|  |
| --- |
| Le programme de Gestion De Pharmacie |

Préparé par

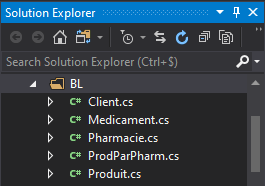
**Mohamed L’HARRAK**

Introduction

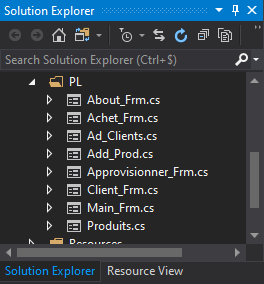
**Le programme de Gestion De Pharmacie est un programme qui permet de gestion et manager les produits et les clients de pharmacie**

Le programme donne un ensemble d’option pour mieux usage.

* Ajouter un nouveau client
* Ajouter un nouveau produit
* Option ajouter le solde.
* Afficher la liste des clients
* Afficher la liste de tous les produits
* Afficher la liste des produits medicaments
* Afficher la liste des produits parapharmaceutiques
* Effectuer un achat
* Approvisionner le stock d'un produit
* Rechercher un client
* Rechercher un produit



BL le fichier qui contient les classes



PL le fichier qui contient les Forms



menuStrip1

Label1

label2

btnUserMenu

btnFourMenu

btnCatMenu

btnProduitMenu



btnMenu

**Code de main\_form**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Windows.Forms;

using System.Collections;

using System.IO;

using System.Runtime.Serialization.Formatters.Binary;

namespace Gestion\_Pharmacie\_Console.PL

{

[Serializable]

public partial class Main\_Frm : Form

{

public static BL.Pharmacie ph = new BL.Pharmacie("Pharmacie AL CHIFA", "Reu d'Arabie Saoudite Tanger, Maroc.");

public void Enregistrer()

{

FileStream fs = new FileStream("Pharmacie.bin", FileMode.Create);

BinaryFormatter bf = new BinaryFormatter();

bf.Serialize(fs, ph);

fs.Close();

}

public BL.Pharmacie Ouvrire()

{

FileStream fs = new FileStream("Pharmacie.bin", FileMode.Open);

BinaryFormatter bf = new BinaryFormatter();

BL.Pharmacie l = (BL.Pharmacie)bf.Deserialize(fs);

fs.Close();

return l;

}

public Main\_Frm()

{

InitializeComponent();

}

private void btnProduitMenu\_Click(object sender, EventArgs e)

{

pnlBut.Top = btnProduitMenu.Top;

PL.Frm\_Produits frm = new PL.Frm\_Produits();

frm.Show();

}

private void btnCatMenu\_Click(object sender, EventArgs e)

{

pnlBut.Top = btnCatMenu.Top;

PL.Client\_Frm frm = new Client\_Frm();

frm.Show();

}

private void btnFourMenu\_Click(object sender, EventArgs e)

{

pnlBut.Top = btnFourMenu.Top;

PL.Achet\_Frm frm = new Achet\_Frm();

frm.Show();

}

private void btnUserMenu\_Click(object sender, EventArgs e)

{

pnlBut.Top = btnUserMenu.Top;

PL.Approvisionner\_Frm frm = new Approvisionner\_Frm();

frm.Show();

}

private void btnMenu\_Click(object sender, EventArgs e)

{

if (panel1.Width == 223)

{

panel1.Size = new Size(75, 537);

}

else

{

panel1.Size = new Size(223, 537);

}

}

private void btnParametre\_Click\_1(object sender, EventArgs e)

{

}

private void Main\_Frm\_Load(object sender, EventArgs e)

{

try

{

Main\_Frm.ph = this.Ouvrire();

// this.chargerData(this.lst);

}

catch (Exception err)

{

MessageBox.Show("Le fichier specifier n'existe pas", "Erreure", MessageBoxButtons.OK);

}

label1.Text = ph.Adresse;

label2.Text = ph.Nom;

}

private void panel3\_Paint(object sender, PaintEventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

Environment.Exit(0);

