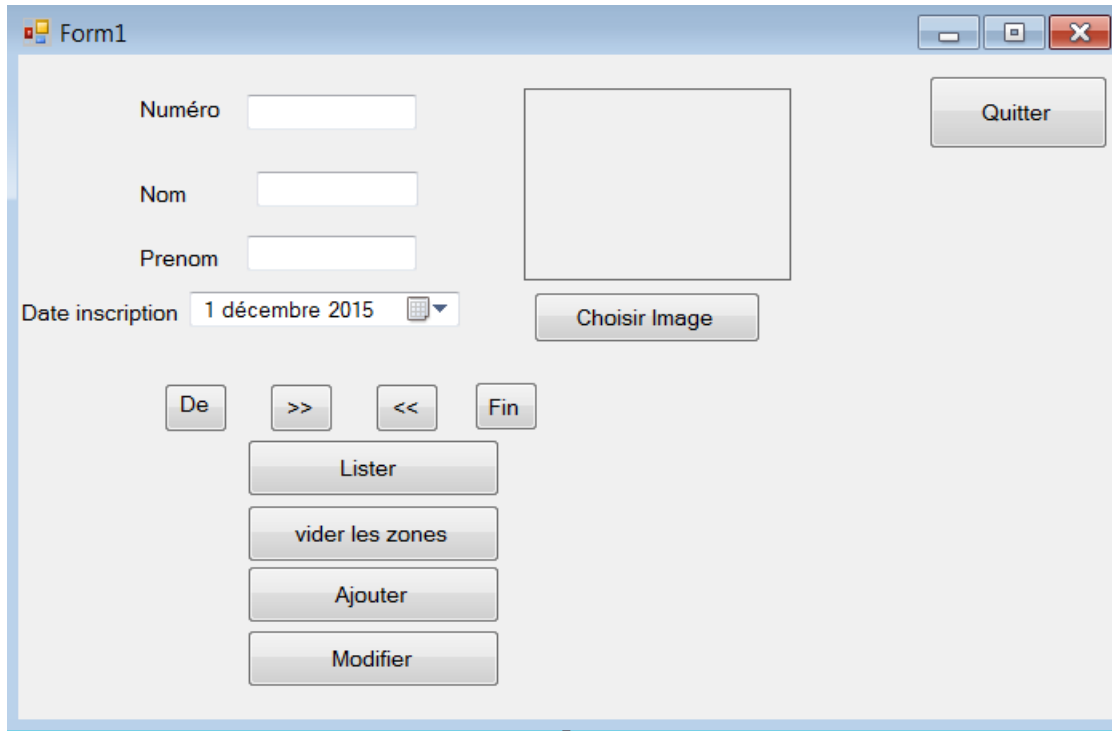


Exemple utilisant l'insertion, l'affichage et la modification avec des données de type BLOB



Pour ajouter un enregistrement contenant une photo, on procède comme suit :

- 1- On cherche le fichier correspondant à la photo dans le disque.
- 2- On utilise des OracleParameters pour chaque colonne à insérer (nom, prenom, dateinscription, photo)
- 3- On affecte des valeurs à chaque paramètre. Pour les colonnes autres que photo, vous devez vous référer aux exemples précédents
- 4- Pour le paramètre correspondant à la colonne photo, vous devez lire le fichier image et le convertir en bytes. Oracle stocke les images sous forme de BLOB (Binary Large Object).
- 5- Envoyer la requête avec les paramètres comme d'habitude avec un OracleCommand.

```
public Form1()
{
    InitializeComponent();
}

OracleConnection conn = new OracleConnection();
private DataSet monDataSet = new DataSet();

string sql1 = "SELECT NUMAD, NOM, PRENOM,photo FROM ETUDIANTSINFO";
string sql2 = "select numad, nom, prenom, dateinscription,photo from etudiantsinfo";

string nomFichier;

private void Form1_Load(object sender, EventArgs e)
{
    try
    {
        string chaineDeconnexion = "Data Source = primogene; User Id = ussger; password = blabl";
        conn.ConnectionString = chaineDeconnexion;
        conn.Open();
        MessageBox.Show(conn.State.ToString());
    }

    catch (Exception sqlConn)
    {
        MessageBox.Show(sqlConn.Message.ToString());
    }
}
```

