
		filter	*.db
supplement	BitBtn1	kind	bkclose
	Speedbutton1	caption	archiver
		glyph	C:\Program Files\Fichiers communs\Borland shared\Images\Buttons\floppy.bmp
standard	Label1	caption	lecteur

4- Double clique sur le composant speedbutton1 et écrire dans sa procédure :

```
if CopyFolder('c:\commerce\base', driverlist.Directory) then
  ShowMessage('Archivage effectué avec succès') else
  ShowMessage('Archivage non effectué');
```

5- il faut inserer avant la procedure BitBtn4Click la fonction suivante :

```
function CopyFolder(FromFld, ToFld: string): boolean;
var fos: TSHFileopStruct;
begin
  FillChar(fos, SizeOf(fos), 0);
  with fos do
  begin
    wFunc := FO_COPY;
    pFrom := PChar(FromFld+#0);
    pTo := PChar(ToFld+#0);
    fFlags := FOF_SILENT or FOF_NOCONFIRMATION or FOF_NOCONFIRMMKDIR;
  end;
  Result := ShFileOperation(fos)=0;
end;
```

6- pour avoir un bon interface on doit cacher les deux composants «FileListBox1» et «FilterComboBox1»comme suite:

- clique par le bouton droit de la souris sur le composant : « FileListBox1 » nommé « fichier » et choisir dans le menu contextuel : « mettre en arrière plan » et mettre ce composant au-dessous du « DirectoryListBox1 » (driverlist).

- aussi, clique par le bouton droit de la souris sur le composant : « FilterComboBox1 » et choisir dans le menu contextuel : « mettre en arrière plan » et mettre ce composant au-dessous du « DirectoryListBox1 » (driverlist).

7- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément archivage (du sous menu outils)

Ensuite : on écrit dans la procédure : **fiche_archivage.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_archivage et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

1- Pour que la fiche archivage se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

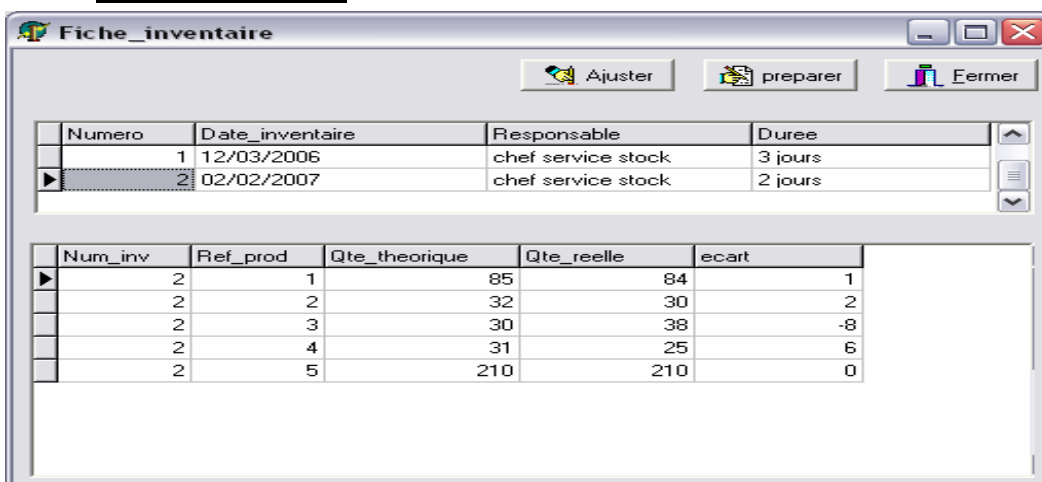
- On tape : **Shift + F12** et on clique sur la « fiche_archivage » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

❖ Cette fiche nécessite la création d'un dossier dans disque dur (nommé par exemple : archive2007) qui sera le dossier où on va archiver notre base de données.

FICHE INVENTAIRE



Numero	Date_inventaire	Responsable	Duree
1	12/03/2006	chef service stock	3 jours
2	02/02/2007	chef service stock	2 jours

Num_inv	Ref_prod	Qte_theorique	Qte_reelle	ecart
2	1	85	84	1
2	2	32	30	2
2	3	30	38	-8
2	4	31	25	6
2	5	210	210	0

1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom:« fiche_inventaire».

Ensuite : (le menu Delphi) : fichier ⇒ enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité:« inventaire»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	bitbtn1	kind	bkclose
		caption	Preparer
	speedbutton1	glyph	(choisissez une image adéquate)
		caption	ajuster
BDE	Table1	glyph	(choisissez une image adéquate)
		Databasename	Commerce
		Tablename	inventaire
		Active	True
Accesbd	Datasource1	Dataset	Table1
Controlebd	Dbgrid1	Datasource	Dataource1
BDE	Table2	Databasename	Commerce
		Tablename	Ligne_inventaire
		Active	True
		Master Source	datasource1
		Master Field	Clique sur « num_inv » et clique sur « numero_i »et clique sur [Ajouter] et clique sur [Ok]
Accesbd	Datasource2	Dataset	Table2
Controlebd	Dbgrid2	Datasource	Dataource2
BDE	Table3	Databasename	Commerce
		Tablename	produit
		Active	True

3- insérer le champ calculé « écart » qui représente la différence entre le stock théorique et le stock réel comme suite :

On double clique sur le composant table2 « ligne_inventaire »

Ensuite : on clique droit sur la petite fenêtre et on clique sur « ajouter tous les champs ».

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « écart » et on choisit le type du champ : « float » et on clique sur [Ok].

Ensuite : on clique sur le composant table2 et dans son inspecteur d'objet on clique sur l'onglés « événements ».

Ensuite : on double clique à droite de l'événement « oncalcfields » et on écrit dans la procédure :

**table2.fieldbyname('ecart').asfloat := table2.fieldbyname('qte_theorique').asfloat –
table2.fieldbyname('qte_reelle').asfloat ;**

4- on prépare l'inventaire par un double clic sur le speedbutton1 « preparer » et on écrit dans sa procédure :
(Supprimer le **begin** et **end** de la procédure et coller tous le code source suivant)

```
var trouver : boolean;
begin
table2.first;
trouver:=false;
while not(table2.eof) do
begin
if table2.fieldbyname('ref_prod').asinteger=table3.fieldbyname('reference').asinteger then
trouver:=true;
table2.next;
end;
if trouver then
showmessage('inventaire deja préparé')
else
begin
table3.first;
while not(table3.eof)do
begin
table2.append;// on copie les produits de la table produit vers la table ligne_invertaire.
table2.FieldByName('ref_prod').asinteger:=table3.FieldByName('reference').asinteger;
table2.FieldByName('qte_theorique').asfloat:=table3.FieldByName('qte_stock').asfloat;
table2.post;
table3.next;
end;
end;
end;
```

5- après avoir saisi les qte_reelle de chaque produit, on ajuste le stock sur ordinateur (la quantité_stock sera donc la quantité réelle) par un double clic sur le speedbutton2 « ajuster » et on écrit dans sa procédure :

```
table3.first;
table2.first;
while not(table3.eof)do
begin
table3.edit;
table3.FieldName('qte_stock').asfloat:=table2.FieldName('qte_reelle').asfloat;
table3.post;
table2.next;
table3.next;
end;
table3.refresh;
```

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

6- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément inventaire (du sous menu outils)

Ensuite : on écrit dans la procédure : **fiche_inventaire.showmodal** ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_inventaire et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

1- Pour que la fiche inventaire se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_inventaire » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)



FICHE CLOTURE



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom:« fiche_Cloture».

Ensuite : (le menu Delphi) : fichier ⇒enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité:« Cloture»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	speedbutton1	caption	Vider Table Facture Achat
		glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp
	speedbutton2	caption	Vider Table Ligne Achat
		glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp
	speedbutton3	caption	Vider Table Facture Vente
		glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp
	speedbutton4	caption	Vider Table Ligne Vente
		glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp

BDE	speedbutton5	caption	Vider Table Inventaire
		glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp
	speedbutton6	caption	Vider Table Ligne Inventaire
		glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp
	Bitbtn1	kind	bkclose
	Table1	Databasename	Commerce
		Tablename	Facture_achat
		Active	True
	Table2	Databasename	Commerce
		Tablename	Ligne_achat
		Active	True
	Table3	Databasename	Commerce
		Tablename	Facture_vente
		Active	True
	Table4	Databasename	Commerce
		Tablename	Ligne_vente
		Active	True
	Table5	Databasename	Commerce
		Tablename	Inventaire
		Active	True
	Table6	Databasename	Commerce
		Tablename	ligne_inventaire
		Active	True

3- double clique sur le « speedbutton1 » et écrire dans sa procédure :

```
While not (table1.eof) do  
table1.delete ;  
showmessage('la table facture achat est vidée');
```

- Tapez **F12** pour retourner à la fiche.

4- double clique sur le « speedbutton2 » et écrire dans sa procédure :

```
While not (table2.eof) do  
Table2.delete ;  
showmessage('la table ligne achat est vidée');
```

- Tapez **F12** pour retourner à la fiche.

5- double clique sur le « speedbutton3 » et écrire dans sa procédure :

```
While not (table3.eof) do  
Table3.delete ;  
showmessage('la table facture vente est vidée');
```

- Tapez **F12** pour retourner à la fiche.

6- double clique sur le « speedbutton4 » et écrire dans sa procédure :

```
While not (table4.eof) do  
Table4.delete ;  
showmessage('la table ligne vente est vidée');
```

- Tapez **F12** pour retourner à la fiche.

7- double clique sur le « speedbutton5 » et écrire dans sa procédure :

```
While not (table5.eof) do  
Table5.delete ;  
showmessage('la table inventaire est vidée');
```

- Tapez **F12** pour retourner à la fiche.

8- double clique sur le « speedbutton6 » et écrire dans sa procédure :

```
While not (table6.eof) do  
Table6.delete ;  
showmessage('la table ligne inventaire est vidée');
```

- Tapez **F12** pour retourner à la fiche.

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

9- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément Cloture (du sous menu outils)

Ensuite : on écrit dans la procédure : **fiche_cloture.showmodal** ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_cloture et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

1- Pour que la fiche cloture se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_cloture » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

FICHE A PROPOS



1- on peut insérer cette fiche par le menu Delphi :

Fichier ⇒ nouveau ⇒ fiche ⇒ boîte a propos ⇒ **Ok**

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name»le nom:«fiche_apropos».

Ensuite : (le menu Delphi): fichier ⇒enregistrer, et dans le dossier « c:\commerce\source » nommer l'unité:«apropos»

Ensuite : on écrit le texte qu'on veut dans les labels et modifier la couleur et la taille des labels par leurs propriétés « FONT ». (Cette fiche est la carte visite du logiciel).

2- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément « a propos » (du sous menu « ? »)

Ensuite : on écrit dans la procédure : **fiche_apropos.showmodal** ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_apropos et on doit cliquer sur **Oui** pour cette demande de confirmation.

L'image de Logo :

1- dans la propriété picture du composant « image1 » on clique sur le bouton **Charger** et chercher l'image voulue. (Delphi propose quelques images dans :

« C:\Program Files\Fichiers communs\Borland Shared\Images\ Splash\256Color ».

2- en fin clique sur **Ouvrir** ensuite sur **Ok**

3- pour que l'image occupe toute la surface de son cadre on modifie sa propriété : Autosize ⇒ true.

4- pour que l'image s'agrandisse si elle a une taille plus petite que la surface de son cadre on modifie sa propriété : Stretch ⇒ true.

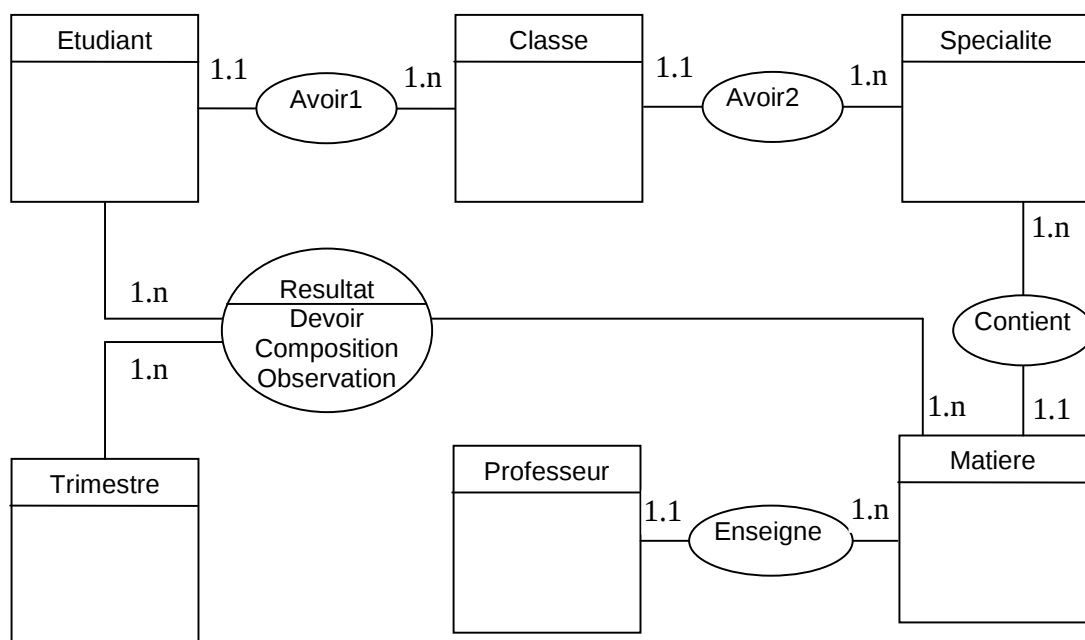
CHAPITRE 3

LOGICIEL GESTION SCOLARITE

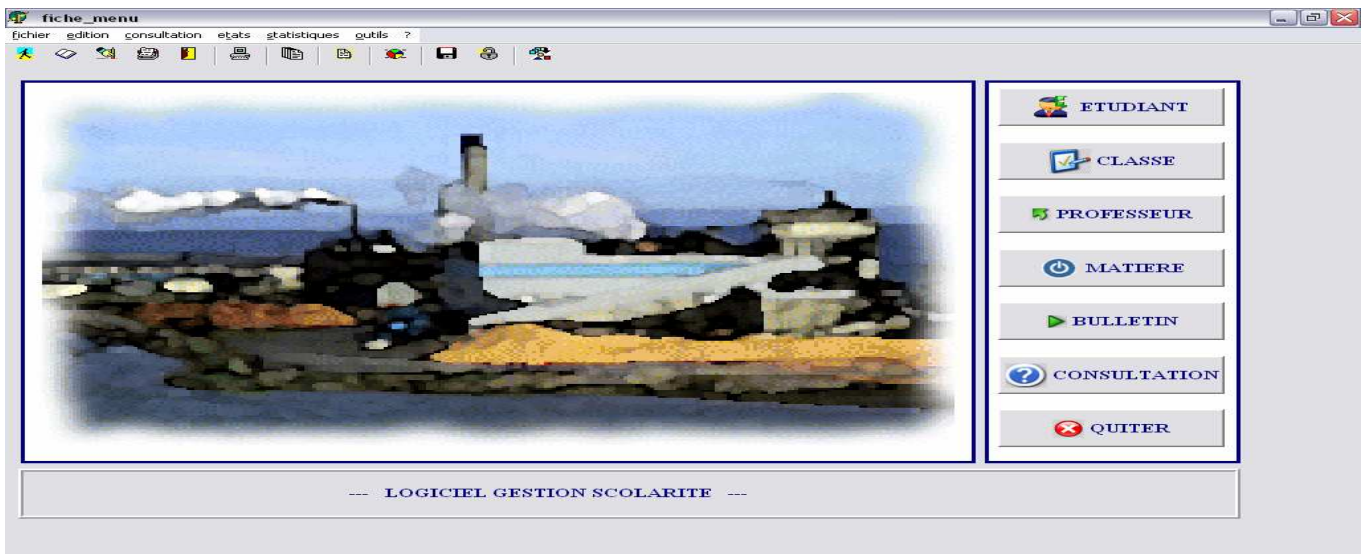
PREPARATION DU LOGICIEL

- 1- créer dans le lecteur C du disque dur un nouveau dossier nommé : scolarite (pour stocker le logiciel).
- 2- créer dans le dossier sclarite 2 autres dossiers nommés : base (pour stocker la BD) et source (pour stocker le code source)
- 3- lancer le BDE de Delphi par : (le menu Delphi) outils ⇒ module base de données.
- 4- créer un nouveau alias pour notre logiciel par : (le menu BDE) outils ⇒ gestionnaire d'alias
Ensuite : clique sur l'option : «  afficher seulement les alias publics »
Ensuite : clique sur le bouton : **Nouveau**
Ensuite : écrire dans : Alias de la base : sclarite
Chemin d'accès : c:\sclarite\base
Ensuite clique sur **OK**
En fin clique sur **Oui** pour que cet alias soit un alias public.
- 5- créer dans cet alias les tables suivantes : N'oubliez pas de les enregistrer dans l'alias « sclarite »

Table: etudiant				Table : matiere				Table : resultat			
Champ	Type	Taille	index	Champ	Type	Taille	index	Champ	Type	Taille	index
Code_e	i		*	Reference_m	i		*	Code_etudiant	i		*
Nom_e	A	20		Designation_m	a	20		Ref_matiere	i		*
Prenom_e	A	20		Coefficient	i			Trimestre	i		*
Date_naissance	D			Ref_specialite				Devoir	n		
Lieu	A	15		Table : classe				Composition	n		
Adresse	A	30		Champ	Type	Taille	index	Observation	a	15	
Doublant	A	3		Ref_classe	i		*	Table protection			
Photo	G			Designation_c	a	20		Champ	Type	Taille	index
Ref_classe	i			Ref_Specialite	i			Password	A	20	*
Table : professeur				Table : specialite							
Champ	Type	Taille	index	Champ	Type	Taille	Index				
Code_p	i		*	Ref_Specialite	i		*				
Nom_p	a	20		Designation_s	a	20					
Prenom_p	a	20									
Diplome	a	30									
Ref_Matiere	i										



FICHE MENU



1- Dans l'environnement Delphi choisir dans le menu : Fichier ⇒ nouvelle application.

