

TP No4

Opération initiale :

Créez un nouveau projet de type C# « Application Windows Forms » ayant comme caractéristiques :

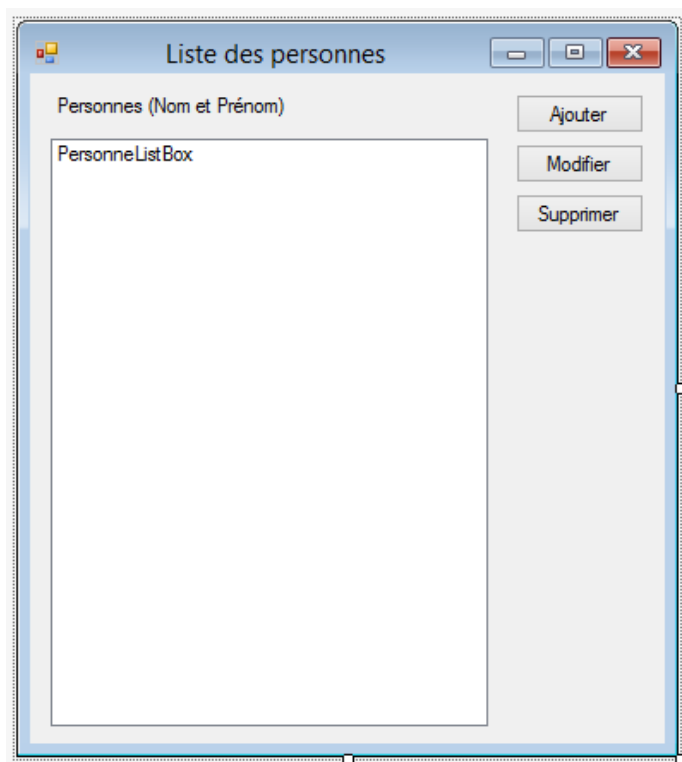
Nom : MaSolution.Crud (ce sera le nom de votre application principale de type forms)

Solution : MaSolution

Exercice1 :

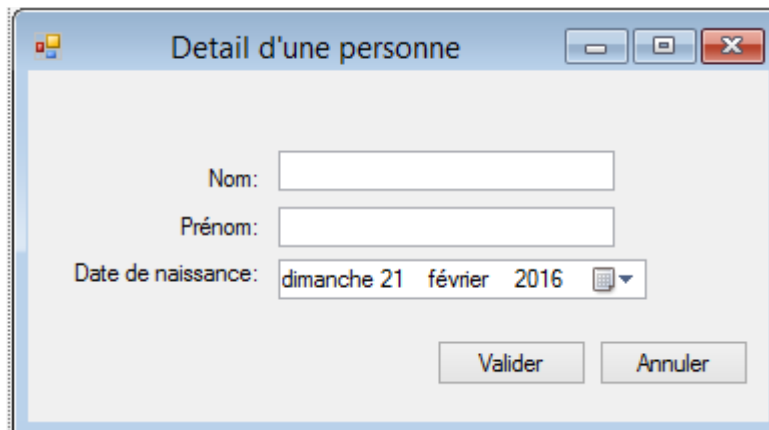
Partie1 :

Créez une fenêtre nommée « MainForm » ayant pour aspect :



Partie 2 :

Créez une fenêtre nommée « PersonneDetailForm » ayant pour aspect :



Exercice 2 :

Partie1 :

Ajoutez à votre solution une Bibliothèque de Classe nommée « MaSolution4.Bo »

Ajoutez le fichier de classe ObjectBase.cs contenant le code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace MaSolution4.Bo
{
    public class ObjetBase
    {
        public Int64 Identifiant { get; set; }
    }
}
```

Ajoutez le fichier de classe Personne.cs contenant le code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace MaSolution4.Bo
{
    public class Personne : ObjetBase
    {
        public String Nom { get; set; }
        public String Prenom { get; set; }
        public DateTime DateNaissance { get; set; }
    }
}
```

Ajoutez une méthode publique IsValid() permettant de tester que le nom et le prenom ont au moins 5 caractères.