Code du bouton Ajouter

```
private void ajouter2_Click(object sender, EventArgs e)
{
    try
    {
        // la requête Sqlajout est paramétrée. Elle a 4 paramètres.
        //les paramètres pour Oracle et C # sont précédés de deux points :
        string sqlajout = " insert into etudiantsinfo" +
            " (numad, nom,prenom,dateinscription,photo) values "+
            "(SeqEtu.nextval,:nom,:prenom,:dateinscription,:photo)";

        // On déclare les paramètres pour chaque colonne de la requête
        // la séquence est pour le numéro.
        OracleParameter oranom = new OracleParameter(":nom", OracleDbType.Varchar2, 30);
        OracleParameter oraprn = new OracleParameter(":prenom", OracleDbType.Varchar2, 40);
        OracleParameter oradate = new OracleParameter(":dateinscription", OracleDbType.Date);
        OracleParameter oraphoto = new OracleParameter(":photo", OracleDbType.Blob);
        // On affecte les valeurs aux paramètres.
        oranom.Value = textNom.Text;
        oraprn.Value = textPrenom.Text;
        oradate.Value = dateIns.Value;

        //on fait du streaming (lecture en continue du fichier le résultat on le met dans une variable de type
        byte (octets).
```

```
        FileStream Streamp = new FileStream(nomFichier, FileMode.Open, FileAccess.Read);
        byte[] buffer1 = new byte[Streamp.Length];
        Streamp.Read(buffer1, 0, System.Convert.ToInt32(Streamp.Length));
        Streamp.Close();
        // On affecte la valeur de buffer1 au paramètre oraphoto
        oraphoto.Value = buffer1;
```

```

OracleCommand oraAjout = new OracleCommand(sqlajout, conn);
oraAjout.CommandType = CommandType.Text;
// En utilisant la propriété Parameters de OracleCommand, on spécifie les paramètres de la requête SQLajout.
oraAjout.Parameters.Add(oranom);
oraAjout.Parameters.Add(oraprn);
oraAjout.Parameters.Add(oradate);
oraAjout.Parameters.Add(oraphoto);
// on execute la requete
oraAjout.ExecuteNonQuery();

// on appelle la fonction dissociier pour pouvoir inserer une deuxième fois.
// cette fonction est définie plus haut.
dissocier();

}

catch (Exception exsqlajout)
{
    MessageBox.Show(exsqlajout.Message.ToString());
}
}

```

La méthode `Stream.Read()`

Lit un bloc d'octets à partir du flux et écrit les données dans une mémoire tampon donnée. (Substitue [Stream.Read\(Byte\[\], Int32, Int32\)](#).)

buffer

Type : [System.Byte\[\]](#)

Tableau d'octets. Lorsque cette méthode est retournée, la mémoire tampon contient le tableau d'octets spécifié dont les valeurs comprises entre offset et (offset + count - 1) sont remplacées par les octets lus dans la source en cours.

offset

Type : [System.Int32](#)

Dans buffer, offset d'octet de base zéro à partir duquel commencer l'enregistrement des données lues dans le flux actuel.

count

Type : [System.Int32](#) Nombre maximal d'octets à lire à partir du flux actuel.

Code de la fonction qui permet de chercher un fichier image dans le disque :

```
private string RechercherFichier()
{
    OpenFileDialog fImage = new OpenFileDialog();
    fImage.Title = "selectionner une image";
    fImage.CheckFileExists = true;
    fImage.InitialDirectory = @"C:\";

    // fImage.InitialDirectory = Application.StartupPath;
    fImage.Filter = "Fichiers images (*.BMP;*.JPG;*.GIF)|*.BMP;*.JPG;*.GIF|All files (*.*)|*.*";
    fImage.FilterIndex = 1;
    fImage.RestoreDirectory = true;

    if (fImage.ShowDialog() == DialogResult.OK)
    {
        nomFichier = fImage.FileName;
        Bitmap bitmap1 = new Bitmap(nomFichier);
    }
    else
    {
        nomFichier = null;
    }
    return nomFichier;
}
```

Code du bouton associer à charger le fichier Image (photo trouvée par la fonction précédente) dans le contrôle PictureBox.(bouton Choisir Image)

```
private void ChercherImg_Click(object sender, EventArgs e)
{
    nomFichier = RechercherFichier();
    if (nomFichier != null)
    {
        picPhoto.Image = System.Drawing.Image.FromFile(nomFichier);
        picPhoto.ImageLocation = nomFichier;
    }
}
```

Afficher une image provenant d'une BD.