Ensuite : insérer dans la fiche de ce nouveau projet (nouveau logiciel) un composant « mainmenu »

Ensuite : double clique sur ce composant et insérer le menu principal suivant :

Fichier	Edition	Consultation	États	Statistiques	Outils	?
étudiant	Bulletin	Étudiants	impressions	scolarité	Archivage	A propos
professeur					Protection	
spécialité					Cloture	
matière						
quitter						

- ❖ Chaque fiche (ainsi que son unité) dans le logiciel doit être nommée par un nom significatif. pour notre fiche actuelle (le menu général du projet) on procède comme suite :

Cliquer sur cette fiche (éviter le composant mainmenu1) et dans son inspecteur d'objet écrire dans la propriété : name ⇒ fiche_menu.

Clique dans le menu Delphi sur: fichier ⇒ enregistrer sous, et dans le dossier «c:\scolarite\source» nommer l'unité : «menu».

Clique dans le menu Delphi sur: fichier ⇒ enregistrer le projet sous, et nommer le projet : scolarite (dans c:\scolarite\source)

2- insérer un composant ToolBar (palette win32).

Ensuite : clique droit sur ce composant et choisir « nouveau bouton » ou « nouveau séparateur »

Ensuite : répéter le clic droit pour insérer 13 boutons qui représentent les éléments du menu principal du logiciel; Et 6 séparateurs pour les grands éléments du menu (fichier, édition, consultation, ...)

Ensuite : insérer un composant imagelist (palette win32)

Ensuite : double clique sur ce composant et clique sur le bouton **Ajouter** pour choisir la gamme des images de notre logiciel dans le chemin : « C:\program files\ fichiers communs\borland shared\images\button »

Ensuite : après le choix de 13 images adéquates aux 13 boutons on clique sur **Ok**

Ensuite : on clique sur le « mainmenu1 » et dans son inspecteur d'objet : propriété images ⇒ imagelist1

Ensuite : on double clique sur le « mainmenu1 » et chaque élément de ce menu sera lié avec une image adéquate par sa propriété : « imageindex »

Enfin: on ferme la petite fenêtre du menu et on clique sur le composant « toolbar1 » et dans sa propriété :

Images ⇒ imagelist1

(Chaque bouton du «toolbar1» peut avoir une image adéquate par le choix de cette image dans la propriété « imageindex » de ce bouton).

3- pour expliquer le rôle de chaque bouton on clique sur ce dernier on écrit dans sa propriété :

« Hint » l'explication adéquate, ensuite et pour voir cette explication on choisit dans sa propriété : showHint ⇒ true

Remarque :

Pour faire défiler le nom du logiciel dans cette fiche :

- 1- on insère un composant timer (palette systeme)
et sa propriété interval = 10

2- on insère un composant Label1
propriété :caption = LOGICIEL GESTION SCOLARITE
propriété : font (essayez de bien traiter le texte).

3- on double clique sur le timer et on écrit :

```
if label1.left > -250 then
label1.left:= label1.left-1
else
label1.left :=700 ;
```

// Les intervalles d'affichage du label1 peuvent varier selon la taille d'écriture (pas sûrement entre -250 et 700).

Programmation du sous menu « QUITTER »

On double clique sur le composant « mainmenu1 » et double clique sur son élément quitter (du sous menu fichier).

Ensuite : on écrit dans la procédure : close;

Ensuite : clique sur la fiche_menu et double clique à droite de son événement « onclosequery » et écrire dans sa procédure :

```
if messagedlg('voulez vous quitter le logiciel?',mtconfirmation,[mbyes,mbno],0)=mryes then
begin
canclose:=true;
application.terminate;
end
else
canclose:=false;
```

Résultat:

Cette technique assure l'affichage de la fenêtre de confirmation de sortie du logiciel pour n'importe quel type de tentative de fermeture du logiciel :

- le menu : fichier ⇒ quitter.
- Par clavier en utilisant les touches : **Alt + F4**
- En cliquant sur la case fermer en haut et à droite de la fenêtre.

4- Pour que la fiche menu occupe tous l'espace de l'écran lors du lancement de l'exécution du logiciel :

- On clique sur cette fiche_menu (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « Window state ⇒ wsmaximized »

5- On enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

L'image de l'arrière plan :

1- insérer un composant « image1 » de la palette « supplément ».

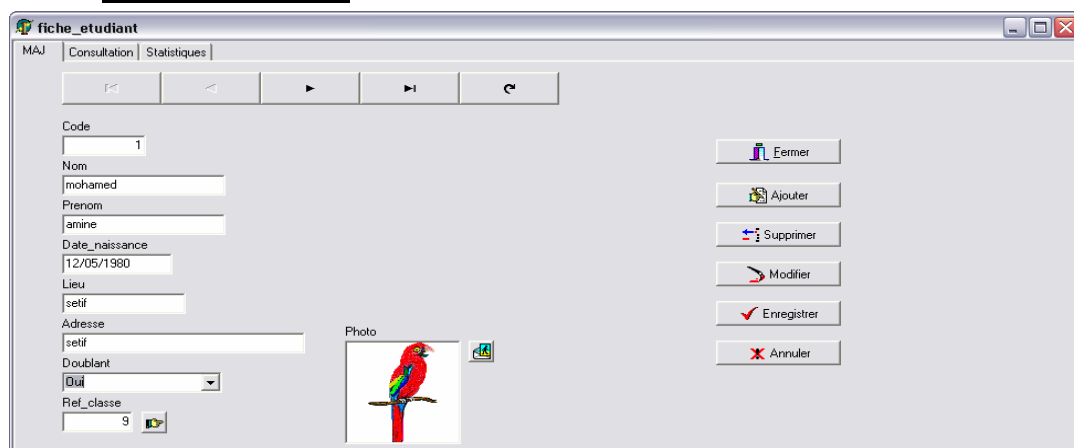
2- dans la propriété Picture sur ce composant « image1 » on clique sur le bouton **Charger** et chercher l'image voulue. (Exemple du chemin de l'image : C:\Program Files\Fichiers communs\Borland Shared\Images\Splash\256Color)

3- en fin clique sur **Ouvrir** ensuite sur **OK**

4- pour que l'image occupe toute la surface de son cadre on modifie sa propriété : Autosize ⇒ true.

5- pour que l'image s'agrandisse si elle a une taille plus petite que la surface de son cadre on modifie sa propriété : Stretch ⇒ true. Aussi modifier sa propriété : align ⇒ alClient.

LA FICHE ETUDIANT



1- Choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_etudiant ».

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « etudiant »

2- Insérer de la palette win32 le composant Pagecontrol1

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « MAJ »

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « Consultation »

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « Statistiques »

3- clique sur l'onglet « MAJ » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	scolarite
		Tablename	etudiant
		Active	True

4- double clique sur le composant table1 et clique droit sur la petite fenêtre et choisir dans la liste « ajouter tous les champs »

5- glisser les champs un par un dans la fiche dans l'ordre de haut en bas (on peut les sélectionner tous et les glisser en même temps, c'est plus professionnel)

6- Insérer de la palette Supplément 5 composants « Speedbutton » et écrire dans la propriété de chacun respectivement :

« Ajouter » « Supprimer » « Modifier » « Enregistrer » « Annuler »

7- double clique sur le « speedbutton1 » (ajouter) et écrire dans sa procédure :

table1.append ; // table1.append = table1.last + table1.insert ⇒ c'est insérer un nouveau enregistrement en fin de la table.

dbedit1.setfocus ;

8- tapez **F12** et double clique sur le composant « speedbutton2 » (Supprimer) et écrire dans sa procédure :
**if messagedlg(' voulez vous supprimer cet enregistrement' ,mtconfirmation,[mbytes,mbno],0)=mryes then
table1.delete ;**

9- tapez **F12** et double clique sur le composant « speedbutton3 » (Modifier) et écrire dans sa procédure :

table1.edit ;

dbedit1.setfocus ;

10- tapez **F12** et double clique sur le composant « speedbutton4 » (Enregistrer) et écrire dans sa procédure :

table1.edit ;

table1.post ;

11- tapez **F12** et double clique sur le composant « speedbutton5 » (Annuler) et écrire dans sa procédure :

table1.edit ;

table1.cancel ;

12- Insérer en haut de cette page un composant « DbNavigateur1 » de la palette « ControleBD ».

Ensuite : Choisir dans sa propriété : datasource ⇒ datasource1.

Ensuite : choisir dans sa propriété *VisibleButtons* les boutons *first, prior, next, last, refresh* ; mettre visible= true.

13- Pour insérer les images des étudiants dans le champ « Photo » de la table1 (table etudiant) à partir d'un fichier image stocké dans le disque dur (généralement on doit scanner les images des étudiants au paravant) on procède comme suite :

- insérer de la palette « dialogues » un composant « OpenPictureDialog1 ».
- Insérer de la palette « supplément » un composant « speedbutton » et choisir dans sa propriété « glyph » l'image : C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\mailpict.bmp
- Double clique sur ce « speedbutton » et écrire dans sa procédure :

OpenPictureDialog1.execute;

dbimage1.picture.loadfromfile(OpenPictureDialog1.filename);

14- Pour mettre le curseur du clavier automatiquement dans le « dedit1 » pour faciliter la saisie du code de l'étudiant on clique sur le composant « table1 » et dans son inspecteur d'objet double clique à droite de son événement : « afterinsert » on écrit dans sa procédure :

dbedit1.setfocus ;

Ensuite on peut contrôler la manipulation du clavier et calculer l'age de l'etudiant par :

- insérer un composant label9 à droite du « dedit4 » et supprimer le texte de sa propriété « caption ».
- clique sur le composant dedit1.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :

If key = vk_return then

Dbedit2.setfocus ;

- tapez **F12** et clique sur le composant dedit2.

- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :

```

If key = vk_return then
Dbedit3.setfocus ;
- tapez F12 et clique sur sur le composant Dbedit3.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
Dbedit4.setfocus ;
- tapez F12 et clique sur sur le composant Dbedit4.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
(Supprimer le begin et end de la procédure et coller tous le code source suivant)
var age : integer ;
begin
If key = vk_return then
Begin
Table1.edit;
Table1.post;
Age:=trunc((date-table1.fieldbyname('date_naissance').asdatetime)/365);
Label9.caption:=inttostr(age)+' ans';
If age <17 then
Showmessage('étudiant mineur');
If age >60 then
Showmessage('étudiant retrété');
Dbedit5.setfocus ;
End;
End;
- tapez F12 et clique sur le composant Dbedit5.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
Dbedit6.setfocus ;
- tapez F12 et clique sur sur le composant Dbedit6.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
Dbcombobox1.setfocus ;
- tapez F12 et clique sur sur le composant Dbcombobox1.
- Double clique sur son événement « OnKeyDown » et écrire dans sa procédure :
If key = vk_return then
Dbedit8.setfocus ;

```

15- clique sur l'onglés « Consultation » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	valeur
controlebd	Dbgrid1	Datasource	Datasource1
Supplément	Speedbutton6	Caption	Previsualiser
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Speedbutton7	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose

16- Pour imprimer la liste des étudiants on procède comme suite :

- Dans le menu Delphi on choisit : fichier ⇒ nouveau
- Dans la boite de dialogue qui s'affiche on choisit l'onglet « affaires » et dans ce dernier on choisit l'icône « expert quick report » et clique sur le bouton **Ok**.
- Dans l'assistant qui s'affiche on clique sur le bouton **lancer l'expert**.
- Choisir dans la liste « alias ou répertoire » l'alias « scolaire ».
- On choisit dans la liste « nom de table » la table à imprimer (etudiant).
- On clique sur le bouton **>>** pour faire passer tous les champs à droite (pour les imprimer tous).
- Ensuite clique sur le bouton **Terminer**

17- Ensuite : on clique sur cette nouvelle fiche en évitant le clic sur le composant « quickrep1 » et dans sa propriété « name » on donne le nom « fiche_etat_etudiant » à cette fiche.

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « etat_etudiant »

18- On tape **Shift+F12** pour afficher la liste des fiches et on choisit la fiche « fiche_etudiant » et on clique sur **Ok**.

Ensuite : pour programmer la prévisualisation de notre liste on double clique sur le bouton « previsualiser » et on écrit dans sa procédure : **fiche_etat_etudiant.quickrep1.preview ;**

Ensuit : on tape **F12** pour retourner à la fiche et pour programmer l'impression de notre liste on double clique sur le bouton « imprimer » et on écrit dans sa procédure : **fiche_etat_etudiant.quickrep1.print** ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_etudiant et la fiche_etat_etudiant et on doit cliquer sur **Oui** pour cette demande de confirmation.

16- On ajoute aussi les options du filtrage et du trie dans cette fiche par l'insertion des composants suivants (toujours dans l'onglés consultation) :

Palette	Composant	Propriété	valeur
Standard	Label1	Caption	Filtrage
	Edit1	Text	

8- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange):

```

if edit1.text <> '' then
begin
table1.close ;
table1.filtered :=true ;
table1.filter :='[Nom_e]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;

```

17- On tape la touche **F12** (retour à la fiche), et pour faire le trie de la table etudiant : clique sur le composant dbgrid1.

Ensuite : dans son inspecteur d'objet on clique sur l'onglés « événements »

Ensuite : on double clique à droite de l'événement « ontitleclick » et on introduit le code source suivant :

```

If column.index = 0 then
Table1.indexname :='c1' ;
If column.index = 1 then
Table1.indexname :='c2' ;
If column.index = 2 then
Table1.indexname :='c3' ;
If column.index = 3 then
Table1.indexname :='c4' ;
If column.index = 4 then
Table1.indexname :='c5' ;
If column.index = 5 then
Table1.indexname :='c6' ;
If column.index = 6 then
Table1.indexname :='c7' ;
If column.index = 7 then
Table1.indexname :='c8' ;

```

Remarque importante :

Le trie et le filtrage dans une table ne réussissent que si on a défini des index secondaires sur chaque champ trié ou filtré.

Pour insérer des index secondaires dans les champs de la table etudiant il faut que cette table ne soit pas en cours d'utilisation, donc sa propriété : active ⇒ false.

Ensuite : on lance le BDE Delphi par : (le menu Delphi) : outils ⇒ module base de données.

Ensuite : on ouvre la table par : (le menu) fichier ⇒ ouvrir ⇒ table et on choisit l'alias « scolarite » et la table « etudiant ».

Ensuite : on clique dans le menu du BDE sur : table ⇒ restructurer.

Ensuite : on choisit dans la liste des choix « propriétés de la table » l'option « index secondaires »

Ensuite : on clique pour chaque champ sur le bouton **Définir** et on fait passer ce champ à droite, on clique sur le bouton **OK** et on propose un nom symbolique à cet index secondaire (c1 pour le champ code_e, c2 pour le champ nom_e,...etc.)

Remarque : le champ photo ne peut pas avoir un index secondaire puisque c'est un champ de type graphique (Un champ de type graphique ne peut être utilisé dans une recherche ou un tri ou un filtrage).

Ensuite : on clique sur le bouton **enregistrer** et on retourne à l'environnement Delphi.

Enfin : on réactive le composant table1 par sa propriété : active $\Rightarrow$ true.

18- clique sur l'onglet « **Statistiques** » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Query1	databasename	scolarité
		Sql	Select Doublant, count(Doublant) as nombre from etudiant group by Doublant
		Active	True
controlebd	DBchart1 (Insérer ce composant sur le composant « pagecontrol1 » dans la page « statistiques ») Et double clique sur ce composant « dbchart1 » Et clique sur l'onglet inférieur « série » Et clique sur le bouton Ajouter Et clique sur le graphe de la forme « sectoriel » Et clique sur le bouton OK Et clique sur l'onglet supérieur « série » Et clique sur l'onglet « source de données » Et choisir dans la liste l'élément « dataset » Et choisir dans la liste dataset l'élément « query1 » Et choisir dans la liste « libellé » l'élément « doublant » Et choisir dans la liste « sectoriel » l'élément « nombre » A la fin clique sur Fermer		

Remarque :

Pour que les statistiques affichent toujours les dernières modifications automatiquement (rafraîchissement automatique des données du query1) : Clique sur le composant table1 et Dans son inspecteur d'objet double clique à droite de son événement « afterpost » (cet événement se déclenche automatiquement si on enregistre des données sur notre table1).
Ensuite : écrire le code source suivant dans la procédure correspondante :

Query1.close ;

Query1.active :=true ;

19- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **OK**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément etudiant (du sous menu fichier)

Ensuite : on écrit dans la procédure : **fiche_etudiant.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_etudiant et on doit cliquer sur **oui** pour cette demande de confirmation.

