Classe de configuration;

class conf

{

// connection

public static SqlConnection cn = new SqlConnection(@"Data Source=ER-RAFAIY-

PC\ERRAFAIY;Initial Catalog=gestion\_Bibliotheque;Integrated Security=True");

//Data Adapter

public static SqlDataAdapter da;

//Data Set

public static DataSet ds=new DataSet();

//Cmman Builder pour trnsfere les donne de dete set ou Base de donne

public static SqlCommandBuilder cmb;

}

//Method de charger le data set

private void charger()

{

//vider data set

conf.ds.Clear();

//commande SQl passer par data adapter

conf.da = new SqlDataAdapter("select \* from FONCTION", conf.cn);

//rambler data set

conf.da.Fill(conf.ds, "fon");

//rambler data GridView1

dataGridView1.DataSource = conf.ds.Tables["fon"];

}

//ajouter nouveaux enregistrement ou data set

//réserver un rows vide ou data set

DataRow dr = conf.ds.Tables["fon"].NewRow();

//rambler le nouveau row

dr[0] = textBox1.Text;

dr[1] = textBox2.Text;

//Ajouter le rows ou data set

conf.ds.Tables["fon"].Rows.Add(dr);

// transfer les donne ou base de donne

conf.cmb = new SqlCommandBuilder(conf.da);

conf.da.Update(conf.ds, "fon");

//Chercher un enregistrement ou data set

foreach (DataRow i in conf.ds.Tables["fon"].Rows)

{

if (i[0].ToString()==textBox1.Text)

{

textBox1.Text = i[0].ToString();

textBox2.Text = i[1].ToString();

}

}

//Modifier un enregistrement ou data set

foreach (DataRow i in conf.ds.Tables["fon"].Rows)

{

if (i[0].ToString() == textBox1.Text)

{

i[1] = textBox2.Text;

// transfer les donne ou base de donne

conf.cmb = new SqlCommandBuilder(conf.da);

conf.da.Update(conf.ds, "fon");

}

}

//Supprimer un enregistrement ou data set

for (int i = 0; i < conf.ds.Tables["fon"].Rows.Count; i++)

{

if (conf.ds.Tables["fon"].Rows[i][0].ToString() == textBox1.Text)

{

conf.ds.Tables["fon"].Rows.Remove(conf.ds.Tables["fon"].Rows[i]);

conf.cmb = new SqlCommandBuilder(conf.da);

conf.da.Update(conf.ds, "fon");

}

}

//Chileren

DataTable listeLivre = ds.Tables[1].Clone();

listeLivre.Rows.Clear();

foreach (DataRow pr in ds.Tables[0].Rows) //parcourir tous les auteurs

{

if (pr[0].ToString().Equals( CBAuteur.SelectedValue.ToString()))

// traiter uniquement l'auteur de combobox

{

foreach (DataRow child in pr.GetChildRows(rel))

{

DataRow dr= listeLivre.NewRow();

dr[0] = child[0]; dr[1] = child[1]; dr[2] = child[2]; dr[3] = child[3];

listeLivre.Rows.Add(dr);

}// fin des livres de cet auteur

break;

}//test de correspondance entre combobox et la table auteur

}//fin du parcours auteur

dataGridView1.DataSource = listeLivre;

//cette instruction permet de appliquer ces contraintes

ds.EnforceConstraints = true;

//\*\*\*\*\*\*\*\* constraint primary key

t2.Constraints.Add("cst\_Primary\_key", t2.Columns[0], true);

//\*\*\*\*\*\*\*\* Solution 2

t1.Constraints.Add("cst\_Primary\_key", new DataColumn[] { t1.Columns[0], t1.Columns[2] }, true);

//declarer la deuxième colonne comme unique

t1.Constraints.Add(new UniqueConstraint("Unique1",t1.Columns[1]));

//\*\*\*\*\*\*\*\* Constraint foreign key

t2.Constraints.Add("cst\_fk\_service\_employe", t1.Columns[0],t2.Columns[1]);

//\*\*\*\*\*\*\*\* Constraint foreign key deuxieme syntaxe

ForeignKeyConstraint fcst = new ForeignKeyConstraint("cst\_fk\_service\_employe", t1.Columns[0],

t2.Columns[1]);

fcst.DeleteRule = Rule.Cascade;

fcst.UpdateRule = Rule.Cascade;

t2.Constraints.Add(fcst);

//relation

rel = new DataRelation("Auteur\_Livre", ds.Tables["TAuteur"].Columns[0], ds.Tables["TLivre"].Columns[1]);

ds.Relations.Add(rel);

//navigation en mode connecter

//déclare

BindingSource BS;

//Utilisation de navigation

BS.DataSource = dt;

dataGridView1.DataSource = dt;

dataGridView1.ReadOnly=true;

//liaison entre les textBox et les colonnes de BS

TxtNumpilote.DataBindings.Add("Text", BS, "numpil");

Txtnompilote.DataBindings.Add("Text", BS, "nompil");

TxtVillePilote.DataBindings.Add("Text", BS, "adr");

TxtSalairePilote.DataBindings.Add("Text", BS, "salaire");

//first

BS.MoveFirst();

BS.MoveLast();

BS.MoveNext();

BS.MovePrevious();

//mode deconnecter

conf.da =new SqlDataAdapter( "select \* from FONCTION",conf.cn);

conf.da.Fill(conf.ds, "t1");

bmb = this.BindingContext[conf.ds.Tables[0]];

dataGridView1.DataSource = conf.ds.Tables[0];

textBox1.DataBindings.Add("text", conf.ds.Tables[0], "ID\_FONCTION");

textBox2.DataBindings.Add("text", conf.ds.Tables[0], "NOM\_FONCTION");

// Aperçu

c1.v = comboBox1.SelectedValue.ToString();

Aperçu a = new Aperçu();

a.Show();

///win imp

CrystalReport1 r = new CrystalReport1();

r.SetParameterValue("Num\_Sta", c1.v);

crystalReportViewer1.ReportSource = r;

crystalReportViewer1.Refresh();