Partie2 :

Ajoutez à votre solution une Bibliothèque de Classe nommée « MaSolution4.Bll »

Ajoutez le fichier de classe PersonneManager.cs contenant le code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace MaSolution4.Bll
{
    public class PersonneManager
    {
        private MaSolution4.Dal.PersonneManager _PersonneManager = null;

        public PersonneManager()
        {
            _PersonneManager = new Dal.PersonneManager();
        }

        public List<MaSolution4.Bo.Personne> GetPersonnes()
        {
            return _PersonneManager.GetPersonnes();
        }

        public MaSolution4.Bo.Personne Add(MaSolution4.Bo.Personne personne)
        {
            return _PersonneManager.Add(personne);
        }

        public Boolean Update(MaSolution4.Bo.Personne personne)
        {
            return _PersonneManager.Update(personne);
        }

        public Boolean Delete(MaSolution4.Bo.Personne personne)
        {
            return _PersonneManager.Delete(personne);
        }
    }
}
```

Pensez à : Ajouter une référence à MaSolution4.Bo et MaSolution4.Dal

Ajoutez à votre solution une Bibliothèque de Classe nommée « MaSolution4.Dal »

Partie 3 :

Créez un dossier appelé : Librairies

Ajouter les fichiers suivants (provenant du fichier Librairies.rar) dans le dossier Librairies:

- FileData.dll
- FileDataProvider.dll
- PersonneData.dll

Ajouter ces librairies aux références de classe de ce projet

Ajoutez le fichier de classe Converter.cs contenant le code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace MaSolution4.Dal
{
    public static class Converter
    {
        public static PersonneData.PersonneDao ToDao(this MaSolution4.Bo.Personne personne)
        {
            return new PersonneData.PersonneDao
            {
                Identifiant = personne.Identifiant,
                Nom = personne.Nom,
                Prenom = personne.Prenom,
                DateNaissance = personne.DateNaissance
            };
        }

        public static List<PersonneData.PersonneDao> ToDao(this
List<MaSolution4.Bo.Personne> personnes)
        {
            List<PersonneData.PersonneDao> result = new List<PersonneData.PersonneDao>();
            foreach (var item in personnes)
            {
                result.Add(item.ToDao());
            }
            return result;
        }

        public static MaSolution4.Bo.Personne ToBo(this PersonneData.PersonneDao personne)
        {
            return new MaSolution4.Bo.Personne
            {
                Identifiant = personne.Identifiant,
                Nom = personne.Nom,
                Prenom = personne.Prenom,
                DateNaissance = personne.DateNaissance
            };
        }

        public static List<MaSolution4.Bo.Personne> ToBo(this List<PersonneData.PersonneDao>
personnes)
        {
            List<MaSolution4.Bo.Personne> result = new List<MaSolution4.Bo.Personne>();
            foreach (var item in personnes)
            {
                result.Add(item.ToBo());
            }
            return result;
        }
    }
}
```

Ajoutez le fichier de classe PersonneManager.cs contenant le code :

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using FileDataProvider;
using PersonneData;

namespace MaSolution4.Dal
{

    public class PersonneManager
    {

        private Command<PersonneDao> _PersonneCommande = null;

        public PersonneManager()
        {
            _PersonneCommande = new Command<PersonneDao>(".\\PersonneData.xml");
        }

        public List<MaSolution4.Bo.Personne> GetPersonnes()
        {
            return _PersonneCommande.GetList().ToBo();
        }

        public MaSolution4.Bo.Personne Add(MaSolution4.Bo.Personne personne)
        {
            return _PersonneCommande.Add(personne.ToDao()).ToBo();
        }

        public Boolean Update(MaSolution4.Bo.Personne personne)
        {
            return _PersonneCommande.Update(personne.ToDao());
        }

        public Boolean Delete(MaSolution4.Bo.Personne personne)
        {
            return _PersonneCommande.Delete(personne.ToDao());
        }
    }
}

```

Pensez à : Ajouter une référence à MaSolution4.Bo

Exercice 3 :

Dans votre projet « MaSolution4 »,

Ajoutez un dossier nommé Dtc,

Créez une classe Personne identique à la classe Personne de l'assembly MaSolution4.Bo, surchargez la méthode ToString pour affichez le nom, le prenom et la date de naissance.

Créez une classe Converter à l'image de la classe Converter de l'assembly MaSolution4.Dal, cette classe convertira des objets Bo.Personne et Dtc.Personne et inversement.

Ajoutez le fichier de classe PersonneManager.cs contenant le code :

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace WindowsForm1.Dtc
{
    public class PersonneManager
    {
        private MaSolution4.Bll.PersonneManager _PersonneManager = null;

        public PersonneManager()
        {
            _PersonneManager = new MaSolution4.Bll.PersonneManager();
        }

        public List<Dtc.Personne> GetPersonnes()
        {
            return _PersonneManager.GetPersonnes().ToDtc();
        }

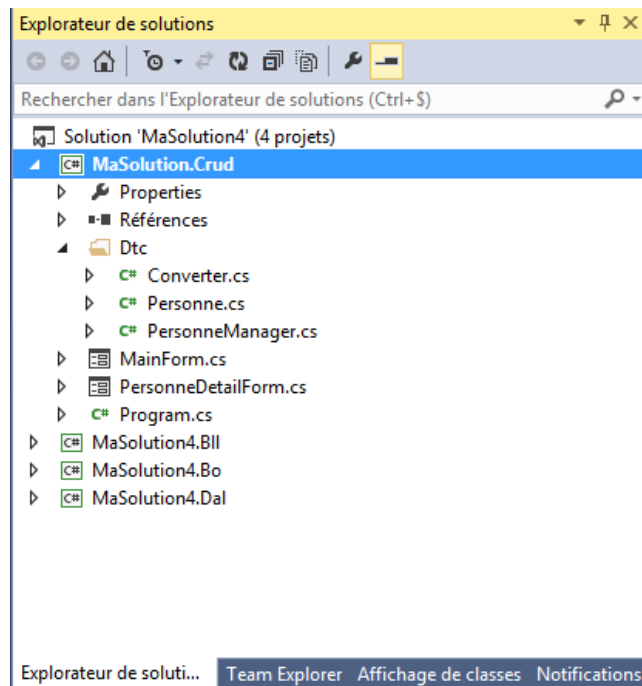
        public MaSolution4.Bo.Personne Add(Dtc.Personne personne)
        {
            return _PersonneManager.Add(personne.ToBo());
        }

        public Boolean Update(Dtc.Personne personne)
        {
            return _PersonneManager.Update(personne.ToBo());
        }

        public Boolean Delete(Dtc.Personne personne)
        {
            return _PersonneManager.Delete(personne.ToBo());
        }
    }
}