Remarques :

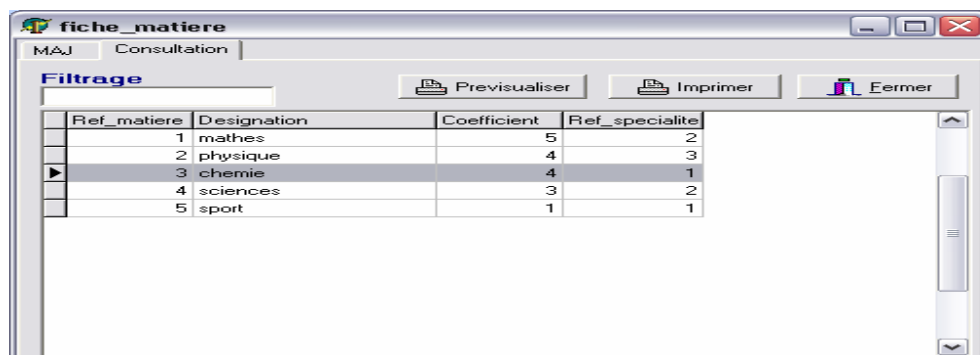
1- Pour que la fiche_etudiant se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_etudiant » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position $\Rightarrow$ poScreenCenter »

2- On enregistre notre travail par le menu Delphi : fichier $\Rightarrow$ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

FICHE MATIERE



- 1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.
Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_matiere ».
Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « matiere »
- 2- Insérer de la palette win32 le composant Pagecontrol1
Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « MAJ »
Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « Consultation »
- 3- Clique sur l'onglet « MAJ » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Scolarite
		Tablename	matiere
		Active	True

- 4- double clique sur le composant table1 et clique droit sur la petite fenêtre et choisir dans la liste « ajouter tous les champs »
- 5- glisser les champs un par un dans la fiche dans l'ordre de haut en bas (on peut les sélectionner tous et les glisser en même temps, c'est plus professionnel)
- 6- Insérer de la palette Supplément 5 composants « Speedbutton » et écrire dans la propriété de chacun respectivement : « Ajouter » « Supprimer » « Modifier » « Engersitrer » « Annuler »
- 7- double clique sur le « speedbutton1 » (ajouter) et écrire dans sa procédure :
table1.append ;
// table1.append = table1.last + table1.insert ⇒ c'est insérer un nouveau enregistrement en fin de la table.
dbedit1.setfocus ;
- 8- taper **F12** et double clique sur le composant « speedbutton2 » (Supprimer) et écrire dans sa procédure :
if messagedlg(' voulez vous supprimer cet enregistrement' ,mtconfirmation,[mbytes,mbno],0)=mryes then
table1.delete ;
- 9- taper **F12** et double clique sur le composant « speedbutton3 » (Modifier) et écrire dans sa procédure :
table1.edit ;
dbedit1.setfocus ;
- 10- taper **F12** et double clique sur le composant « speedbutton4 » (Enregistrer) et écrire dans sa procédure :
table1.edit ;
table1.post ;
- 11- taper **F12** et double clique sur le composant « speedbutton5 » (Annuler) et écrire dans sa procédure :
table1.edit ;
table1.cancel ;
- 12- Insérer en haut de cette page un composant « DbNavigateur1 » de la palette « ControleBD ».
Ensuite : Choisir dans sa propriété : datasource ⇒ datasource1.
Ensuite : choisir dans sa propriété *VisibleButtons* les boutons *first, prior, next, last, refresh* ; mettre visible= true.
- 13- Clique sur l'onglet « Consultation » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	valeur
controlebd	Dbgrid1	Datasource	Datasource1
Supplement	Speedbutton6	Caption	Previsualiser
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Speedbutton7	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose

- 14- Pour imprimer la liste des matières on procède comme suite :
 - Dans le menu Delphi on choisit : fichier ⇒ nouveau.
 - Dans la boîte de dialogue qui s'affiche on choisit l'onglet « affaires » et dans ce dernier on choisit l'icône « expert quick report » et clique sur le bouton **Ok.**
 - Dans l'assistant qui s'affiche on clique sur le bouton **lancer l'expert.**
 - Choisir dans la liste « alias ou répertoire » l'alias « scolarite ».
 - On choisit dans la liste « nom de table » la table à imprimer (matiere).
 - On clique sur le bouton **>>** pour faire passer tous les champs à droite (pour les imprimer tous).
 - Ensuite clique sur le bouton **Terminer**
- 15- Ensuite : on clique sur cette nouvelle fiche en évitant le clic sur le composant « quickrep1 » et dans sa propriété « name » on donne le nom « fiche_etat_matiere » à cette fiche.
Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « etat_matiere »

16- On tape **Shift+F12** pour afficher la liste des fiches et on choisit la fiche « fiche_matiere » et on clique sur . Ensuite : pour programmer la prévisualisation de notre liste on double clique sur le bouton « previsualiser » et on écrit dans sa procédure : **fiche_etat_matiere.quickrep1.preview ;**

Ensuite : on tape **F12** pour retourner à la fiche et pour programmer l'impression de notre liste on double clique sur le bouton « imprimer » et on écrit dans sa procédure : **fiche_etat_matiere.quickrep1.print ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_matiere et la fiche_etat_matiere et on doit cliquer sur pour cette demande de confirmation.

17- On ajoute aussi les options du filtrage et du trie dans cette fiche par l'insertion des composants suivants (toujours dans l'onglet consultation) :

Palette	Composant	Propriété	Valeur
Standard	Label1	Caption	Filtrage
	Edit1	Text	

18- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange):

```
if edit1.text <> '' then
begin
table1.close ;
table1.filtered := true ;
table1.filter := '[designation_m]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;
```

19- On tape la touche **F12** (retour à la fiche), et pour faire le trie de la table matiere : clique sur le composant dbgrid1.

Ensuite : dans son inspecteur d'objet on clique sur l'onglet « événements »

Ensuite : on double clique à droite de l'événement « onclick » et on introduit le code suivant :

```
If column.index = 0 then
Table1.indexname := 'c1' ;
If column.index = 1 then
Table1.indexname := 'c2' ;
If column.index = 2 then
Table1.indexname := 'c3' ;
```

Remarque importante :

Le trie et le filtrage dans une table ne réussissent que si on a défini des index secondaires sur chaque champ trié ou filtré.

Pour insérer des index secondaires dans les champs de la table matiere il faut que cette table ne soit pas en cours d'utilisation donc sa propriété : active ⇒ false.

Ensuite : on lance le BDE Delphi par : (le menu Delphi) : outils ⇒ module base de données.

Ensuite : on ouvre la table par : (le menu) fichier ⇒ ouvrir ⇒ table et on choisit l'alias « scolarite » et la table « matiere ».

Ensuite : on clique dans le menu du BDE sur : table ⇒ restructurer.

Ensuite : on choisit dans la liste des choix « propriétés de la table » l'option « index secondaires »

Ensuite : on clique pour chaque champ sur le bouton et on fait passer ce champ à droite, on clique sur le bouton et on propose un nom symbolique à cet index secondaire (c1 pour le champ reference_m, c2 pour le champ designation_m,...etc.)

Ensuite : on clique sur le bouton et on retourne à l'environnement delphi.

Enfin : on réactive le composant table1 par sa propriété : active ⇒ true.

20- Cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur .

Ensuite : double clique sur le composant « mainmenu1 » et double clique sur son élément matiere (sous menu fichier)

Ensuite : écrire dans la procédure : **fiche_matiere.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_matiere et on doit cliquer sur pour cette demande de confirmation.

Remarques :

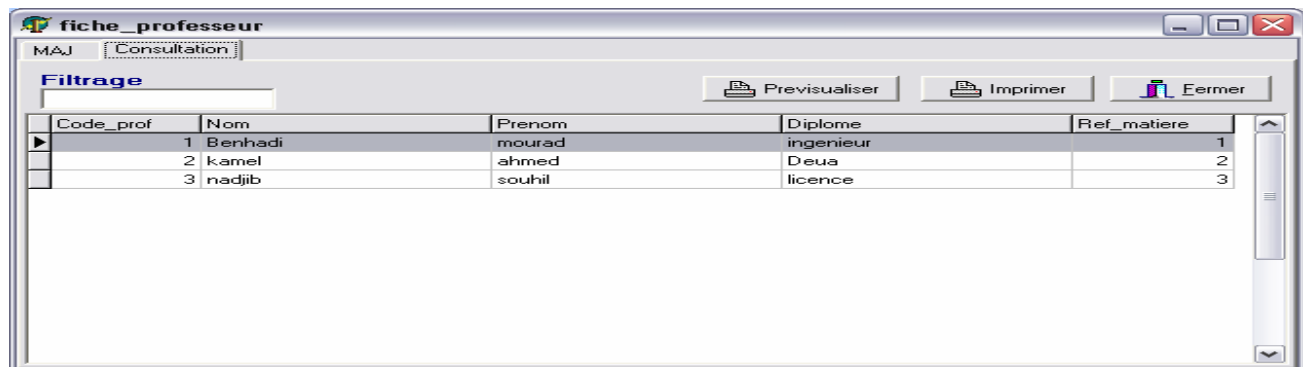
1- Pour que la fiche matiere se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_matiere » et on clique sur **OK**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

 FICHE PROFESSEUR



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_professeur ».

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « professeur »

2- Insérer de la palette win32 le composant Pagecontrol1

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « MAJ »

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « Consultation »

3- clique sur l'onglet « MAJ » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	scolarite
		Tablename	professeur
		Active	True

4- double clique sur le composant table1 et clique droit sur la petite fenêtre et choisir dans la liste « ajouter tous les champs »

5- glisser les champs un par un dans la fiche dans l'ordre de haut en bas (on peut les sélectionner tous et les glisser en même temps, c'est plus professionnel).

6- Insérer de la palette Supplément 5 composants « Speedbutton » et écrire dans la propriété de chacun respectivement :

« Ajouter » « Supprimer » « Modifier » « Engregistrer » « Annuler »

7- double clique sur le « speedbutton1 » (ajouter) et écrire dans sa procédure :

table1.append ;

// table1.append = table1.last + table1.insert ⇒ c'est insérer un nouveau enregistrement en fin de la table.

dbedit1.setfocus ;

8- tapez **F12** et double clique sur le composant « speedbutton2 » (Supprimer) et écrire dans sa procédure :

if messagedlg(' voulez vous supprimer cet enregistrement',mtconfirmation,[mbytes,mbno],0)=mryes then table1.delete ;

9- tapez **F12** et double clique sur le composant « speedbutton3 » (Modifier) et écrire dans sa procédure :

table1.edit ;

dbedit1.setfocus ;

10- tapez **F12** et double clique sur le composant « speedbutton4 » (Enregistrer) et écrire dans sa procédure :

table1.edit ;

table1.post ;

11- tapez **F12** et double clique sur le composant « speedbutton5 » (Annuler) et écrire dans sa procédure :

table1.edit ;

table1.cancel ;

12- Insérer en haut de cette page un composant « DbNavigateur1 » de la palette « ControleBD ».

Ensuite : Choisir dans sa propriété : datasource ⇒ datasource1.

Ensuite : choisir dans sa propriété *VisibleButtons* les boutons *first*, *prior*, *next*, *last*, *refresh* ; mettre visible= true.

13- clique sur l'onglet « Consultation » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	valeur
controlebd	Dbgrid1	Datasource	Datasource1
Supplément	Speedbutton6	Caption	Previsualiser
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Speedbutton7	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose

14- Pour imprimer la liste des professeurs on procède comme suite :

- Dans le menu Delphi on choisit : fichier ⇒ nouveau
- Dans la boite de dialogue qui s'affiche on choisit l'onglet « affaires » et dans ce dernier on choisit l'icône « expert quick report » et clique sur le bouton **Ok.**
- Dans l'assistant qui s'affiche on clique sur le bouton **lancer l'expert.**
- Choisir dans la liste « alias ou répertoire » l'alias « *scolarite* ».
- On choisit dans la liste « nom de table » la table à imprimer (professeur).
- On clique sur le bouton **>>** pour faire passer tous les champs à droite (pour les imprimer tous).
- Ensuite clique sur le bouton **Terminer**

Ensuite : on clique sur cette nouvelle fiche en évitant le clic sur le composant « quickrep1 » et dans sa propriété « name » on donne le nom « *fiche_etat_professeur* » à cette fiche.

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « *etat_professeur* »

15- On tape **Shift+F12** pour afficher la liste des fiches et on choisit la fiche « *fiche_professeur* » et on clique sur **Ok.**

Ensuite : pour programmer la prévisualisation de notre liste on double clique sur le bouton « previsualiser » et on écrit dans sa procédure : **fiche_etat_professeur.quickrep1.preview ;**

Ensuite : on tape **F12** pour retourner à la fiche et pour programmer l'impression de notre liste on double clique sur le bouton « imprimer » et on écrit dans sa procédure : **fiche_etat_professeur.quickrep1.print ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la *fiche_professeur* et la *fiche_etat_professeur* et on doit cliquer sur **Oui** pour cette demande de confirmation.

16- (toujours dans l'onglet consultation) :

Palette	Composant	Propriété	valeur
Standard	Label1	Caption	Filtrage
	Edit1	Text	

17- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange) :

```

if edit1.text <> '' then
begin
table1.close ;
table1.filtered :=true ;
table1.filter :='[nom_p]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;

```

18- On tape la touche **F12** (retour à la fiche), et pour faire le trie de la table professeur : clique sur le composant dbgrid1.

Ensuite : dans son inspecteur d'objet on clique sur l'onglet « événements »

Ensuite : on double clique à droite de l'événement « onclick » et on introduit le code source suivant :

```

If column.index = 0 then
Table1.indexname := 'c1' ;
If column.index = 1 then
Table1.indexname := 'c2' ;
If column.index = 2 then
Table1.indexname := 'c3' ;
If column.index = 3 then
Table1.indexname := 'c4' ;
If column.index = 4 then
Table1.indexname := 'c5' ;

```

Remarque importante :

Le tri et le filtrage dans une table ne réussissent que si on a défini des index secondaires les champs triés ou filtrés.

Pour insérer des index secondaires dans les champs de la table professeur il faut que cette table ne soit pas en cours d'utilisation donc sa propriété : active ⇒ false.

Ensuite : on lance le BDE Delphi par : (le menu Delphi) : outils ⇒ module base de données.

Ensuite : on ouvre la table par : (le menu) fichier ⇒ ouvrir ⇒ table et on choisit l'alias « scolaire » et la table « professeur ».

Ensuite : on clique dans le menu du BDE sur table ⇒ restructurer.

Ensuite : on choisit dans la liste des choix « propriétés de la table » l'option « index secondaires »

Ensuite : on clique pour chaque champ sur le bouton **Définir** et on fait passer ce champ à droite, on clique sur le bouton **OK** et on propose un nom symbolique à cet index secondaire (c1 pour le champ code_p, c2 pour le champ nom_p,...etc.)

Ensuite : on clique sur le bouton **enregistrer** et on retourne à l'environnement delphi.

Enfin : on réactive le composant table1 par sa propriété : active ⇒ true.

19- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément professeur (du sous menu fichier)

Ensuite : on écrit dans la procédure : **fiche_professeur.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_professeur et on doit cliquer sur **oui** pour cette demande de confirmation.

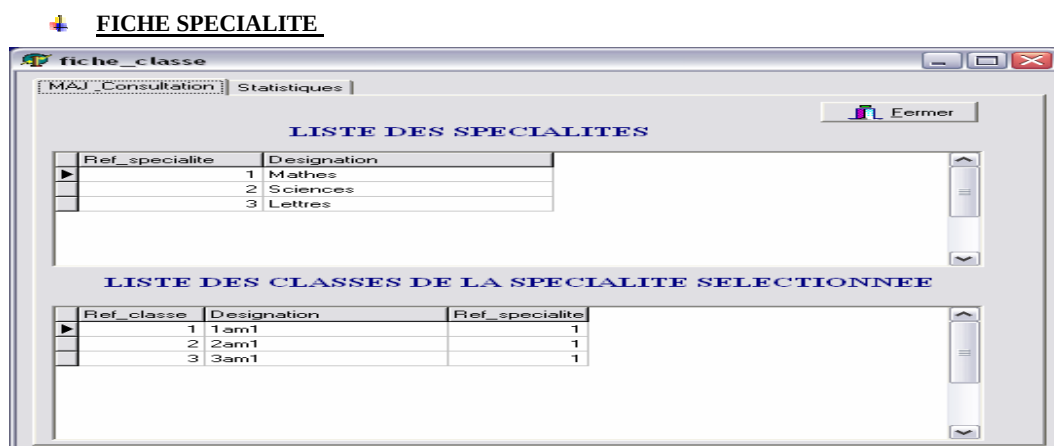
Remarques :

1- Pour que la fiche professeur se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_professeur » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_specialite ».

Ensuite : choisir dans le menu Delphi : fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité : « specialite »

2- Insérer de la palette win32 le composant Pagecontrol1

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « MAJ & Consultation »

Clique droit sur ce composant et clique sur « nouvelle page » et écrire dans sa propriété caption « Statistiques »

3- clique sur l'onglés « MAJ Consultation » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	scolarite
		Tablename	specialite
		Active	True
Accesbd	Datasource1	Dataset	Table1
BDE	Table2	Databasename	scolarite
		Tablename	classe
		Active	True
Accesbd	Datasource2	Dataset	Table2
controlebd	Dbgrid1	Datasource	Datasource1
	Dbgrid2	Datasource	Datasource2
Supplement	Bitbtn1	Kind	Bkclose

4- pour que la table classe soit liée par une relation maître détail avec la table specialite (la table specialite ⇒ maître (père) et la table classe ⇒ détail (fils)) on procède comme suite :

On désactive la table classe (active ⇒ false) et dans le BDE on insère des index secondaires pour chaque champ de cette table (ref_classe ⇒ c1, designation_c ⇒ c2 , ref_specialite ⇒ c3)

Ensuite on retourne à la fiche_specialite et on réactive la table classe (active ⇒ true).

On clique sur le composant table2 « classe », et choisir dans sa propriété : mastersource ⇒ datasource1.

Ensuite : dans sa propriété « masterfield » on clique sur petit bouton .

Ensuite : on choisit dans la liste en haut l'index secondaire « c3 » (c'est l'index du champ ref_specialite de la table classe)

Ensuite : on choisit à gauche le champ «ref_specialite» et à droite le champ « ref_specialite».

Ensuite : clique sur le bouton « ajouter » ensuite sur le bouton .

