

IV. Éléments de correction de DAI :

DOSSIER I : Service d'accueil

1- Les objets nécessaires (1pt)

- 3 objets sqlDataAdapter : 1 pour TypeEm, 1 pour Emission et 1 pour la jointure
- Un objet DataSet
- Un objet sqlConnection

2- Initialisation de ces objets (1pt) :

```
Private cn As SqlConnection
Private adptType, adptEm, adptem1 As SqlDataAdapter
Private dset As New DataSet
Sub connecter()
    Try
        cn = New SqlConnection("initial catalog=dbGEm;data
source=dbServer;integrated security=true;")
        cn.Open()
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
End Sub

Sub charger()
    adptType = New SqlDataAdapter("select * from typeem", cn)
    adptEm = New SqlDataAdapter("select * from emission", cn)
    adptem1 = New SqlDataAdapter("select codeem as Code,TitreEm as
Titre,datehem as Date,nomtype as Type from emission e inner join typeem t on
e.codetype=t.codetype", cn)
    adptType.Fill(dset, "typeem")
    adptEm.Fill(dset, "emission")
    adptem1.Fill(dset, "emission1")
EndSub
```

```
Sub chargerTypes( ) (1,5 pt)
    cbType.valueMember= "codeType"
    cbType.DisplayMember= "nomType"
    cbType.datasource=dset.tables("TypeEm")
end sub
```

```
Sub Lister( ) (1,5 pt)
    Table.datasource= dset.tables("Emission1")
End sub
```

cmdAjouter : permet d'ajouter une nouvelle émission, avec vérification de la validité du formulaire (vérification : 0,5 pt ajout : 1pt)

```
Private Sub cmdAjouter_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdAjouter.Click (1 pt)
    If isValid() = False Then
        MessageBox.Show("Formulaire incomplet")
        Exit Sub
    End If
    Dim L As DataRow
    L = dset.Tables("emission").NewRow
    L(1) = Me.txtTitre.Text
    L(2) = Me.dateEm.Value
    L(3) = Me.txtDuree.Text
    L(4) = Me.cbType.SelectedValue
```

```

dset.Tables("emission").Rows.Add(L)
Dim b As New SqlCommandBuilder(Me.adptEm)
adptEm.Update(dset, "emission")
dset.Clear()
charger()
listerTypes()
listerem()
End Sub
'-----
Function isValid() As Boolean (0,5 pt)
    If txtTitre.Text.Trim.Equals("") Or txtDuree.Text.Trim.Equals("") Or
cbType.SelectedIndex = -1 Then
        Return False
    End If
    Return True
End Function

```

cmdModifier : il permet de modifier une émission déjà inscrite (récupérer : 0,5 pt modifier : 1pt)

```

Private Sub cmdModifier_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cmdAjouter.Click
Dim c, p As Integer
    p = Me.table.CurrentRow.Index
    If p = -1 Then Exit Sub
    c = Integer.Parse(Me.table.Rows(p).Cells(0).Value.ToString())
    If isValid() = False Then
        MessageBox.Show("Formulaire incomplet")
    Exit Sub
End If
Dim L As DataRow
    L = dset.Tables("emission").Rows(p)
    L(1) = Me.txtTitre.Text
    L(2) = Me.dateEm.Value
    L(3) = Me.txtDuree.Text
    L(4) = Me.cbType.SelectedValue
Dim b As New SqlCommandBuilder(Me.adptEm)
adptEm.Update(dset, "emission")
dset.Clear()
charger()
listerTypes()
listerem()
End Sub

```

cmdFin : end. (0,5 pt)

Donner le code de la procédure événementielle de la zone texte txtRech (1,5 pt) :

```

Private Sub txtRech_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txtRech.TextChanged
    Dim dv As New DataView(dset.Tables("emission1"))
    dv.RowFilter = "titre like '" + txtRech.Text + "%'"
    table.DataSource = dv
End Sub

```

DOSSIER II : Service Web

- 1- La fonction connecter() (2 pts) :

```
<?php
function connecter(){
    mysql_connect("localhost","user","12345") or die("err de serveur");
    mysql_select_db("dbgem") or die("err de bd");
}
?>
```

- 2- le code HTML de notre page web est le suivant (3 pts):

```
<html>
  <head>
    < ?php
      function connecter () {...}      ?>  </head>
  <body>
    <h1 align="center">Listing des émissions</h1>
    <form method="POST" action="voter.php">
      <table align="center" border="2" width="60%">
        <caption>Liste de toutes les émissions</caption>
        <tr>
          <th>Code</th>
          <th>Titre</th>
          <th>Date</th>
          <th>Type</th>
          <th>Selectionner</th> </tr>
        <?php
          connecter();
          $r=mysql_query("select codeem as Code, TitreEm as Titre, dateHeureEm as
            Date, nomtype as Type from emission e inner join typeem t on
            e.codetype=t.codetype");
          while($L=mysql_fetch_row($r))
            echo
              "<tr><td>$L[0]</td><td>$L[1]</td><td>$L[2]</td><td>$L[3]</td><td><input
                type='checkbox' value=$L[0] name=T[]></td></tr>";
              ?>
            </table>
            <input type="submit" value=" Voter ">
          </form>
        </body></html>
```

- 3- Le code du fichier « voter.php » (3 pts):

```
<?php
    $a=$_POST["T"];
    connecter();
    foreach($a as $v)
      $r=mysql_query("update emission set votes=votes+1 where codeem=$v");
    header("location:index.php");
?>
```