}

private void Main\_Frm\_FormClosing(object sender, FormClosingEventArgs e)

{

this.Enregistrer();

DialogResult result = MessageBox.Show("Voulez-vous vraiment quitter", "Attantion", MessageBoxButtons.YesNo, MessageBoxIcon.Exclamation);

if (result == DialogResult.No)

{

e.Cancel = true;

}

}

private void fournisseurToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.About\_Frm frm = new About\_Frm();

frm.Show();

}

private void ajouterUnNouveauProduitToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.Add\_Prod frm = new Add\_Prod();

frm.Show();

}

private void gestionDeProduitToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.Frm\_Produits frm = new PL.Frm\_Produits();

frm.Show();

}

private void acheterToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.Achet\_Frm frm = new Achet\_Frm();

frm.Show();

}

private void approvisionnerLeStockToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.Approvisionner\_Frm frm = new Approvisionner\_Frm();

frm.Show();

}

private void gestionDeCatégorieToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.Client\_Frm frm = new Client\_Frm();

frm.Show();

}

private void ajouterUnNouveauCatégoriesToolStripMenuItem\_Click(object sender, EventArgs e)

{

PL.Ad\_Clients frm = new Ad\_Clients();

frm.Show();

}

private void label1\_Click(object sender, EventArgs e)

{

}

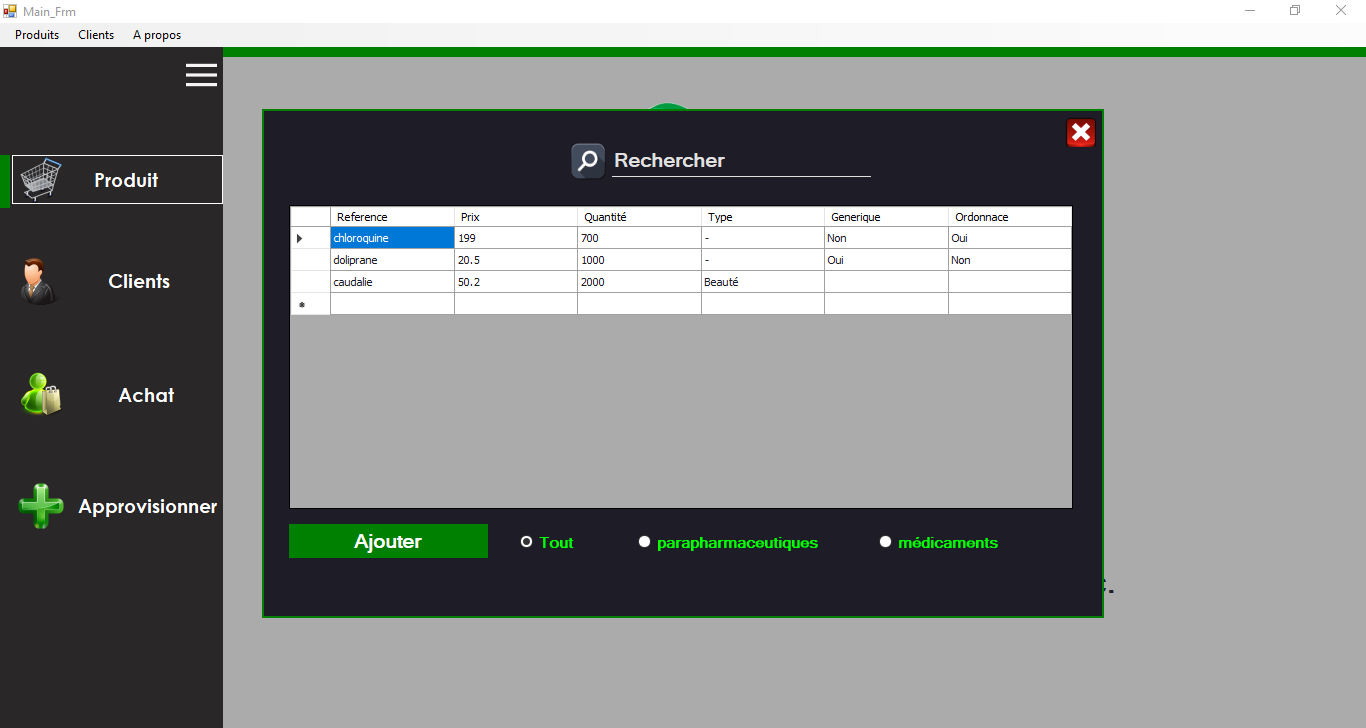
private void pictureBox2\_Click(object sender, EventArgs e)