5- Clique sur l'onglés « Statistiques » du composant « pagecontrol1 » et insérer dans son cadre intérieur les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Query1	databasename	scolarite
		Sql	Select specialite.designation_s, count(ref_classe) as nombre from specialite, classe where specialite.ref_specialite = classe.ref_specialite group by specialite.designation_s
		Active	True
controlebd	DBchart1 (Insérer ce composant sur le composant « pagecontrole1 » dans la page « statistiques ») Et double clique sur ce composant « dbchart1 » Et clique sur l'onglé inférieur « série » Et clique sur le bouton Ajouter Et clique sur le graphe de la forme « sectoriel » Et clique sur le bouton Ok Et clique sur l'onglé supérieur « série » Et clique sur l'onglé « source de données » Et choisir dans la liste l'élément « dataset » Et choisir dans la liste dataset l'élément « query1 » Et choisir dans la liste « libellé » l'élément « specialite.designation_s » Et choisir dans la liste « sectoriel » l'élément « nombre » A la fin clique sur Fermer		

Remarque :

Pour que les statistiques affichent toujours les dernières modifications automatiquement (rafraîchissement automatique des données du query1) : Clique sur le composant table1 (specialite) et Dans son inspecteur d'objet double clique à droite de son événement « afterpost »

Ensuite : écrire le code source suivant dans la procédure correspondante :

Query1.close ;

Query1.active :=true ;

Aussi:

Clique sur le composant table2 (classe) et Dans son inspecteur d'objet double clique à droite de son événement « afterpost »

Ensuite : écrire le code source suivant dans la procédure correspondante :

Query1.close ;

Query1.active :=true ;

6- Cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément specialite (du sous menu fichier)

Ensuite : on écrit dans la procédure : **fiche_specialite.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_specialite et on doit cliquer sur **Oui** .

Remarques :

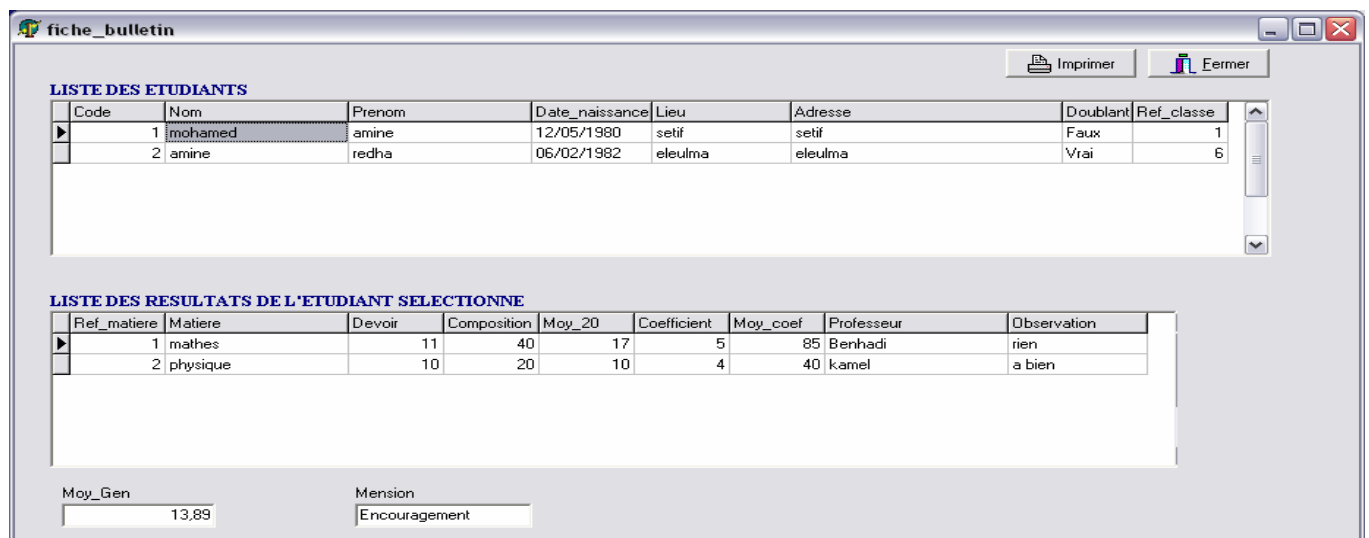
1- Pour que la fiche_specialite se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_specialite » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats)

FICHE BULLETIN



Code	Nom	Prenom	Date_naissance	Lieu	Adresse	Doublant	Ref_classe
1	mohamed	amine	12/05/1980	setif	setif	Faux	1
2	amine	redha	06/02/1982	eleulma	eleulma	Vrai	6

Ref_matiere	Matiere	Devoir	Composition	Moy_20	Coefficient	Moy_coef	Professeur	Observation
1	mathes	11	40	17	5	85	Benhadi	rien
2	physique	10	20	10	4	40	kamel	a bien

Moy_Gen: 13.89 Mention: Encouragement

1- Choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_bulletin».

Ensuite : choisir dans le menu Delphi: fichier⇒enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité: «bulletin»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	Scolarite
		Tablename	Etudiant
		Active	True
accesbd	Datasource1	Dataset	Table1
BDE	Table2	Databasename	Scolarite
		Tablename	Resultat
		Active	True
		mastersource	Datasource1
		masterfield	Clique sur "code_etudiant" à gauche et sur "code_e" à droite et sur Ajouter ensuite sur Ok
accesbd	Datasource2	Dataset	Table2
BDE	Table3	Databasename	Scolarite
		Tablename	Matiere
		Active	True

accesbd	Datasource3	Dataset	Table3
BDE	Table4	Databasename	Scolarite
		Tablename	Professeur
		Active	True
		mastersource	Datasource3
		masterfield	Clique sur "Ref_matiere" à gauche et sur "Reference_m" à droite et sur Ajouter ensuite sur Ok
controlebd	Dbgrid1	Datasource	Datasource1
	Dbgrid2	Datasource	Datasource2
Supplement	Speedbutton1	Caption	Imprimer
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose
BDE	Query1	databasename	Scolarite
		Sql	select sum((devoir+composition)/3*coefficient)/sum(coefficient) as Moy_Gen from matiere,resultat where (ref_matiere =reference_m) and (code_etudiant = :p)
		Params	Clique sur « p » et choisir dans la propriété datatype : « ftinteger »
		Active	True
controlebd	dbnavigator	datasource	Datasource1
standard	Label1	Caption	Resultat
	Edit1	text	

3- on peut renforcer cette fenêtre par les champs calculés comme suite :

On double clique sur le composant table2 « resultat ».

Ensuite : on clique droit sur la petite fenêtre et on clique sur « ajouter tous les champs ».

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « matiere » et on choisit le type du champ : « string » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « moy_20 » et on choisit le type du champ : « float » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « coefficient » et on choisit le type du champ : « integer » et on clique sur .

Ensuite : on clique droit une autre fois sur cette petite fenêtre et on clique sur « nouveau champ ».

Ensuite : on écrit le nom du champ : « moy_coef » et on choisit le type du champ : « Float » et on clique sur .

A la fin : on ferme la petite fenêtre.

Ensuite : on clique sur composant « table2 » (resultat) et dans son inspecteur d'objet on clique sur l'onglet événement.

Ensuite : on double clique à droite de l'événement « oncalcfield » et on écrit dans la procédure :

table3.Open;

table3.setkey;

table3.fieldbyname('reference_m').asinteger:=table2.fieldbyname('ref_matiere').asinteger;

if table3.gotokey then

begin

table2.fieldbyname('matiere').asString:=table3.fieldbyname('designation_m').asString;

table2.fieldbyname('coefficient').asinteger:=table3.fieldbyname('coefficient').asinteger;

table2.fieldbyname('moy_20').asfloat:=

(table2.fieldbyname('devoir').asfloat+table2.fieldbyname('composition').asfloat)/3;

table2.fieldbyname('moy_coef').asfloat:=

table2.fieldbyname('moy_20').asfloat*table2.fieldbyname('coefficient').asinteger;

query1.close;

query1.parambyname('p').asinteger:=table1.fieldbyname('code_e').asinteger;

query1.active:=true;

if (query1.fieldbyname('moy_gen').asfloat >=0) and (query1.fieldbyname('moy_gen').asfloat <5) then

edit1.text:='Blâme';

if (query1.fieldbyname('moy_gen').asfloat >=5) and (query1.fieldbyname('moy_gen').asfloat <7) then

edit1.text:='Avertissement';

if (query1.fieldbyname('moy_gen').asfloat >=7) and (query1.fieldbyname('moy_gen').asfloat <10) then

edit1.text:='Rien';

if (query1.fieldbyname('moy_gen').asfloat >=10) and (query1.fieldbyname('moy_gen').asfloat <12) then

edit1.text:='Tableau d'honneur';

if (query1.fieldbyname('moy_gen').asfloat >=12) and (query1.fieldbyname('moy_gen').asfloat <14) then

edit1.text:='Encouragement';

if (query1.fieldbyname('moy_gen').asfloat >=14) and (query1.fieldbyname('moy_gen').asfloat <=20) then

edit1.text:='Félicitation';

end;

4- On veut ajouter le nom du professeur dans la table résultat; mais, on ne peut pas le faire comme champ calculé puisque une matière a plusieurs enseignants possibles, donc la connaissance (la saisie) de la référence de matière peut nous faire connaître sa désignation et son coefficient; mais on ne peut pas connaître le nom de son enseignant puisque il y a plusieurs possibilités. Dans ce cas et pour plus de souplesse dans la programmation, la méthode de conception Merise accepte de faire des redondances. Dans notre cas on va recréer le champ " nom du professeur " dans la table résultat pour indiquer l'enseignant de chaque matière et cela en procédant comme suite :

- fermer le logiciel par le menu delphi : fichier ⇒ tous fermer; cela assure que la table resultat ne sera pas en cours d'utilisation.
- lancer le BDE Delphi par : (le menu Delphi) : Outils ⇒ Module base de données.
- ouvrir la table resultat par : (le menu BDE) Fichier ⇒ Ouvrir ⇒ table et on choisit l'alias « scolarite » et la table « resultat ».
- cliquer dans le menu du BDE sur : table ⇒ restructurer.
- Ecrire à la fin des champs notre nouveau champ "professeur" avec comme type "A" et taille "20".
- cliquer sur le bouton **enregistrer** et retourner à l'environnement Delphi.
- Réouvrir le logiciel "scolarité" par le menu Delphi : Fichier ⇒ Ouvrir un projet.
- Choisir dans le menu delphi : Fichier ⇒ Nouvelle fiche.
- dans l'inspecteur d'objet de la fiche, écrire dans la propriété « name » le nom : « fiche_liste_professeur».
- choisir dans le menu Delphi: fichier⇒enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité: «liste_professeur».
- insérer dans cette fiche les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	scolarite
		Tablename	professeur
		Active	True
Accesbde	Datasource1	Dataset	Table1
controleBD	Dbgrid1	Datasource	Datasource1

- Pour avoir une ligne de sélection permanente dans le Dbgrid1 qui indique l'enregistrement résultat de la recherche :
 - ☞ Cliquer à gauche de la propriété «  Options » du dbgrid1 (exactement sur la case .
 - ☞ Choisir dans la sous propriété : DGRowSelect ⇒true.
 - ☞ Choisir dans la sous propriété : DGAlwaysShowSelection ⇒true.
- Taper **Shift + F12** et on retourne à la "fiche_bulletin".
- Cliquer sur le composant "dbgrid2" et choisir son événement "onkeydown" et écrire dans sa procédure :

If key = vk_return then

begin

Fiche_liste_professeur.showmodal; // pour afficher la liste des professeurs et en choisir un.

Table2.edit; // préparer la table résultat pour écrire le nom du professeur.

Table2.fieldbyname('professeur').asstring:= Fiche_liste_professeur.Table1.fieldbyname('nom').asstring;

Table2.post;

End ;

5- on tape **F12** pour retourner à la fiche et on double clique sur le composant « query1 ».

Ensuite : clique doit sur la petite fenêtre et clique dans le menu contextuel sur : « ajouter tous les champs ».

Ensuite : glisser le champ « moy_gen » de la requête vers le bas de la fiche, à la fin fermer la petite fenêtre.

6- le « dbgrid2 » affiche le champ « code_etudiant » (de la table resultat) qui contient bien sûr le même code de l'étudiant actuelle (puisque il y'a une liaison maître_détail entre la table resultat et la table etudiant par leurs champs communs code_etudiant et code_e), donc ce champ répètera pour tous les enregistrements du « dbgrid2 » le même code étudiant, il vaut mieux alors de ne pas afficher ce dernier dans le « dbgrid2 ». pour supprimer ce champ :

- double clique sur le « dbgrid2 ».
- clique en haut de la petite fenêtre sur le bouton « ajouter tous les champs »
- supprimer le champ « code_etudiant » et fermer la petite fenêtre.

(Le champ « code_etudiant » ne sera pas supprimé de la table « resultat » mais seulement il ne sera pas affiché)

7- pour préparer l'impression de notre bulletin on procède comme suite :

On insère une nouvelle fiche à notre logiciel par le choix dans le menu Delphi de : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom: « fiche_etat_bulletin ».

Ensuite : (le menu Delphi): fichier ⇒enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité: «etat_bulletin ».

Ensuite : on insère dans cette fiche un composant « quickrep1 » de la palette « qreport ».

Ensuite : on tape **F12** et en haut de l'unité on écrit dans la liste «uses» le nom «bulletin» et on compile **Ctrl+F9** pour assurer la liaison entre la fiche_bulletin et la fiche_etat_bulletin.

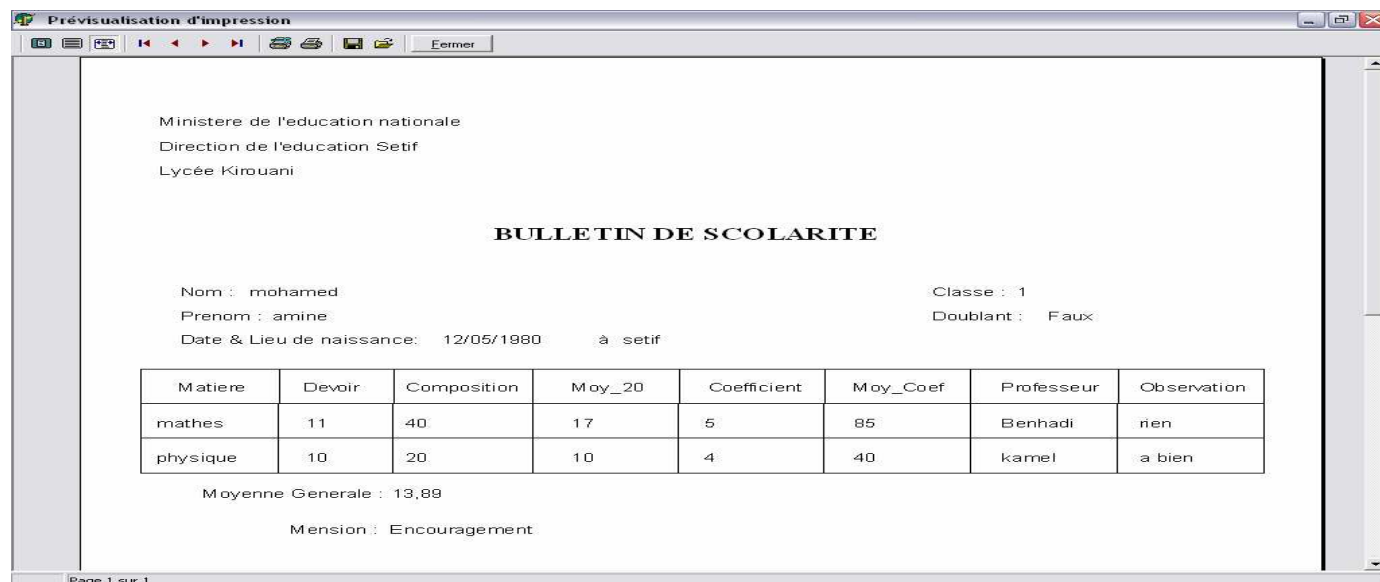
Nb : par fois Delphi répète quelques unités de la partie uses si on écrit une nouvelle unité dans cette partie, donc on doit supprimer les unités dupliquées si Delphi signal ce problème de répétition après la compilation du logiciel.

Ensuite : dans l'inspecteur d'objet du quickrep1 on choisit dans la propriété : dataset ⇒ « fiche bulletin.table2 »

Ensuite : dans l'inspecteur d'objet de ce composant on écrit dans la propriété : zoom $\Rightarrow$ 70

Ensuite : choisir dans sa propriété Bands: `hastitle⇒true`; `hascolumnheader⇒true`; `hasdetail⇒true`; `hassummary⇒ true`.

Ensuite : on insère les composants suivants de la palette « greport » sur les bandes de ce quickrep1 comme suite:



bande	Composant	Propriété	valeur
Titre (agrandir la hauteur de cette bande)	qrlabel1	caption	Ministère de l'éducation nationale
	qrlabel2	caption	direction de l'éducation Sétif
	Qrlabel3	caption	lycée Kirouani
	Qrlabel4	caption	BULLETIN DE SCOLARITE
	Qrlabel5	caption	nom
	qrdbtext1	dataset	fiche_bulletin.table1
		datafield	Nom_e
	Qrlabel6	caption	prenom
	Qrdbtext2	dataset	fiche_bulletin.table1
		datafield	Prenom_e
	Qrlabel7	caption	date & lieu de naissance
	Qrdbtext3	dataset	fiche_bulletin.table1
		datafield	Date_naissance
	Qrlabel8	caption	à
	Qrdbtext4	dataset	fiche_bulletin.table1
		datafield	Lieu
	Qrlabel9	caption	classe
	Qrdbtext5	dataset	fiche_bulletin.table1
		datafield	Ref_classe
	Qrlabel10	caption	doublant
Qrdbtext6	dataset	fiche_bulletin.table1	
	datafield	doublant	
Entête de colonnes	Qrlabel11	caption	Matiere
	Qrlabel12	caption	Devoir
	Qrlabel13	caption	composition
	Qrlabel14	caption	Moy_20
	Qrlabel15	caption	coefficient
	Qrlabel16	caption	Moy_coef
	Qrlabel17	caption	professeur
	Qrlabel18	caption	observation

Detail	Qrdbtext7	dataset	fiche_bulletin.table2
		datafield	Matiere
	Qrdbtext8	dataset	fiche_bulletin.table2
		datafield	Devoir
	Qrdbtext9	dataset	fiche_bulletin.table2
		datafield	composition
	Qrdbtext10	dataset	fiche_bulletin.table2
		datafield	Moy_20
	Qrdbtext11	dataset	fiche_bulletin.table2
		datafield	coefficient
Resumé (agrandir la hauteur de cette bande)	Qrdbtext12	dataset	fiche_bulletin.table2
		datafield	Moy_coef
	Qrdbtext13	dataset	fiche_bulletin.table2
		datafield	professeur
	Qrdbtext14	dataset	fiche_bulletin.table2
		datafield	observation
	Qrlabel19	caption	Moyenne générale
	Qrdbtext15	dataset	fiche_bulletin.query1
		datafield	moy_gen
	Qrlabel20	caption	Resultat
	Qrlabel21	caption	

Ensuite : on tape **Shift + F12** et on affiche la fiche « fiche_bulletin »

Ensuite : on double clique sur le bouton « imprimer » et on écrit dans sa procédure :

Fiche_etat_bulletin.qrlabel21.caption := edit1.text ;

Fiche_etat_bulletin.quickrep1.preview ; // le bouton imprimer existe en haut de la fenêtre de prévisualisation.

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

8- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur .

Ensuite : on double clique sur le composant «mainmenu1» et double clique sur son élément bulletin (sous menu Edition)

Ensuite : on écrit dans la procédure : **fiche_bulletin.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_bulletin et on doit cliquer sur pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_bulletin et la fiche_etat_bulletin et on doit cliquer sur aussi pour cette demande de confirmation.

Remarques :

1- Pour que la fiche bulletin se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_bulletin » et on clique sur .
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

Remarque importante:

Pour contrôler le format d'affichage des nombres réels (champ de type float) dans une BD (limiter le nombre de chiffres après la virgule) on procède comme suite :

- double clique sur le composant « table2 ».
- clique sur le champ à formater (pour nous on va manipuler les deux champs «moy_20» et «moy_coef»)
- Écrire dans sa propriété :

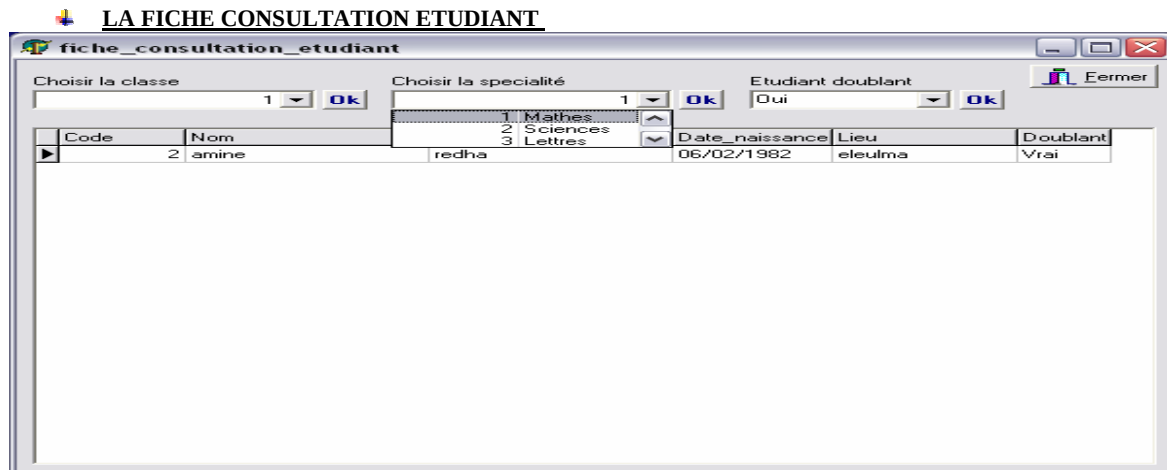
Display format	##.##
----------------	-------

(# Représente un chiffre, donc 2 chiffres avant et après la virgule)

- Ensuite double clique sur le composant « query1 ».
- clique sur le champ à formater (pour nous on va manipuler le champ «moy_gen»)
- Écrire dans sa propriété :

Display format	##.##
----------------	-------

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom : « fiche_consultation_etudiant ».

Ensuite : (le menu Delphi): fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité: « consultation_etudiant »

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	bitbtn	kind	bkclose
standard	label1	Caption	Choisir la classe
BDE	Table1	databasename	scolarite
		tablename	ssecla
		active	true
Accésbd	Datasource1	dataset	Table1
controlebd	dblookupcombobox1	listesource	datasource1
		listefield	Ref_classe; designation_c
		keyfield	Ref_classe
BDE	Query1	databasename	Scolarite
		Sql	select * from etudiant where ref_classe = :p
		Params	Clique sur « p » et choisir dans la propriété datatype : « ftinteger »
		Active	True
Accésbd	Datasource2	dataset	Query1
supplement	Speedbutton1	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
standard	label2	Caption	Choisir la specialite
BDE	Table2	databasename	Scolarite
		tablename	specialite
		active	true
Accésbd	Datasource3	dataset	Table2
controlebd	dblookupcombobox2	listesource	datasource3
		listefield	Ref_specialite; designation_s
		keyfield	Ref_specialite
	Dbgrid1	datasource	Datasource2
supplement	Speedbutton2	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
standard	label3	Caption	Etudiant doublant
	Combobox1	text	
		Items	Oui Non
supplement	Speedbutton3	glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\find.bmp
BDE	Query2	databasename	Scolarite
		Sql	select * from etudiant t1,classe t2 where (t1.ref_classe = t2.ref_classe) and (t2.ref_specialite = :p)
		Params	Clique sur « p » et choisir dans la propriété datatype : « ftinteger »
		Active	True
	Query3	databasename	Scolarite
		Sql	select * from etudiant where doublant = :p
		Params	Clique sur « p » et choisir dans la propriété datatype : « ftstring »
		Active	True

3- double clique sur le composant « speedbutton1 » et écrire :

```
query1.close;
query1.parambyname('p').asinteger:=table1.fieldbyname('ref_classe').asinteger;
query1.active:=true;
datasource2.dataset:=query1;
```

4- tapez **F12** et double clique sur le composant « speedbutton2 » et écrire:

```
query2.close;
query2.parambyname('p').asinteger:=table2.fieldbyname('ref_specialite').asinteger;
query2.active:=true;
datasource2.dataset:=query2;
```

5- tapez **F12** et double clique sur le composant « speedbutton3 » et écrire:

```
query3.close;
query3.parambyname('p').asString:= combobox1.text ;
query3.active:=true;
datasource2.dataset:=query3;
```

6- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **SHIFT + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément étudiant (du sous menu consultation)

Ensuite : on écrit dans la procédure : **fiche_consultation_etudiant.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_consultation_etudiant et on doit cliquer sur **oui** pour cette demande de confirmation.

Remarques :

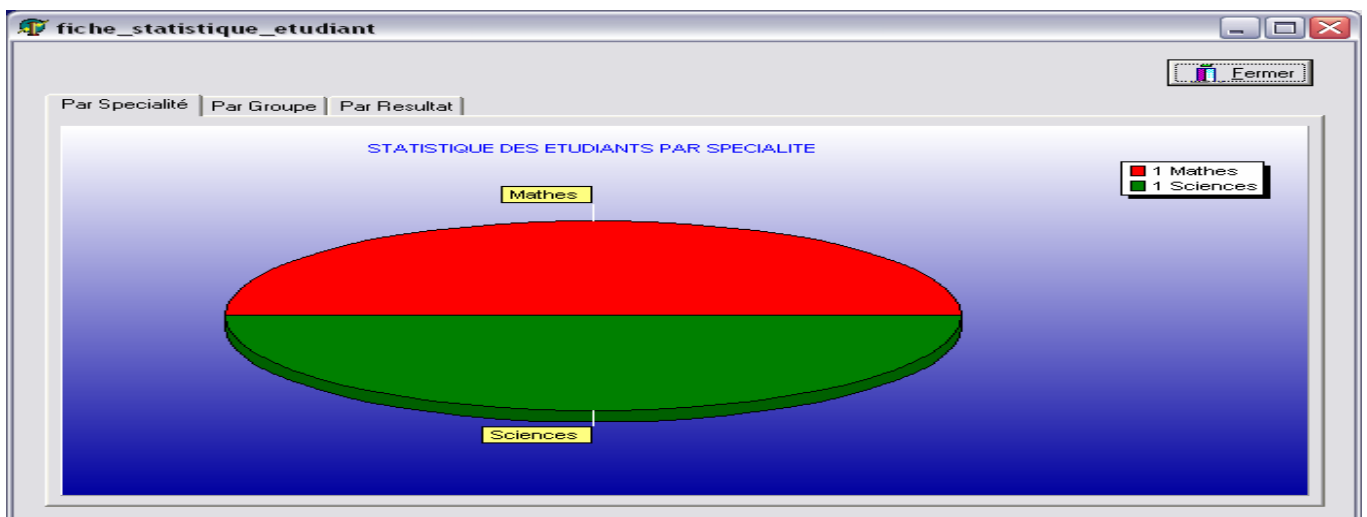
1- Pour que la fiche_consultation_etudiant se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **SHIFT + F12** et on clique sur la « fiche_consultation_etudiant » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

FICHE STATISTIQUES DES ETUDIANTS



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name»le nom:«fiche_statistiques_etudiant».

Ensuite:(le menu Delphi): fichier ⇒enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité:«statistiques_etudiant»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	databasename	scolarite
		tablename	specialite
		Active	true
Accèsbd	Datasource1	Dataset	Table1
BDE	Query1	databasename	scolarite
		Sql	select t1.designation_s,count(t3.code_e)as nombre from specialite t1, classe t2, etudiant t3 where (t1.ref_specialite = t2.ref_specialite) and (t2.ref_classe = t3.ref_classe) group by t1.designation_s
		Active	True
	Query2	databasename	scolarite
		Sql	select t1.designation_c,count(t2.code_e)as nombre from classe t1, etudiant t2 where (t1.ref_classe = t2.ref_classe) and(t1.ref_specialite = :p) group by t1.designation_c
		Params	Clique sur « p » et choisir dans la propriété datatype : « finteger »
	Query3	Active	True
		databasename	scolarite
		Sql	select Doublant,count(Doublant)as nombre from etudiant group by doublant
supplement	btnbit	kind	bkclose
win32	Pagecontrol1		
	Clique droit sur ce composant clique sur « nouvelle page »	etCaption	Par Spécialité
	Clique droit sur ce composant clique sur « nouvelle page »	etCaption	Par Groupe
	Clique droit sur ce composant clique sur « nouvelle page »	etCaption	Par Resultat
Controlebd	Dbgrid1(dans la page« Par Groupe»)	Datasource	Datasource1
	dBchart1 (Insérer ce composant sur le composant « pagecontrol1 » dans la page « Par Spécialité») Et double clique sur ce composant « dBchart1 » Et clique sur l'onglé inférieur « série » Et clique sur le bouton Ajouter Et clique sur le graphe de la forme « sectoriel » Et clique sur le bouton Ok Et clique sur l'onglé supérieur « série » Et clique sur l'onglé « source de données » Et choisir dans la liste l'élément « dataset » Et choisir dans la liste dataset l'élément « query1 » Et choisir dans la liste « libellé » l'élément « désignation_s » Et choisir dans la liste « sectoriel » l'élément « nombre » A la fin clique sur Fermer		
Controlebd	dBchart2 (Insérer ce composant sur le composant « pagecontrol1 » dans la page « Par Groupe»)		
	Et double clique sur ce composant « dBchart2 » Et clique sur l'onglé inférieur « série » Et clique sur le bouton Ajouter Et clique sur le graphe de la forme « sectoriel » Et clique sur le bouton Ok Et clique sur l'onglé supérieur « série » Et clique sur l'onglé « source de données » Et choisir dans la liste l'élément « dataset » Et choisir dans la liste dataset l'élément « query2 » Et choisir dans la liste « libellé » l'élément « désignation_c » Et choisir dans la liste « sectoriel » l'élément « nombre » A la fin clique sur Fermer		

Controlebd	dBchart3 (Insérer ce composant sur le composant « pagecontrole1 » dans la page « Par Resultat») Et double clique sur ce composant « dbchart3 » Et clique sur l'onglé inférieur « série » Et clique sur le bouton Ajouter Et clique sur le graphe de la forme « sectoriel » Et clique sur le bouton Ok Et clique sur l'onglé supérieur « série » Et clique sur l'onglé « source de données » Et choisir dans la liste l'élément « dataset » Et choisir dans la liste dataset l'élément « query3 » Et choisir dans la liste « libellé » l'élément « Doublant » Et choisir dans la liste « sectoriel » l'élément « nombre » A la fin clique sur Fermer
------------	--

3- Dans l'événement "Afterscroll" du composant table1 on écrit :

```
query2.close;
query2.parambyname('p').asinteger:=table1.fieldbyname('ref_specialite').asinteger;
query2.active:=true;
```

Remarque :

Pour que la fiche statistiques etudiant affiche toujours les dernières modifications automatiquement (rafraîchissement automatique des données du query1 et query2 et query3) : Cliquez sur la fiche_statistiques_etudiant

Dans son inspecteur d'objet double clique à droite de l'événement « onactivate »

Ensuite : écrire le code source suivant dans la procédure correspondante :

```
Query1.close ;
Query1.active :=true ;
Query2.close ;
Query2.active :=true ;
Query3.close ;
Query3.active :=true ;
```

4- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément scolarité (sous menu statistiques)

Ensuite : on écrit dans la procédure : **fiche_statistiques_etudiant.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_statistiques_etudiant et on doit cliquer sur **Oui** pour cette demande de confirmation.

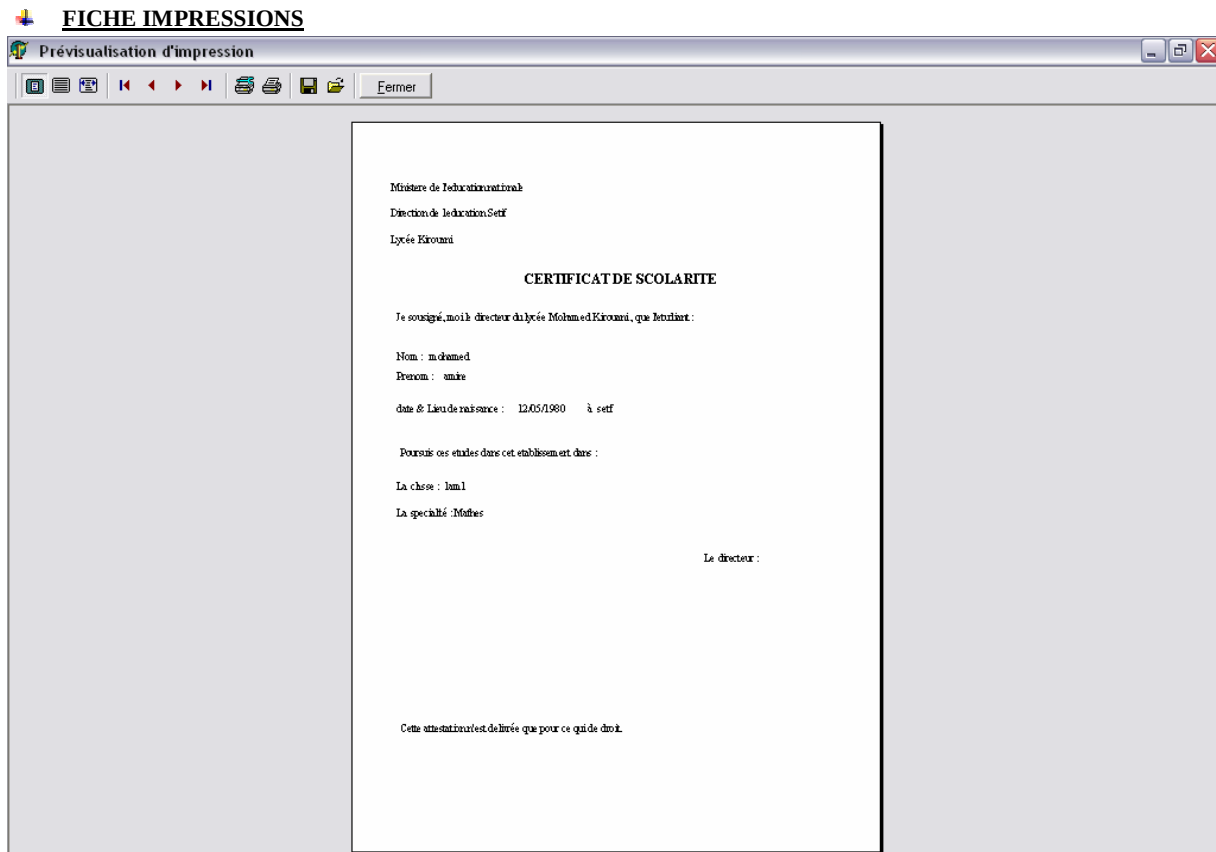
Remarques :

1- Pour que la fiche statistique etudiant se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_statistique_etudiant » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

- ❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name»le nom:«fiche_liste_etudiant».

Ensuite:(le menu Delphi): fichier ⇒ enregistrer, et dans le dossier « c:\scolarité\source » nommer l'unité:« liste_etudiant»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
BDE	Table1	Databasename	scolarité
		Tablename	etudiant
		Active	True
Accesbd	Datasource1	Dataset	Table1
controleBD	Dbgrid1	Datasource	Datasource1
Standard	Label1	Caption	Filtrage
	Edit1	Text	
Supplément	Speedbutton1	Caption	Imprimer certificat scolarite
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Speedbutton2	Caption	Imprimer carte etudiant
		Glyph	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\printer.bmp
	Bitbtn1	Kind	Bkclose

3- Pour faire le filtrage on double clique sur le composant edit1 et on écrit dans la procédure (onchange):

```

if edit1.text <> '' then
begin
table1.close ;
table1.filtered :=true ;
table1.filter :='[Nom_e]=' + #39 + edit1.text + '*' + #39 ;
table1.active := true ;
end
else
begin
table1.close ;
table1.filtered := false ;
table1.active := true ;
end ;

```

4- pour préparer l'impression du **CERTIFICAT DE SCOLARITE** et de la carte etudiant on procède comme suite :

On insère une nouvelle fiche à notre logiciel par le choix dans le menu Delphi de : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété « name » le nom: « fiche_etat_liste_etudiant ».

Ensuite : (le menu Delphi): fichier ⇒ enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité: « etat_liste_etudiant ».

5- pour imprimer le certificat de scolarite on insère dans cette fiche un composant « quickrep1 » de la palette « qreport ».

Ensuite : on tape **F12** et en haut de l'unité on écrit dans la liste « uses » le nom « liste_etudiant » et on compile **Ctrl+F9** pour assurer la liaison entre la fiche_liste_etudiant et la fiche_etat_liste_etudiant.

Nb : par fois Delphi répète quelques unités de la partie uses si on écrit une nouvelle unité dans cette partie, donc on doit supprimer les unités dupliquées si Delphi signal ce problème de répétition après la compilation du logiciel.

Ensuite : dans l'inspecteur d'objet de ce composant on écrit dans la propriété : zoom ⇒ 70

Ensuite : choisir dans sa propriété Bands: hastitle ⇒ true;

Ensuite : on agrandis la largeur de la bande titre et dans cette bande on insère les composants suivants de la palette « qreport » :

bande	Composant	Propriété	valeur
Titre	qlabel1	caption	Ministère de l'éducation nationale
	qlabel2	caption	direction de l'éducation Sétif
	Qlabel3	caption	lycée Kirouani
	Qlabel4	caption	CERTIFICAT DE SCOLARITE
	Qlabel5	caption	J'atteste, moi le directeur du lycée, que l'etudiant (e)
	Qlabel6	caption	nom
	Qrdbtext1	dataset	fiche_liste_etudiant.table1
		datafield	Nom_e
	Qlabel7	caption	prenom
	Qrdbtext2	dataset	fiche_liste_etudiant.table1
		datafield	Prenom_e
	Qlabel8	caption	date & lieu de naissance
	Qrdbtext3	dataset	fiche_liste_etudiant.table1
		datafield	Date_naissance
	Qlabel9	caption	à
	Qrdbtext4	dataset	fiche_liste_etudiant.table1
		datafield	Lieu
	Qlabel10	caption	A suivi ses études cette année scolaire dans :
	Qlabel11	caption	classe
	Qrdbtext5	dataset	fiche_liste_etudiant.table1
		datafield	Ref_classe
	Qlabel12	caption	Cette attestation n'est livrée que pour servir à ce que de droit

Ensuite : on tape **Shift + F12** et on affiche la fiche « fiche_liste_etudiant »

Ensuite : on double clique sur le bouton « imprimer certificat scolarite » et on écrit dans sa procédure :

Fiche_etat_liste_etudiant.quickrep1.preview ;

// le bouton imprimer existe en haut de la fenêtre de prévisualisation.

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

6- pour imprimer la **CARTE ETUDIANT** on insère dans la fiche_etat_liste_etudiant un composant « quickrep2 » de la palette « qreport ».

Ensuite : dans l'inspecteur d'objet de ce composant on écrit dans la propriété : zoom ⇒ 70

Ensuite : choisir dans sa propriété Bands: hastitle ⇒ true;

Ensuite : on double clique sur ce composant « quickrep2 »

Choisir dans la fenêtre la liste : taille de papier ⇒ « taille personnalisée »

Écrire : largeur ⇒ 140 et longueur ⇒ 100 et clique sur **OK**

Ensuite : on agrandis la largeur de la bande titre et dans cette bande on insère les composants suivants :

bande	Composant	Propriété	valeur
Titre	qlabel13	caption	Ministère de l'éducation nationale
	Qlabel14	caption	direction de l'éducation Sétif
	Qlabel15	caption	lycée Kirouani
	Qlabel16	caption	CARTE ETUDIANT
	Qlabel17	caption	nom
	Qrdbtext6	dataset	fiche_liste_etudiant.table1
		datafield	Nom_e
	Qlabel18	caption	prenom

	Qrdbtext7	dataset	fiche_liste_etudiant.table1
		datafield	Prenom_e
	Qrlabel19	caption	date & lieu de naissance
	Qrdbtext8	dataset	fiche_liste_etudiant.table1
		datafield	Date_naissance
	Qrlabel20	caption	à
	Qrdbtext9	dataset	fiche_liste_etudiant.table1
		datafield	Lieu
	Qrlabel21	caption	classe
	Qrdbtext10	dataset	fiche_liste_etudiant.table1
		datafield	Ref_classe
	Qrlabel22	caption	photo
	Qrdbimage1	dataset	fiche_liste_etudiant.table1
		datafield	photo

Ensuite : on tape **Shift + F12** et on affiche la fiche « fiche_liste_etudiant »

Ensuite : on double clique sur le bouton « imprimer carte etudiant » et on écrit dans sa procédure :

Fiche_etat_liste_etudiant.quickrep2.preview ;

// le bouton imprimer existe en haut de la fenêtre de prévisualisation.

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

Remarque importante :

Pour imprimer la certificat de scolarité ou la carte étudiant de l'étudiant en cours (étudiant sélectionné) on ne choisit rien dans la propriété « dataset » du composant « quickrep » (ie : cette propriété doit rester vide), cela aura comme résultat que ces imprimées concerneront à chaque fois l'étudiant sélectionné.

7- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément impressions (du sous menu états)

Ensuite : on écrit dans la procédure : **fiche_liste_etudiant.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_liste_etudiant et on doit cliquer sur **Oui** pour cette demande de confirmation ; on recompile par **Ctrl + F9** pour faire la liaison aussi entre la fiche_liste_etudiant et la fiche_etat_liste_etudiant et on doit cliquer sur **Oui** aussi pour cette demande de confirmation.

Remarques :

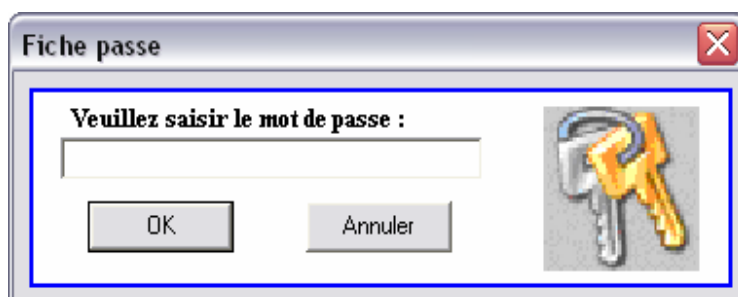
1- Pour que la fiche liste etudiant se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_liste_etudiant » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

FICHE MOT DE PASSE



1-choisir dans le menu Delphi : fichier ⇒ nouveau ⇒ dialogue ⇒ dialogue de mot de passe.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom: « fiche_passe ».

Ensuite : (le menu Delphi) : fichier ⇒enregistrer, et dans le dossier « c:\scolarité\source » nommer l'unité : « passe »

2- insérer un composant « table1 » (palette BDE) et manipuler ces propriétés :

Databasename ⇒ scolarite

table name ⇒ protection

active ⇒ true.

3- Double clique sur le bouton **OK** et écrire dans sa procédure :

```

if password.text=table1.fieldbyname('password').asstring then
fiche_menu.showmodal
else
begin
showmessage('mot de passe incorrecte');
password.clear;
//le composant password est un simple composant Edit; donc password.clear; veut dire vider son texte.
password.setfocus;
end;
```

3- double clique sur le Bouton **Annuler** et écrire dans sa procédure : application.terminate ;

Remarques importantes:

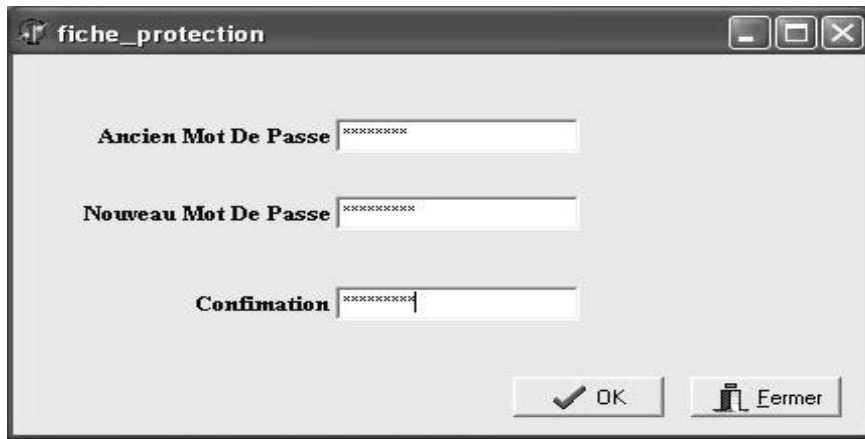
a- Pour que la fiche_passe se lance la première lors de chaque exécution du logiciel on choisit dans le menu Delphi : Projet ⇒ options ⇒ dans l'onglet « fiche » ⇒ dans la liste « fiche principale » ⇒ choisir : « fiche_passe » ⇒ **OK**

b- il faut ajouter les deux unités : messages, dialogs, dans la liste uses de l'unité « passe »

Nb : par fois Delphi répète quelques unités de la partie uses si on écrit une nouvelle unité dans cette partie, donc on doit supprimer les unités dupliquées si Delphi signal ce problème de répétition après la compilation du logiciel.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

FICHE PROTECTION



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name»le nom:« fiche_protection ».

Ensuite : (le menu Delphi) : fichier ⇒enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité:« protection»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	bitbtn1	kind	bkclose
	bitbtn2	kind	bkok
standard	label1	caption	Ancien mot de passe
	edit1	text	
		passwordchar	*
	label2	caption	Nouveau mot de passe
	edit2	text	
		passwordchar	*
	label3	caption	confirmation
	edit3	text	
		passwordchar	*
BDE	table1	databasename	scolarite
		tablename	protection
		Active	True

3- double clique sur le composant « BitBtn2 » et écrire dans sa procédure:

```

If edit1.text = table1.fieldbyname('password').asstring then
If edit3.text = edit2.text then
Begin
Table1.edit;
Table1.fieldbyname('password').asstring:=edit2.text;
Table1.post;
Showmessage(' la modification du mot de passe est effectuée ')
End
Else Showmessage(' la confirmation du mot de passe est incorrecte ')
Else Showmessage(' l'ancien mot de passe est incorrecte ');

```

4- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément protection (du sous menu outils)

Ensuite : on écrit dans la procédure : **fiche_protection.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_protection et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

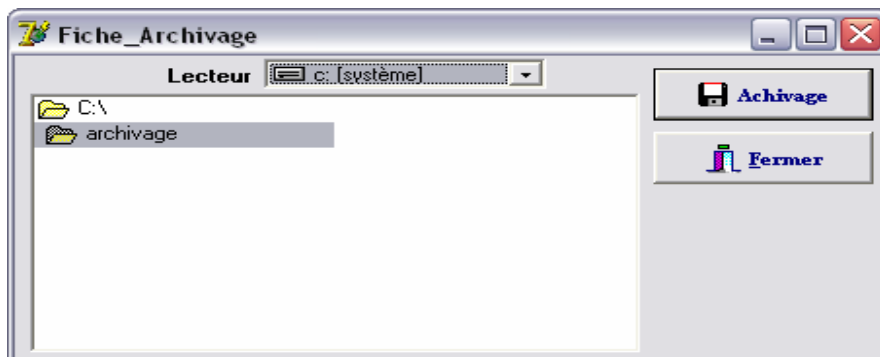
1- Pour que la fiche protection se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_protection » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

FICHE ARCHIVAGE



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom:« fiche_archivage ».

Ensuite : (le menu Delphi) : fichier ⇒enregistrer, et dans le dossier « c:\scolarité\source » nommer l'unité:« archivage»

Palette	Composant	Propriété	Valeur
win3.1	FileListBox1	name	fichier
		Mask	*.db
	DirectoryListBox1	name	driverlist
	DriveComboBox1	name	driver
		Dirlist	driverlist
supplement	FilterComboBox1	Filelist	fichier
		filter	*.db
	BitBtn1	kind	bkclos
standard	Speedbutton1	caption	archiver
		glyph	C:\Program Files\Fichiers communs\Borland shared\Images\Buttons\floppy.bmp
standard	Label1	caption	lecteur

2- tapez **F12** et dans la liste uses de l'unité archivage Il faut ajouter l'unité : ShellAPI, .

3- tapez **F12** pour revenir à la fiche et Insérer les composants suivants :

4- Double clique sur le composant speedbutton1 et écrire dans sa procédure :

```
if CopyFolder('c:\scolaritelbase', driverlist.Directory) then
  ShowMessage('Archivage effectué avec succès') else
  ShowMessage('Archivage non effectué');
```

5- il faut inserer avant la procedure BitBtn4Click la fonction suivante :

```
function CopyFolder(FromFld, ToFld: string): boolean;
var fos: TSHFileopStruct;
begin
  FillChar(fos, SizeOf(fos),0);
  with fos do
    begin
      wFunc := FO_COPY;
      pFrom := PChar(FromFld+#0);
      pTo := PChar(ToFld+#0);
      fFlags := FOF_SILENT or FOF_NOCONFIRMATION or FOF_NOCONFIRMMKDIR;
    end;
  Result := ShFileOperation(fos)=0;
end;
```

6- pour avoir un bon interface on doit cacher les deux composants «FileListBox1» et «FilterComboBox1»comme suite:

- clique par le bouton droit de la souris sur le composant : « FileListBox1 » nommé « fichier » et choisir dans le menu contextuel : « mettre en arrière plan » et mettre ce composant au-dessous du « DirectoryListBox1 » (driverlist).

- aussi, clique par le bouton droit de la souris sur le composant : « FilterComboBox1 » et choisir dans le menu contextuel : « mettre en arrière plan » et mettre ce composant au-dessous du « DirectoryListBox1 » (driverlist).

7- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément archivage (du sous menu outils)

Ensuite : on écrit dans la procédure : **fiche_archivage.showmodal** ;

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_archivage et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

1- Pour que la fiche archivage se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

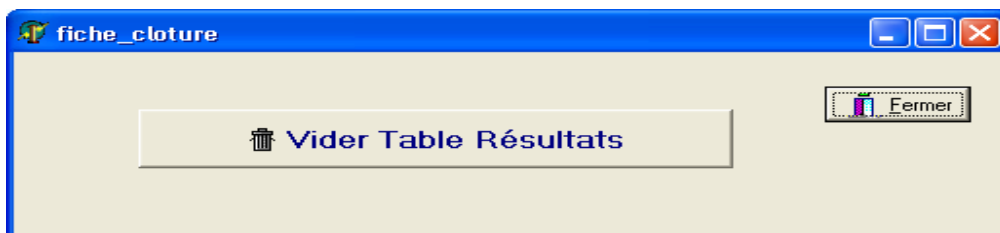
- On tape : **Shift + F12** et on clique sur la « fiche_archivage » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter»

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

❖ Cette fiche nécessite la création d'un dossier dans disque dur (nommé par exemple : archive2007) qui sera le dossier où on va archiver notre base de données.

FICHE CLOTURE



1- choisir dans le menu Delphi : fichier ⇒ nouvelle fiche.

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom:« fiche_Clature».

Ensuite : (le menu Delphi) : fichier ⇒enregistrer, et dans le dossier « c:\scolaritel\source » nommer l'unité:« Clature»

2- Insérer les composants suivants :

Palette	Composant	Propriété	Valeur
supplement	speedbutton1	caption	Vider Table Resultats
		lyphgl	C:\Program Files\Fichiers communs\Borland Shared\Images\Buttons\trash.bmp
	Bitbtn1	kind	bkclos
BDE	Table1	Databasename	scolarite
		Tablename	resultat
		Active	True
controleBD	Dbgrid1	Datasource	Datasource1

3- double clique sur le « speedbutton1 » et écrire dans sa procédure :

```
While not (table1.eof) do
  table1.delete ;
  showmessage('la table resultats est vidée');
```

En fin : on enregistre tous le travail par : (le menu Delphi) : fichier ⇒ tout enregistrer

2- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément Cloture (du sous menu outils)

Ensuite : on écrit dans la procédure : **fiche_cloture.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_cloture et on doit cliquer sur **Oui** pour cette demande de confirmation.

Remarques :

1- Pour que la fiche cloture se place au milieu de l'écran si on l'appelle lors du lancement de l'exécution du logiciel :

- On tape : **Shift + F12** et on clique sur la « fiche_cloture » et on clique sur **Ok**.
- On clique sur cette fiche (éviter de cliquer sur l'un de ses composants).
- On choisit dans sa propriété : « position ⇒ poScreenCenter »

2- on enregistre notre travail par le menu Delphi : fichier ⇒ tout enregistrer.

❖ Lancer l'exécution par **F9** et tester cette fiche (saisir des données et vérifier les résultats de tous les traitements possibles)

FICHE A PROPOS



1- on peut insérer cette fiche par le menu Delphi :

Fichier ⇒ nouveau ⇒ fiche ⇒ boîte à propos ⇒ **Ok**

Ensuite : dans l'inspecteur d'objet de la fiche on écrit dans la propriété «name» le nom:«fiche_apropos».

Ensuite : (le menu Delphi): fichier⇒enregistrer, et dans le dossier « c:\scolarite\source » nommer l'unité:« a propos »

Ensuite : on écrit le texte qu'on veut dans les labels et modifier la couleur et la taille des labels par leurs propriété « FONT ». (Cette fiche est la carte visite du logiciel).