DOSSIER III : Service de programmation des émissions

1) Le détail de la classe Emission est le suivant (14 pts : 1pt par question):

```
public abstract class Emission implements Serializable{
    private String TitreEm;
    private String duréeEm;
    private Date dateEm;
    //-----
    -----
    public Emission(String TitreEm, String duréeEm, Date dateEm) {
        this.TitreEm=TitreEm;
        this.duréeEm=duréeEm;
        this.dateEm=dateEm;
    }
    //-----
    -----
    public Emission(String TitreEm, String duréeEm) {
        this.TitreEm=TitreEm;
        this.duréeEm=duréeEm;
        this.dateEm=new Date();
    }
    //-----
    -----
    @Override
    public boolean equals(Object ob)
    {
        Emission a=(Emission) ob;
        return TitreEm.equals(a.TitreEm);
    }
    //-----
    -----
    @Override
    public String toString() {
        SimpleDateFormat sdf = new SimpleDateFormat("jj/MM/yy");
        return "Titre :" + TitreEm+", la durée :" + duréeEm+", Date  
d'émission :" + sdf.format(dateEm);
    }
    //-----
    public abstract String getInfos();
}
```

2) Le détail de la classe Documentaire le suivant :

```
public class Documentaire extends Emission{
    private String theme;
    public Documentaire(String TitreEm, String duréeEm, Date
    dateEm,String t) {
        super(TitreEm, duréeEm, dateEm);
        theme=t;
    }
    public String getInfos(){
        return super.toString()+"\n"+"Theme:" + theme;
    }
}
```

3) Le détail de la classe Fiction le suivant :

```
public class Fiction extends Emission{
    private String réalisateur, producteur;
    public Fiction(String TitreEm, String duréeEm, Date
dateEm,Stringr,String p) {
        super(TitreEm, duréeEm, dateEm);
        réalisateur=r;
        producteur=p; }
    public String getInfos(){
        return
        super.toString()+"\nRéalisateur:"+réalisateur+"\nProducteur :"+producteur;
    } }
}
```

4) Le détail de la classe Programme le suivant :

```
public class Programme{
private ArrayList<Emission>Liste=new ArrayList<Emission>() ;
//-----
public boolean addEmission(Emission x) {
return Liste.add(x);    }
//-----
    public Emission delEmission(int index) {
if((index<0)|| (index>=Liste.size())) return null;
return Liste.remove(index);
}
//-----
public boolean delEmission(Emission p){
return Liste.remove(p);
}
//-----
public Emission setEmission(int index,Emission x){
if((index<0)|| (index>=Liste.size())) return null;
return Liste.set(index,x) ;
}

//-----
public void affiche( ){
String r=""
for(Emission e:Liste) r+=e+"\n";
System.out.println(r) ;
}
//-----
public Boolean saveAll(String f) {
Try{ObjectOutputStream ob=new ObjectOutputStream(FileOutputStream(f));
for(Emission e:Liste)
{ob.writeObject(e) ;
ob.flush(); }
ob.close();
return true;
}catch(Exception ex)
{
ex.printStackTrace();
return false; } }

//-----
}
```

```

    public boolean loadAll(String f) {
        Liste.clear();
        try{ObjectInputStream ob=new ObjectInputStream(FileInputStream(f));
        while(true)
        {Emission em;
        try{
        em=(Emission)ob.readObject();
        Liste.add(em);
        }
        Catch(Exception ex1){break;}
        }
        ob.close();
        return true;
        }catch(Exception ex)
        {
        ex.printStackTrace();
        return false;
        }

    }
}

```

DOSSIER IV : Administration distante de l'application

```

public class Connexion{
private String URL,User, Password ;
private Connection cn;
//-----
public Connexion(String URL, String User, String Password) {
    this.URL= URL;
    this.User= User;
    this.Password= Password;}
//-----
    public boolean connecter( ){
    try{
    Cn=DriverManager.getConnection(URL,User,Password);
    }catch(Exception ex)
        { Ex.printStackTrace(); }}
//-----
    public ResultSet lire(String requete){
        try{
        Statement st=cn.createStatement();
        ResultSet rs=st.executeQuery(requete);
        return rs;
        }catch(Exception ex)
        { return null; } }
//-----
    public Vector<Object[]> toVector(String requete, int n) {
    try{
    Vector v=new Vector();
    ResultSet r= lire(requete);

```

```

while(r.next())
{
Object[] t = new Object[n];
for(int i=0;i<n;i++)
    t[i]= r.getObject(i+1);
v.add(t);
}
return v;
}catch(Exception ex){return null;}
}
//-----
public Boolean miseAJour(String requete){
    try{
Statement st=cn.createStatement();
st.executeUpdate(requete);
return true;
}catch(Exception ex)
{ return false; }
}
//-----
public boolean fermer( ){
try{
cn.close();
return true;
}catch(Exception ex)
{ Ex.printStackTrace();
return false; }}

```

```

public class Client {
private String IP;
private int Port;
public Client (String IP, int Port)
{.....}
public Object demande(String requete)
{Try
{ Socket C = new Socket (IP,Port);
ObjectInputStream in = new ObjectInputStream(C.getInputStream());
PrintStream out = new PrintStream(new
BufferedOutputStream(C.getOutputStream()));
Out.println(requete);
return in.readObject();
}
catch (Exception e) {return null;}
}
}}

```