Lorsque vous utilisez un DataSet, il suffit de faire un DataBindings sur le contrôle approprié (PictureBox) pour

La fonction `afficherTxt()`, permet de remplir le DataSet avec la requête sql2, en utilisant un OracleDataAdapter.

Elle appelle ensuite la fonction `lier()` pour faire les DataBindings.

```
private void afficherTxt()
{
    try
    {
        OracleDataAdapter Adapter2 = new OracleDataAdapter(sql2, conn);
        if (monDataSet.Tables.Contains("resEtudiants"))
        {
            monDataSet.Tables["resEtudiants"].Clear();
        }
        Adapter2.Fill(monDataSet, "resEtudiants");
        Adapter2.Dispose();
        // on appelle la fonction lier pour faire la liaison des données du DataSet avec les zones de Text.
        lier();
    }
}
```

```

        catch (Exception exsql2)
        {
            MessageBox.Show(exsql2.Message.ToString());
        }
    }

    private void lier()
    {
        textNumad.DataBindings.Add("text", monDataSet, "resEtudiants.numad");
        textNom.DataBindings.Add("text", monDataSet, "resEtudiants.nom");
        textPrenom.DataBindings.Add("text", monDataSet, "resEtudiants.Prenom");
        dateIns.DataBindings.Add("text", monDataSet, "resEtudiants.dateinscription");
        picPhoto.DataBindings.Add("image", monDataSet, "resEtudiants.photo", true);
    }
}

```

Code correspondant au bouton Lister :

```

private void ListerTout_Click(object sender, EventArgs e)
{
    afficherTxt();
}

```

Code de la fonction appelée par le bouton Vider les zones : Tout contrôle lié par un DataBindings doit être dissocié.

```
private void dissocier()
{
    textNumad.DataBindings.Clear();
    textNom.DataBindings.Clear();
    textPrenom.DataBindings.Clear();
    dateIns.DataBindings.Clear();
    picPhoto.DataBindings.Clear();
    // effacer le contenu
    textNumad.Clear();
    textNom.Clear();
    textPrenom.Clear();
    dateIns.Value = DateTime.Now;
    picPhoto.Image = null;
}
```

Compléments : Voici le code associé au bouton Modifier.

Il utilise le OracleParameter comme dans une insertion.

Le Fichier de type image en converti en tableau de Bytes. Le résultat est dans buffer1. C'est buffer1 qu'on affecte comme valeur au paramètre correspondant à la photo (oraphoto)


```

private void Modifier_Click(object sender, EventArgs e)
{
    try
    {
        string sqlmodif = "update etudiantsinfo set nom =:nom,photo=:photo where numad=:numad";
        OracleParameter oranum = new OracleParameter(":numad", OracleDbType.Int32);
        OracleParameter oranom = new OracleParameter(":nom", OracleDbType.Varchar2, 30);
        OracleParameter oraphoto = new OracleParameter(":photo", OracleDbType.Blob);
        // on affecte les valeurs aux paramètres.
        oranom.Value = textNom.Text;
        oranum.Value = textNumad.Text;
        //on fait du streaming (lecture en continue du fichier le résultat est mis dans un tableau de bytes.
        FileStream Streamp = new FileStream(nomFichier, FileMode.Open, FileAccess.Read);
        byte[] buffer1 = new byte[Streamp.Length];
        Streamp.Read(buffer1, 0, System.Convert.ToInt32(Streamp.Length));
        Streamp.Close();
        oraphoto.Value = buffer1;

        OracleCommand oraModif = new OracleCommand(sqlmodif, conn);
        oraModif.CommandType = CommandType.Text;
        //En utilisant la propriété Prameteres de OracleCommand, on spécifie les paramètres de la requete sqlmodif.
        oraModif.Parameters.Add(oranom);
        oraModif.Parameters.Add(oraphoto);
        //remarquer que oranum est en dernier.
        oraModif.Parameters.Add(oranum);
        // on execute la requête
        oraModif.ExecuteNonQuery();
        // on appelle la fonction dissocier pour pouvoir inserer ou modifier une deuxième fois.
        dissocier();
    }
    catch (Exception exsqlModif)
    {
        MessageBox.Show(exsqlModif.Message.ToString());
    }
}

```