{

}

}

}



Frm\_Produits

btnlogin

radioButton1

radioButton2

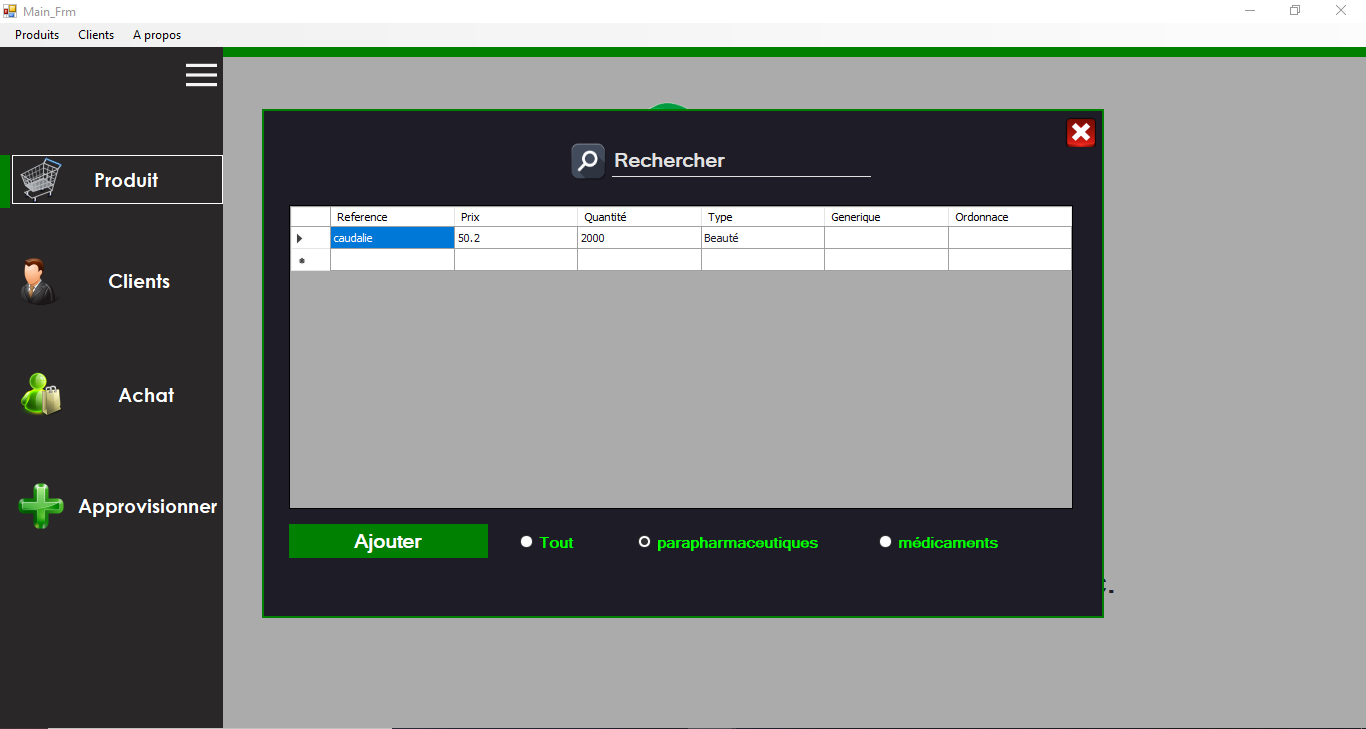
2

radioButton3

dataGridView1

button2

txtID



Afficher le form de ajouter les produits

Afficher les produits médicaments uniquement

Afficher les produits parapharmaceutiques uniquement

**Code de main form**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Gestion\_Pharmacie\_Console.PL

{

[Serializable]

public partial class Frm\_Produits : Form

{

public Frm\_Produits()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

}

private void txtID\_Enter(object sender, EventArgs e)