```

Vous devriez avoir les fichiers .cs comme dans la capture d'écran ci-dessous :



Pensez à : Ajouter une référence à MaSolution4.Bo et MaSolution4.Bll

Exercice 4 :

Adaptez le code ci-dessous à votre formulaire « MainForm » :

```
using MaSolution4.Bo;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace WindowsForm1
{
    public partial class MainForm : Form
    {
        Dtc.PersonneManager _PersonneManager = null;

        public MainForm()
        {
            InitializeComponent();
        }

        private void AjouterPersonneButton_Click(object sender, EventArgs e)
        {
            PersonneDetailForm personneDetailForm = new PersonneDetailForm();
            personneDetailForm.ShowDialog();
        }

        private void ModifierButton_Click(object sender, EventArgs e)
        {
            PersonneDetailForm personneDetailForm = new PersonneDetailForm();
            personneDetailForm.ShowDialog();
        }
    }
}
```

```

private void SupprimerButton_Click(object sender, EventArgs e)
{
    if (PersonneListBox.SelectedIndex >= 0)
    {
        if (MessageBox.Show("Êtes vous sur de vouloir supprimer cette personne ?",
            "Confirmation de suppression!!",
            MessageBoxButtons.YesNo) ==
System.Windows.Forms.DialogResult.Yes)
        {
            _PersonneManager.Delete((Dtc.Personne)PersonneListBox.SelectedItem);
            ListBoxPersonnelInitialize();
        }
    }
    else
    {
        MessageBox.Show("Vous devez sélectionner une personne");
    }
}

private void MainForm_Load(object sender, EventArgs e)
{
    _PersonneManager = new Dtc.PersonneManager();
    _PersonneManager.Add(new Dtc.Personne { Nom = "Nom 1", Prenom = "Prenom1",
DateNaissance = DateTime.Now });
    ListBoxPersonnelInitialize();
}

private void ListBoxPersonnelInitialize()
{
    List<Dtc.Personne> personnes = _PersonneManager.GetPersonnes();


    PersonneListBox.Items.Clear();
    foreach (var item in personnes)
    {
        PersonneListBox.Items.Add(item);
    }
}
}
}

```

Exercice 5 :

Partie 1 :

Sur le clique du bouton [Ajouter] de la fenêtre MainForm, affichez la fenêtre PersonneDetailForm avec un formulaire vide, si l'utilisateur clique sur [Valider] ajoutez la personne à la liste des personnes (PersonneListBox) si l'utilisateur clique sur [Annuler] fermez simplement la fenêtre PersonneDetailForm.

 Pensez également à ajouter la personne via la couche BLL et Dal ;)

!! Pensez à vérifier que la personne est valide (IsValid)

Partie 2 :

Sur le clique du bouton [Modifier] de la fenêtre MainForm, affichez la fenêtre PersonneDetailForm avec un formulaire contenant la personne selectionnez dans la liste de personne (PersonneListBox), si l'utilisateur clique sur [Valider] mettez à jour la personne à la liste des personnes (PersonneListBox) si l'utilisateur clique sur [Annuler] fermez simplement la fenêtre PersonneDetailForm.

👤 Pensez également à modifier la personne via la couche Bll et Dal ;)

!! Pensez à vérifier que la personne est valide (isValid)

Exercice Bonus :

Ajoutez une liste d'adresse (composant ListBox) à l'écran PersonneDetailForm et ajouter un écran de détail pour l'affichage des adresses d'une personne, rappel le diagramme de classe de la couche Bo était le suivant dans le TP3:

