

Iset Sfax	Matière : Dotnet
Département Informatique	DSI2

TP5 : Les interfaces

Exercice 1

Les **animaux** peuvent crier, et manger

1-Ecrire les interfaces suivantes

```
interface ICrier{ string crier(); }
```

```
interface IManger{ string Manger(); }
```

2-Ecrire les classes **chien,chat,cheval** qui héritent de la classe animal et des interfaces Crier et Manger. Un animal possède un champ nom , categorie, resultatcri, resultatmanger

un chien crie en faisant "j'aboie" et manger en faisant "je mange très vite"

un chat crie en faisant "je miole" et manger en faisant "je mange trop vite"

un cheval crie en faisant "j'ébroue" et manger en faisant "je mange comme un cheval"

3-Ajouter une fenêtre Pour ajouter les animaux dans une liste et deux bouton Crier et Manger.

Le bouton Crier demande a tous les animaux de crier, de même pour manger.

Les animaux seront affichés dans un datagrid
colonnes(nom,categorie,resultatcri,resultatmanger)

Exemple après avoir cliqué sur Crier:

Nom	Categorie	resultatcri	resultatmanger
Berger	Chien	J'aboie	
Chat 1	Chat	Je miole	

puis avoir cliqué sur Manger:

Nom	Categorie	resultatcri	resultatmanger
Berger	Chien	J'aboie	je mange très vite
Chat 1	Chat	Je miole	je mange trop vite

Exercice 2

1/Créer un projet services dans la solution exercice2 de type windows forms

Soient les interfaces suivantes :

Interface **idbemployes**

```
{ void Ajouter(employe emp) ;  
void Supprimer (employe emp) ; }
```

et l'interface **ihistorique**

```
{void AjouterHisto(Historique h) ; }
```

Un **employe**(cin,nom,prenom) et **historique**(date,description).

On veut mettre en place un service de traitement des employés indépendant du type de la base de données.

Class **ServiceEmpls**

```
{  
ihistorique _histodata ;  
idbemployes _employesdata  
public ServiceEmpls(ihistorique histodata,idbemployes employesdata){...}  
public void CreerEmployes(employe emp)//après l'ajout dans la base il faut ajouter  
l'employe x est ajoute,nom=y,prenom=z a la date d'aujourd'hui  
public void SupprimerEmployes(emploue emp)//même chose  
}
```

Dans cette classe on ne connaît pas le type de la base de données.

2/dans le projet exercice (Front+DB) on va mettre en place l'accès à la BD simplifiée et la fenêtre de saisie.

a/Ecrire les classes **dbemployes** et **dbhistorique** qui implémentent les interfaces de telle sorte on ajoute les données dans un fichier StreamWriter swr=new StreamWriter("fichemps.txt") ;

```
swr.Write("Code=x ;Nom=y ;Prenom=z") ;swr.Close() ;
```

b/Dans le program Main appeler la form1 en lui passant les objets **dbemployes** et **dbhistorique** :

```
dbemployes dbemps1=new dbemployes() ;
```

```
dbhistorique dbemps1=new dbhistorique () ;
```

```
Form1 f1=new Form1(dbemps1,dbhisto) ;
```

```
Application.Run(f1);
```

3/Testez l'ajout des employes et vérifiez le fichier fichemps.txt dans le répertoire debug.

Dans form1 il faut créer un attribut **ServiceEmps** _serviceempls ;

Cet attribut doit être instancié dans le constructeur

On crée un employé puis on appelle _serviceempls.Ajouter(unemp) ;