2- cette fiche est maintenant prête à être liée avec le menu principal du logiciel comme suite :

On tape les touches : **Shift + F12** et on clique sur la « fiche_menu » et on clique sur **Ok**.

Ensuite : on double clique sur le composant « mainmenu1 » et double clique sur son élément « a propos » (du sous menu « ? »)

Ensuite : on écrit dans la procédure : **fiche_apropos.showmodal ;**

Ensuite : on compile notre logiciel par les touches **Ctrl + F9** et automatiquement Delphi vous demande la confirmation de la liaison entre la fiche_menu et la fiche_apropos et on doit cliquer sur ☐ pour cette demande de confirmation.

L'image de Logo :

1- dans la propriété picture du composant « image1 » on clique sur le bouton et chercher l'image voulue.

(Delphi propose quelques images dans :

« C:\Program Files\Fichiers communs\Borland Shared\Images\Splash\256Color ».

2- en fin clique sur ensuite sur

3- pour que l'image occupe toute la surface de son cadre on modifie sa propriété : Autosize ⇒ true.

4- pour que l'image s'agrandisse si elle a une taille plus petite que la surface de son cadre on modifie sa propriété : Stretch ⇒ true.

❖ Lancer l'exécution par **F9** et tester cette fiche (vérifier les résultats de tous les traitements possibles)

CHAPITRE 4

TRUCS

&

ASTUCES

CRÉER UNE INSTALLATION

Lorsque vous commencerez à faire des logiciels assez grand, vous aurez sûrement envie de créer **un programme d'installation** (pour tous type de logiciels, qu'il soit réalisé en Delphi, en C++ Builder, en Java...)

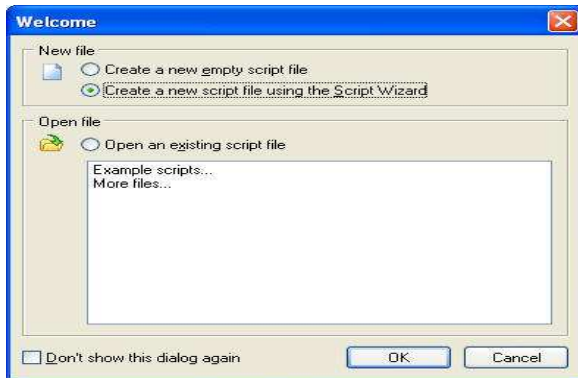
On va montrer comment créer un programme d'installation pour le Logiciel Gestion Commerciale :

1- Télécharger Inno Setup

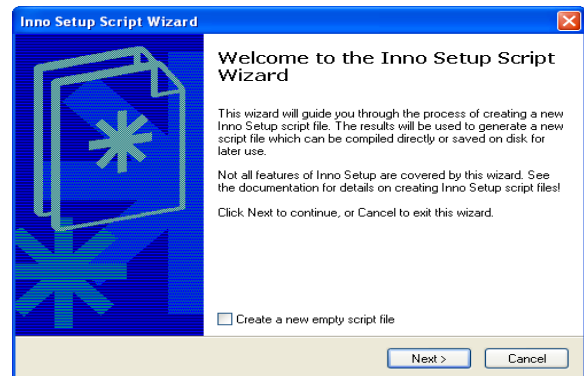
Il existe de nombreux outils permettant de créer une installation : **InstallShield** , **Inno Setup** ...

2- Créer une nouvelle installation

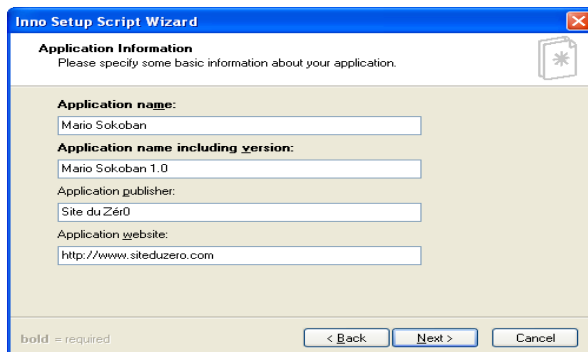
Cochez l'option : "Create a new script file using the Script Wizard" :



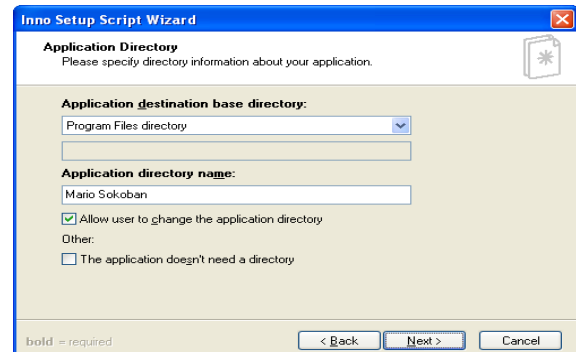
Cliquez sur OK.



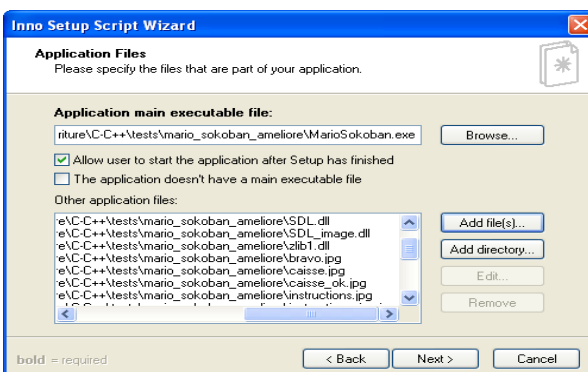
Ne cochez pas la case, cliquez sur Next,



Vous devez rentrer le nom de votre programme, le nom de votre programme avec le numéro de version, le nom du créateur ainsi que le site web du programme.



Choisir le dossier d'installation du programme. Laisser cocher la case "Allow user to change the application directory". Cela permettra à l'utilisateur de changer le chemin d'installation.



les fichiers à emballer.

le chemin de l'exécutable (le .exe du logiciel).

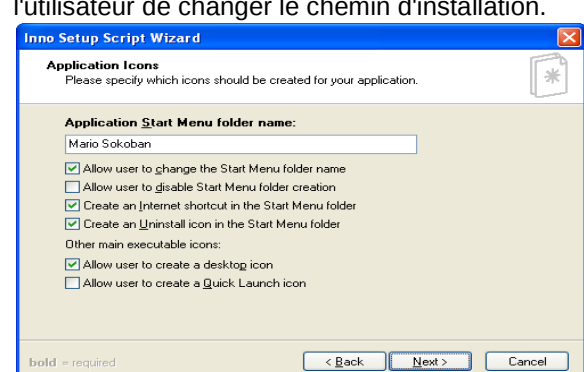
Donc : c:\commerce\source\commerce.exe

en-dessous, laisser cochée la case :

elle permet de laisser la possibilité à l'utilisateur de démarrer le programme automatiquement à la fin de l'installation.

La case "The application doesn't have a main executable file" ne sera généralement pas cochée. Elle ne sert que pour les programmes ne possédant pas de .exe principal.

les "Other application files". Vous devez indiquer **tous**



On vous demande le nom du dossier dans le menu démarrer.

les fichiers dont a besoin votre programme pour fonctionner.

(les DLL dont a besoin le programme, les images, les sons etc...)

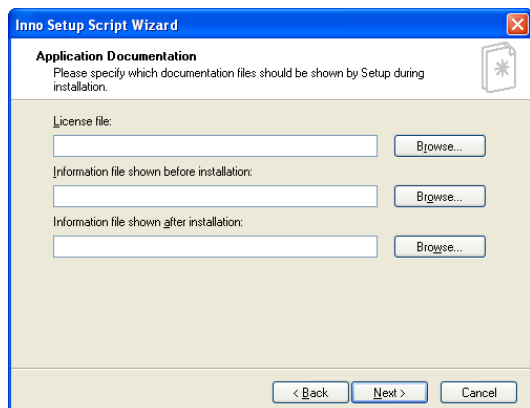
Allow user to change Start Menu folder name : l'utilisateur peut changer le nom du dossier du menu démarrer.

Allow user to disable Start Menu folder creation : l'utilisateur peut désactiver la création des raccourcis dans le menu démarrer.

Create an Internet Shortcut in the Start Menu folder : un lien vers votre site web sera ajouté au Menu Démarrer

Create an Uninstall icon in the Start Menu folder : ajoute une icône de désinstallation dans le menu démarrer.

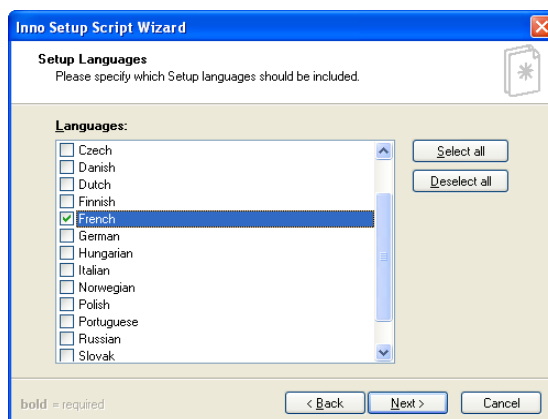
Allow user to create a desktop icon : laisse la possibilité à l'utilisateur de créer un raccourci sur le bureau. **Allow user to create a Quick Launch icon** : l'utilisateur peut créer un raccourci dans la barre Quick Launch. C'est une barre de raccourcis située juste à droite du menu Démarrer.



Les fichiers texte à afficher avant et après l'installation (ainsi que la licence du programme).

Vous pouvez indiquer n'importe quel fichier .txt (ou .rtf si vous voulez faire un peu de mise en forme comme mettre de la couleur, du gras...). exemple, vous pourriez indiquer les bugs connus de votre programme, les améliorations apportées par la nouvelle version etc.

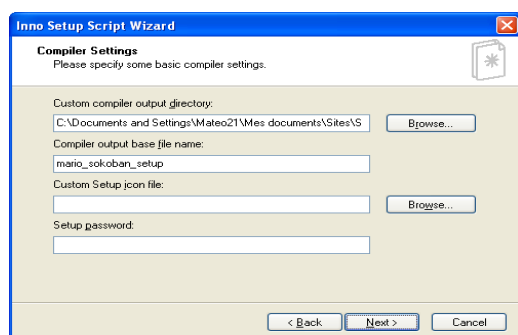
La partie "License File" sera utile notamment si vous distribuez votre programme sous licence libre (GNU / GPL) comme ça se fait le plus souvent pour les programmes Open Source.



Les langues disponibles dans le programme d'installation.

Si vous cochez plusieurs langues, on demandera la langue désirée au début de l'installation.

Par



Le premier champ permet d'indiquer dans quel dossier devra être créé le programme d'installation.

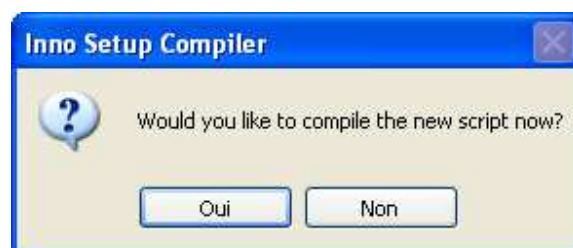
Ensuite, on vous demande le nom du programme d'installation.

Le troisième champ permet de choisir un fichier d'icône (.ico) personnalisé pour l'installation.

Le 4ème champ permet de protéger l'installation par mot de passe. Seuls ceux qui connaissent le mot de passe pourront installer votre programme.

La fenêtre suivante est la dernière :

vous n'avez plus qu'à cliquer sur Finish !



Compiler l'installation, cliquez sur Oui
Au bout de quelques secondes,
le programme d'installation a été généré !

CONCEPTION DES FICHIERS HELP POUR LES LOGICIELS

On procède en 3 étapes :

I. DANS L'EDITEUR DE TEXTE MICROSOFT WORD :

1- On insère le nombre de pages nécessaire à notre fichier d'aide (à savoir le nombre de fiches du logiciel qu'on veut expliquer).

L'insertion d'une nouvelle page se fait comme suite :

Choisir dans le menu Word : insertion ⇒ saut ⇒  ⇒ .

2- On écrit en suite dans chaque page l'explication du fonctionnement d'une fiche du logiciel.

3- En haut de chaque page il faut écrire un mot clé qui va être indexé ensuite par le fichier d'aide, l'indexation se fait par l'insertion du symbole # pour indexer une page d'explication (appelée Map) attribuée à une fiche ou un message de dialogue.

Donc on suit les étapes suivantes :

- On pose le curseur devant le mot clé d'une page choisie :

Exemple : pour la page d'explication de la fiche Client on écrit dans la première ligne de la page le titre : **Client** ensuite on met le curseur devant ce mot.

- Dans le menu on choisit : insertion ⇒ note de bas de page.

Une boîte de dialogue s'affiche ou on écrit dans sa zone de texte « personnalisé » le symbole # et on ferme cette boîte par .

Le curseur se pointe automatiquement en bas de page précédé par le symbole choisi (#), on écrit le mot identificateur de cette page d'aide (généralement ça sera le mot clé de la page, donc pour notre exemple on écrit **Client**).

- On répète ce procédé pour toutes les pages du fichier d'aide.

4- On finit l'opération en enregistrant ce fichier Word avec format RTF (rich text format) (format text riche).

II. DANS LE COMPILATEUR HCW :

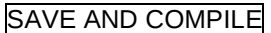
1- Dans cette étape on doit lancer le compilateur HCW.EXE proposé par l'environnement Delphi pour formater et compiler notre fichier Help (chemin de ce fichier : c:\program files\borland\delphi7\help\tools).

2- On choisit dans son menu : file ⇒ new et on choisit le type du fichier Help (pour nous on choisit : Help Project) Et on donne un nom à ce projet (l'extension sera par défaut HLP).

3- Une fenêtre d'édition du Help s'affiche et on manipule les boutons à droite comme suite :

- Clique sur le bouton  ou on se pointe vers le fichier RTF contenant le texte d'aide créé en Word et cela en cliquant sur le bouton  (possibilité d'addition de plusieurs fichiers d'aide RFT).

Clique sur le bouton  ou on va indexer les pages d'explication des fiches et messages de dialogue, et cela en cliquant pour chaque Map sur le bouton  et on insère dans la zone « Topic ID » le mot clé du Map (l'un des mots clés d'indexation écrit dans les bas de pages en Word qu'on les a précédé par les symboles #. Pour notre page d'exemple on écrira : Client) ; et dans la zone « Mapped numeric value » on insère un numéro de cette page d'aide (un nombre entier >=1).

- En fin on compile notre fichier en cliquant sur le bouton :  et notre fichier Help final sera généré (il faut que le compilateur indique 0 notes (pas d'erreurs de compilation)).

III. DANS NOTRE LOGICIEL DELPHI :

1- On fait la liaison entre notre logiciel et notre fichier Help en insérant la commande suivante dans l'événement « onCreate » du menu principal :

application.helpfile := 'c:\mes documents\Tp.hlp' ;

{En supposant que le fichier Help été enregistré sous le nom Tp.hlp dans le dossier mes documents}.

2- Pour lier un bouton d'aide à une page d'aide (Map) on écrit dans son événement « OnClick » :

application.helpcommand(help_context,5) ;

//5 est le numéro de la page d'aide (Map) liée à ce bouton d'aide

Cette instruction peut être utilisée aussi dans le menu : ? ⇒ Aide.

3- pour lier une fiche à sa page help correspondante :

- On clique sur cette fiche en évitant tous ses composants.
- On écrit dans sa propriété : « helpcontext » le numéro de sa page d'aide (exemple 10), cette page (10) s'affichera lors de l'exécution du logiciel si on tape dans cette fiche sur la touche **F1**.
- Aussi on modifie la propriété KeyPreview ⇒ true.