{

if (txtID.Text == "Rechercher")

{

txtID.Text = "";

txtID.ForeColor = Color.WhiteSmoke;

}

}

private void txtID\_TextChanged(object sender, EventArgs e)

{

}

private void radioButton2\_CheckedChanged(object sender, EventArgs e)

{

dataGridView1.Rows.Clear();

foreach (BL.Produit P in Main\_Frm.ph.Produits1)

{

if (P != null && P is BL.ProdParPharm)

{

DataGridViewRow ligne = (DataGridViewRow)dataGridView1.Rows[0].Clone();

ligne.Cells[0].Value = P.Reference;

ligne.Cells[1].Value = P.Prix;

ligne.Cells[2].Value = P.Qte;

ligne.Cells[3].Value = ((BL.ProdParPharm)P).Type;

dataGridView1.Rows.Add(ligne);

}

}

}

private void Frm\_Produits\_Load(object sender, EventArgs e)

{

this.chargerData();

}

private void panel6\_Paint(object sender, PaintEventArgs e)

{

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

this.chargerData();

}

private void btnlogin\_Click(object sender, EventArgs e)

{

this.Close();

PL.Add\_Prod frm = new Add\_Prod();

frm.Show();

}

private void button2\_Click(object sender, EventArgs e)

{

}

private void button2\_Click\_1(object sender, EventArgs e)

{

foreach (BL.ProdParPharm pa in Main\_Frm.ph.Produits1)

{

this.dataGridView1.Rows.Add(pa.Reference.ToString(), Convert.ToString(pa.Prix).ToString(), Convert.ToString(pa.Qte).ToString());

}

}

private void radioButton3\_CheckedChanged(object sender, EventArgs e)

{

dataGridView1.Rows.Clear();

foreach (BL.Produit P in Main\_Frm.ph.Produits1)

{

if (P != null && P is BL.Medicament)

{

DataGridViewRow ligne = (DataGridViewRow)dataGridView1.Rows[0].Clone();

ligne.Cells[0].Value = P.Reference;

ligne.Cells[1].Value = P.Prix;

ligne.Cells[2].Value = P.Qte;

ligne.Cells[3].Value = "-";

ligne.Cells[4].Value = ((BL.Medicament)P).gen;

ligne.Cells[5].Value = ((BL.Medicament)P).ord;

dataGridView1.Rows.Add(ligne);

}

}

}

public void chargerData()

{

dataGridView1.Rows.Clear();

foreach (BL.Produit P in Main\_Frm.ph.Produits1)

{

if (P != null)

{

DataGridViewRow ligne = (DataGridViewRow)dataGridView1.Rows[0].Clone();

ligne.Cells[0].Value = P.Reference;

ligne.Cells[1].Value = P.Prix;

ligne.Cells[2].Value = P.Qte;

if (P is BL.Medicament)

{

ligne.Cells[3].Value = "-";

ligne.Cells[4].Value = ((BL.Medicament)P).gen;

ligne.Cells[5].Value = ((BL.Medicament)P).ord;

}

if (P is BL.ProdParPharm)

{

//ligne.Cells[3].Value = "Produit Para";

ligne.Cells[3].Value = ((BL.ProdParPharm)P).Type;

}

dataGridView1.Rows.Add(ligne);

}

}

}

private void button2\_Click\_2(object sender, EventArgs e)

{

try

{

dataGridView1.Rows.Clear();

BL.Produit prod = Main\_Frm.ph.rechercherProduit(txtID.Text);

if (prod != null)

{

DataGridViewRow ligne = (DataGridViewRow)dataGridView1.Rows[0].Clone();

ligne.Cells[0].Value = prod.Reference;

ligne.Cells[1].Value = prod.Prix;

ligne.Cells[2].Value = prod.Qte;

if (prod is BL.Medicament)

{

ligne.Cells[3].Value = "-";

ligne.Cells[4].Value = ((BL.Medicament)prod).gen;

ligne.Cells[5].Value = ((BL.Medicament)prod).ord;

}

if (prod is BL.ProdParPharm)

{

//ligne.Cells[3].Value = "Produit Para";

ligne.Cells[3].Value = ((BL.ProdParPharm)prod).Type;

}

dataGridView1.Rows.Add(ligne);

}

}

catch

{

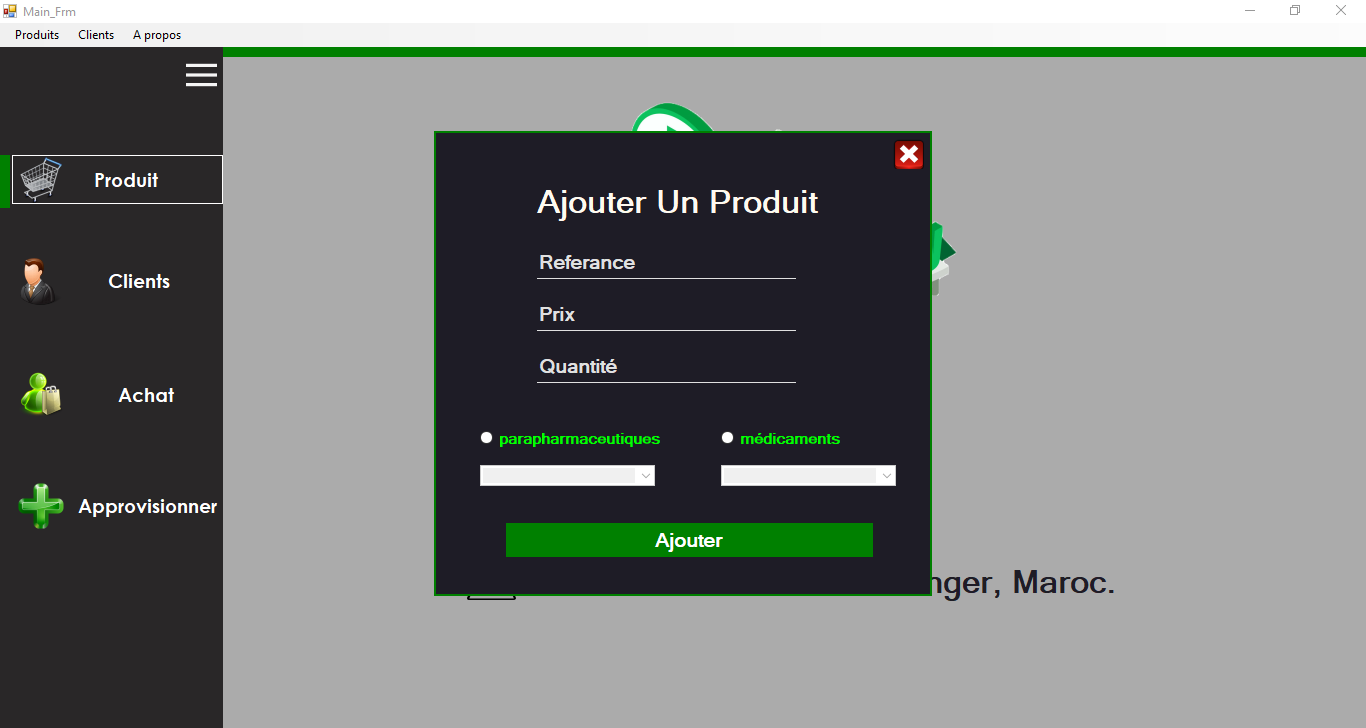
MessageBox.Show("ce prodtuit ne exist pas");