4- Pour lier un message dialogue à une page d'aide (Map) on écrit ce message comme suite :

MessageDlg('je vous propose un Help', MTInformation, [MBKO], 25) ;

{25 est le numéro de page d'aide (Map) liée à ce message de dialogue}

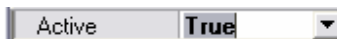
SKIN DATA

C'est un utilitaire qui aide à modifier l'interface d'un logiciel (améliorer la partie visuelle du logiciel)

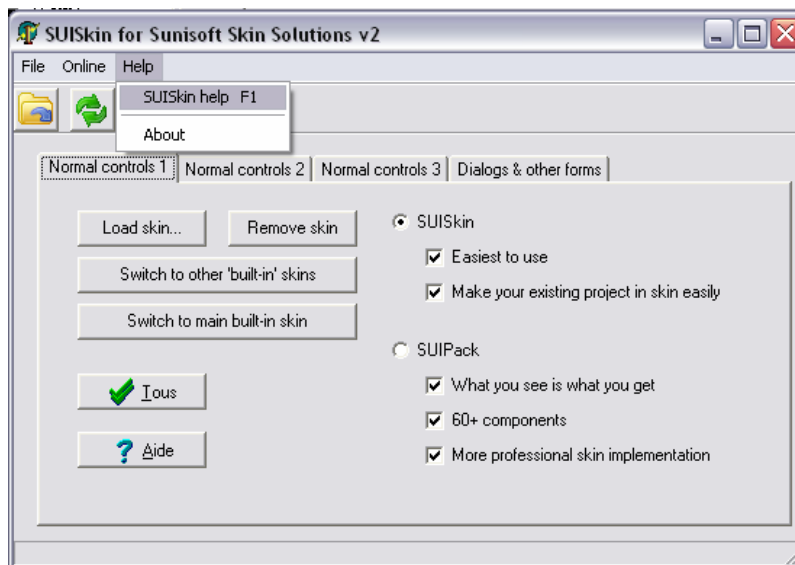
- on doit avant télécharger ce composant d'un site web quelconques (plusieurs sites de développement proposent ce composant avec une grande variété d'interfaces possible).
- Après on doit installer ce composant pour que Delphi puisse l'utiliser (après l'installation de ce composant, apparaît à la fin de la palette  des composants comme un objet à insérer dans une fiche de votre application).
- Il vaut mieux insérer le composant skindata dans la première fiche lancée du logiciel qui sera généralement la fiche de mot de passe, cela assure la modification de l'interface de tous les autres fiches suivantes.
- Après l'insertion de ce composant on choisit le modèle d'interface voulu dans sa propriété

 SkinFile |  ← Clic Ici

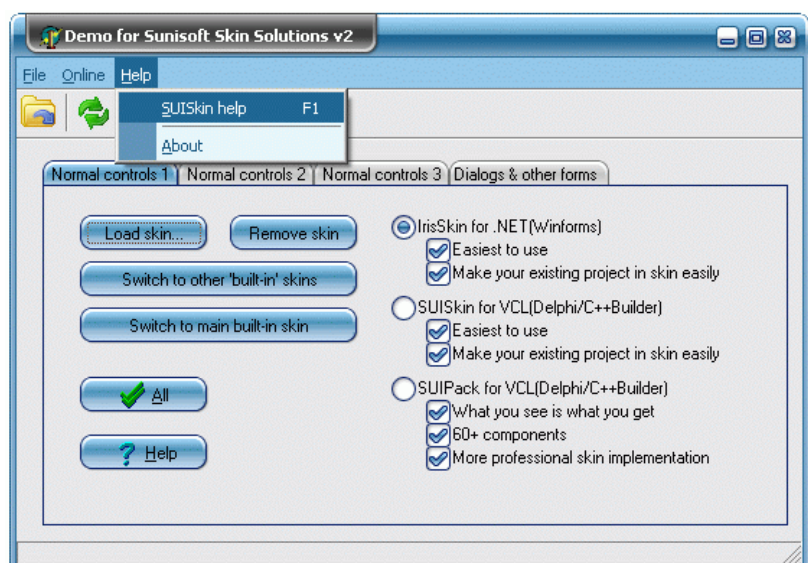
(on trouve tous les modèles proposés par skindata dans le dossier skins du chemin d'installation de skindata)

- On suit on choisit dans la propriété :  Active **True**

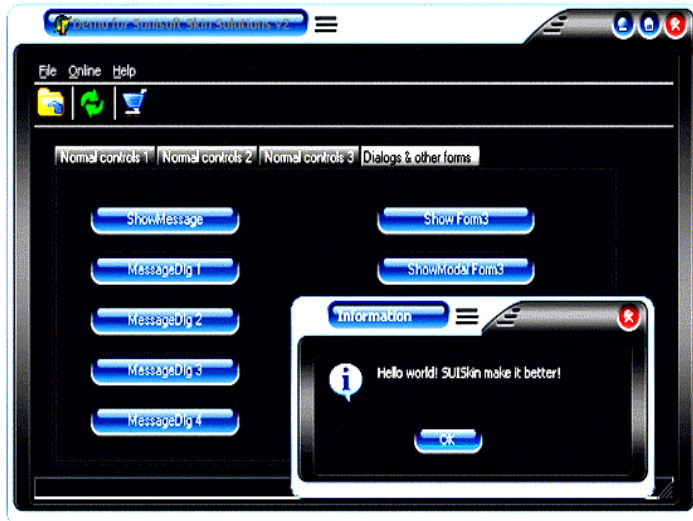
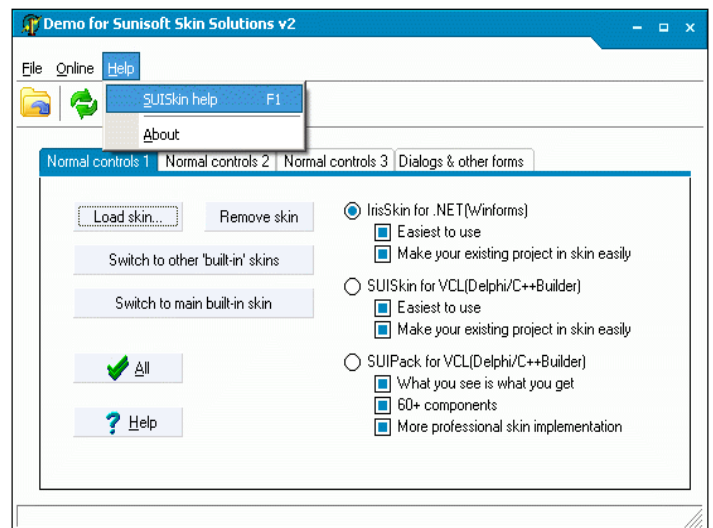
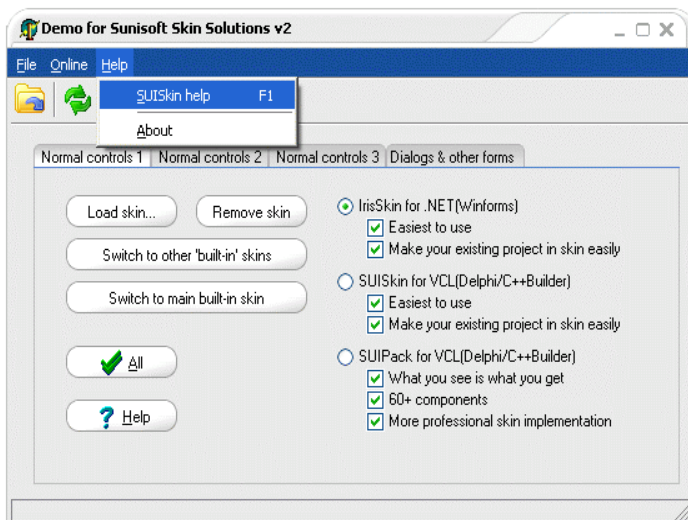
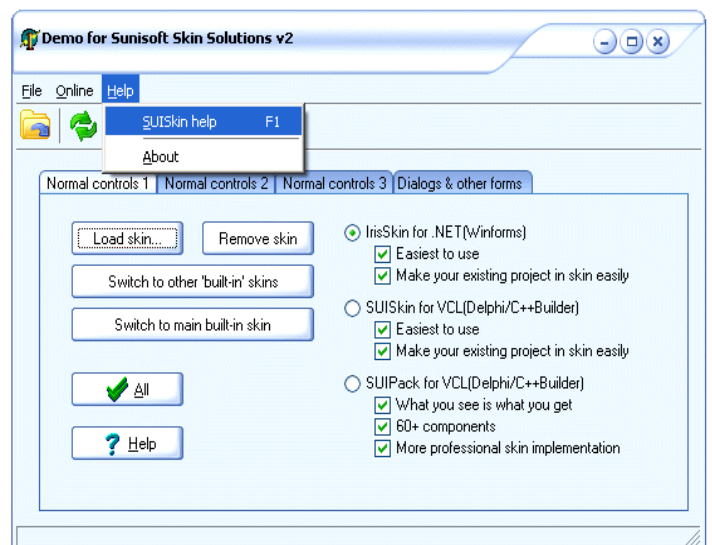
On lance l'exécution et tout le logiciel changera d'interface.



La forme de la fiche après avoir utilisé le composant SKINDATA



QUELQUES MODÈLES

Une fiche avec le
modele STEELUne fiche avec le
modele DEEPUne fiche avec le
modele RealOneUne fiche avec le
modele MSN

ANNEXE

LES COMPOSANTS DELPHI LES PLUS UTILISÉS

COMPOSANT		PROPRIETE	EVENEMENTS
STANDARD			
MAINMENU Création d'un menu principal		<u>Items</u> : (ou bien double clique sur le MainMenu) pour afficher l'éditeur du menu principal. <u>Images</u> : pour lier un composant image (avec sa gamme des icônes) au MainMenu. Pour chaque élément du menu : <u>Caption</u> : pour donner un nom à l'élément. <u>Imageindex</u> : pour lier une image à cet élément. <u>Shortcut</u> : affecter un raccourci clavier à cet élément.	<u>OnClick</u> : pour chaque élément du menu principal on introduit dans cet événement le code source adéquat.
POPUPMENU Création d'un menu turbo		<u>Items</u> : (ou double clique sur le PopupMenu) pour afficher l'éditeur du menu turbo. <u>Images</u> : pour lier un composant image (avec sa gamme des icônes) aux PopupMenu. Pour chaque élément du menu : <u>Caption</u> : pour donner un nom à l'élément. <u>Imageindex</u> : pour lier une image à cet élément. <u>Shortcut</u> : affecter un raccourci clavier à cet élément.	<u>OnClick</u> : pour chaque élément du menu turbo on introduit dans cet événement le code source adéquat.
LABEL Éclaircir l'interface		<u>Caption</u> : pour donner un nom d'interface au composant. <u>Font</u> : pour manipuler le format du texte.	<u>OnClick</u> <u>Onmousemove</u> .
EDIT Zone de saisie		<u>Font</u> : pour manipuler le format du texte. <u>Text</u> : représente le texte saisi (propriété à manipuler)	<u>OnChange</u> <u>Onkeydown</u> .
BUTTON Bouton d'action encapsulant un code source		<u>Caption</u> : pour donner un nom d'interface au bouton <u>Font</u> : pour manipuler le format du texte. <u>Hint</u> : pour introduire le texte d'aide. <u>Showhint</u> : pour afficher le texte d'aide. <u>PopupMenu</u> : assure la liaison avec un PopupMenu.	<u>OnClick</u> <u>Onkeydown</u> <u>Onmousemove</u>
CHECKBOX Case à cocher (choix libre)		<u>Caption</u> : pour donner un texte d'interface au bouton <u>Checked</u> : contrôle l'état (cocher ou décocher)	<u>OnClick</u> <u>Onmousemove</u> .
RADIOBUTTON Case à option (choix exclusif)		<u>Caption</u> : pour introduire le texte d'interface. <u>Checked</u> : pour voir si le composant est coché ou non	<u>OnClick</u> <u>Onmousemove</u> .
COMBOBOX Liste déroulante de choix		<u>Items</u> : pour introduire les éléments de la liste ligne par ligne. <u>Text</u> : contient la valeur choisie parmi celles de la liste	<u>OnChange</u> <u>OnClick</u> <u>Onkeydown</u> .
SUPPLEMENT			
BITBTN Bouton autoprogrammé		<u>Kind</u> : choisir le comportement du bouton. (bkclose pour fermer une fiche)	<u>OnClick</u> <u>Onkeydown</u> <u>Onmousemove</u>
SPEEDBUTTON Bouton d'action favorisé		<u>Gliph</u> : affecter une image (icône) à ce bouton. <u>Hint</u> : introduire un texte d'aide au bouton. <u>Showhint</u> : afficher le texte d'aide	<u>OnClick</u> <u>Onmousemove</u>
IMAGE Image d'arrière plan ou de logo		<u>Stretch</u> : la taille de l'image sera égale la taille du cadre <u>Autosize</u> : agrandir l'image à la taille de son cadre. <u>Align</u> : l'image occupera tous l'espace de la fenêtre (alclient) <u>Picture</u> : choisir une image dans le disque dur.	

WIN32			
PAGECONTROL Propose des pages à onglets		Clique par le bouton droite de la souris sur le composant et choisir dans le menu turbo « nouvelle page ».on clique sur le cadre intérieur de chaque page « tabsheet » et on introduit dans sa propriété « caption » le nom de la page.	Onchange Onmousemove
IMAGELIST Liste d'icônes à introduire dans un menu ou une barre d'outils.		Double clique sur le composant et dans la boîte de dialogue qui s'affiche on clique sur le bouton «ajouter » pour insérer une nouvelle image à la gamme déjà choisie. à la fin il faut supprimer le double flou de chaque image par le bouton « supprimer ».	Onchange
ANIMATE Propose des effets d'animation		<u>CommonAVI</u> : cette propriété propose une liste des effets d'animation. <u>Active</u> : cette propriété provoque le lancement l'animation (active :=true)	Onstart Onstop
DATETIMEPICKER Pointeur vers une date		Ce composant contrôle la date sous forme d'un calendrier, la date choisie sera dans sa propriété « date ».	Onchange OnClick Onkeydown.
TOOLBAR Création d'une barre d'outils		Ce composant se place automatiquement dans la partie haute (entête) de la fiche, mais on peut manipuler sa position par la propriété « Align ». Pour ajouter un nouveau bouton on clique par le bouton droite de la souris sur ce composant et on choisit sur le menu turbo l'élément « nouveau bouton », à la fin on fait la liaison entre une gamme d'images préparée dans le composant «imagelist» et la barre d'outils par sa propriété «images»	OnClick OnMouseMove.
SYSTEME			
TIMER Boucle chronologique		Ce composant assure la répétition d'une action dans un intervalle de temps fixé par la propriété « interval ». Par le double clic sur ce composant on pourra créer une procédure « Ontimer » qui contiendra les instructions à répéter consécutivement.	OnTimer
MEDIAPLAYER Animation multimédia		<u>Filename</u> : cette propriété assure la liaison entre ce composant et un fichier vidéo, audio.... Qu'on va utiliser. <u>Devicetype</u> : représente le type du fichier multimédia à manipuler. L'instruction qui déclenche l'effet multimédia est: MediaPlayer1.play ;	OnClick
ACCESBD			
DATASOURCE Composant intermédiaire d'accès BD		<u>Dataset</u> : représente la source de données (table, requête SQL ...) liée avec ce composant.	OnChange OnUpdateData
BDE			
TABLE Composant d'accès à une table d'une BD		<u>Databasename</u> : représente l'alias de la table <u>Tablename</u> : la table à manipuler. <u>Name</u> : le nom de programmation de la table. <u>Active</u> :=true pour activer la connexion à la BD.	Beforeopen, Afteropen Beforeedit, Afteredit Beforeinsert, Afterinsert...
QUERY Requête SQL		<u>Databasename</u> : l'alias de la (les) table(s) qu'on va manipuler dans la requête. <u>SQL</u> : cette propriété lance une petite fenêtre d'édition du script SQL de cette requête. <u>Params</u> : pour donner le type de chaque paramètre. <u>Active</u> :=true pour activer la requête.	Beforeopen, Afteropen Beforeedit, Afteredit Beforeinsert, Afterinsert

CONTROLBD			
DBGRID Affichage de la source de données (champs et enregistrements)		<u>Datasource</u> : cette propriété sera liée avec le composant DATASOURCE représentant la source de données à afficher.	<u>OnColExit</u> <u>OnTitleClick</u>
DBNAVIGATEUR Représente les principaux opérations de manipulation des BD.		<u>Datasource</u> : cette propriété sera liée avec le composant DATASOURCE représentant la source de données à contrôler.	<u>Beforeaction</u> <u>onclick</u>
DBEDIT Affiche un champ d'une source de données.		<u>Datasource</u> : représente la source de données. <u>Datafield</u> : représente le champ de données à afficher.	<u>OnChange</u> <u>Onkeydown.</u>
DBCOMBOBOX Affiche un champ d'une source de données et propose une liste de valeurs à introduire.		<u>Datasource</u> : représente la source de données. <u>Datafield</u> : représente le champ de données à afficher. <u>Items</u> : contient la liste des valeurs à proposer à introduire dans la BD.	<u>OnChange</u> <u>OnClick</u> <u>Onkeydown.</u>
DBLOOKUPCOMBOBOX Afficher un champ d'une source de données et proposer une liste de valeurs à introduire à la BD d'une autre source		<u>Datasource</u> : représente la source de données. <u>Datafield</u> : représente le champ de données à afficher. <u>Listesource</u> : la source des données de la liste. <u>Listefield</u> : le(s) champ(s) dans la liste <u>Keyfield</u> : le champ qu'on affichera sa valeur.	<u>OnClick</u> <u>Onkeydown.</u>
DBCHART Représentation graphique d'une source de données.		On double clique sur ce composant et dans la boîte de dialogue qui s'affiche on clique sur le bouton ajouter de l'onglet série (inférieur) pour choisir le type de la représentation graphique, et dans l'onglet série (supérieur) On choisit dans le sous onglet source de données dans la liste l'élément « dataset » et on fait la liaison avec notre data source et on choisit le champ d'étiquettes « libellés » et le champ de quantifications « sectorielle »	<u>OnClick</u> <u>OnZoom</u>
QREPORT			
QUICKREP Etat d'impression		<u>Bands</u> : pour introduire les bandes dans l'état <u>Dataset</u> : représente la source de données à imprimer <u>Zoom</u> : contrôle la taille de l'état à la conception	<u>Onpreview</u> <u>Beforeprint</u>
QRLABEL Interface de l'état		Caption : insertion du texte à imprimer (texte fixe). Font : pour manipuler le format du texte à imprimer.	<u>Onprint</u>
QRDBTEXT Imprimer les données d'un champ d'une BD		<u>Datasource</u> : représente la source de données. <u>Datafield</u> : représente le champ de données à afficher.	<u>Onprint</u>
QRSHAPE Formes d'encadrement		Ce composant aide à encadrer les éléments de l'état d'impression pour mieux les présenter. Sa propriété « shape » nous donne le choix du style d'encadrement.	
DIALOGUES			
OPENDIALOG Boîte dialogue ouvrir		Ce composant utilise les DLL Windows pour afficher la boîte de dialogue standard d'ouverture d'un fichier. La propriété « filename » donne le nom et l'emplacement du fichier ouvert.	<u>Onclose</u> <u>Onshow</u>
FONTDIALOG Boîte dialogue police		utilise les DLL Windows pour afficher la boîte de dialogue standard de manipulation de la police d'écriture. La propriété « font » donne les paramètres de la police choisie.	<u>Onclose</u> <u>Onshow</u>
COLORDIALOG Boîte dialogue couleur		utilise les DLL Windows pour afficher la boîte de dialogue standard de manipulation de la couleur. La propriété « color » donne les paramètres de la couleur choisie.	<u>Onclose</u> <u>Onshow</u>