}

}

}

}



Add\_Prod

btnAdd

rdMed

rdParaPhar

cmbParaPhar

cmbMedi

txtQte

txtPrix

txtRef



**Code de Add\_Prod**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Gestion\_Pharmacie\_Console.PL

{

[Serializable]

public partial class Add\_Prod : Form

{

public Add\_Prod()

{

InitializeComponent();

}

private void txtRef\_TextChanged(object sender, EventArgs e)

{

}

private void txtRef\_Enter(object sender, EventArgs e)

{

if (txtRef.Text == "Referance")

{

txtRef.Text = "";

txtRef.ForeColor = Color.WhiteSmoke;

}

}

private void txtPrix\_TextChanged(object sender, EventArgs e)

{

}

private void txtPrix\_Enter(object sender, EventArgs e)

{

if (txtPrix.Text == "Prix")

{

txtPrix.Text = "";

txtPrix.ForeColor = Color.WhiteSmoke;

}

}

private void txtQte\_Enter(object sender, EventArgs e)

{

if (txtQte.Text == "Quantité")

{

txtQte.Text = "";

txtQte.ForeColor = Color.WhiteSmoke;

}

}

private void Add\_Prod\_Load(object sender, EventArgs e)

{

this.cmbMedi.Enabled = false;

this.cmbParaPhar.Enabled = false;

}

private void radioButton2\_CheckedChanged(object sender, EventArgs e)

{

this.cmbParaPhar.Enabled = true;

this.cmbMedi.Enabled = false;

cmbMedi.Text = "";

}

private void radioButton3\_CheckedChanged(object sender, EventArgs e)

{

this.cmbParaPhar.Enabled = false;

this.cmbMedi.Enabled = true;

cmbParaPhar.Text = "";

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

PL.Frm\_Produits frm = new Frm\_Produits();

frm.Show();

}

private void btnlogin\_Click(object sender, EventArgs e)

{

try

{

string Ref = txtRef.Text;

BL.Produit prodSearch = Main\_Frm.ph.rechercherProduit(Ref);

if (prodSearch == null)

{

double prixProd = Convert.ToDouble(txtPrix.Text);

int qteProd = int.Parse(txtQte.Text);

if (cmbParaPhar.SelectedIndex == cmbParaPhar.FindStringExact("Beauté"))

{

BL.ProdParPharm p = new BL.ProdParPharm(Ref, prixProd, qteProd, "Beauté");

Main\_Frm.ph.AjouterProduit(p);

}

if (cmbParaPhar.SelectedIndex == cmbParaPhar.FindStringExact("Cosmétiqeu"))

{

BL.ProdParPharm p = new BL.ProdParPharm(Ref, prixProd, qteProd, "Cosmétiqeu");

Main\_Frm.ph.AjouterProduit(p);

}

if (cmbParaPhar.SelectedIndex == cmbParaPhar.FindStringExact("Diététique"))

{

BL.ProdParPharm p = new BL.ProdParPharm(Ref, prixProd, qteProd, "Diététique");

Main\_Frm.ph.AjouterProduit(p);

}

if (cmbMedi.SelectedIndex == cmbMedi.FindStringExact("Generique"))

{

BL.Medicament p = new BL.Medicament(Ref, prixProd, qteProd, true, false);

Main\_Frm.ph.AjouterProduit(p);

}

if (cmbMedi.SelectedIndex == cmbMedi.FindStringExact("Ordannace"))

{

BL.Medicament p = new BL.Medicament(Ref, prixProd, qteProd, false, true);

Main\_Frm.ph.AjouterProduit(p);

}

MessageBox.Show("Produit ajouter avec succès !", "Ajouter", MessageBoxButtons.OK, MessageBoxIcon.Information);

txtRef.Text="Referance";

txtPrix.Text = "Prix";

txtQte.Text = "Quantité";

cmbMedi.Text = "";

cmbParaPhar.Text = "";

}

else

{

MessageBox.Show("ce produit exest diga !", "atantion", MessageBoxButtons.OK, MessageBoxIcon.Error);

txtRef.Focus();

txtRef.SelectionStart = 0;

txtRef.SelectionLength = txtRef.TextLength;

}

}

catch

{

MessageBox.Show("");

}

}

private void panel2\_Paint(object sender, PaintEventArgs e)

{

}

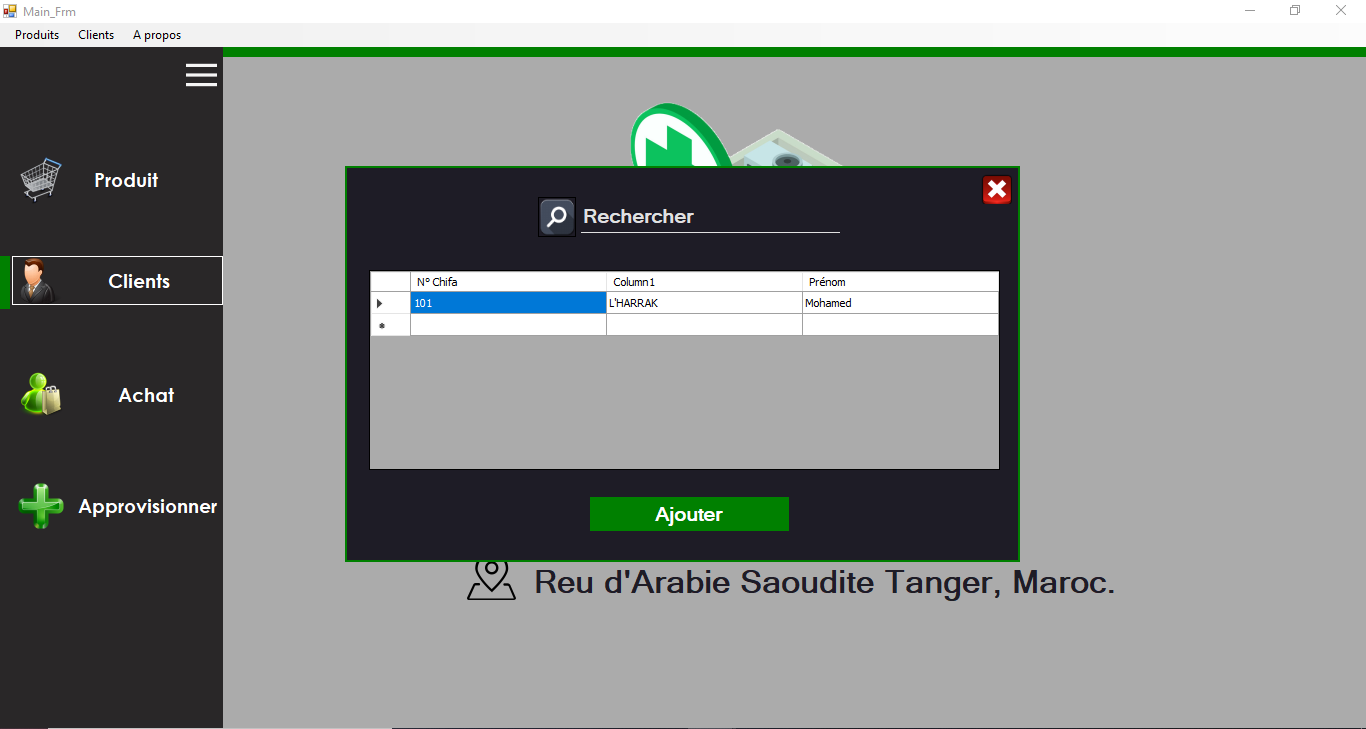
private void panel3\_Paint(object sender, PaintEventArgs e)

{

}

}

}



Client\_Frm

btnlogin

dataGridView1

button2

txtSearch

**Code de Client**\_**Frm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Gestion\_Pharmacie\_Console.PL

{

[Serializable]

public partial class Client\_Frm : Form

{

public Client\_Frm()

{

InitializeComponent();

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void btnlogin\_Click(object sender, EventArgs e)

{

this.Close();

PL.Ad\_Clients frm = new Ad\_Clients();

frm.Show();

}

private void txtID\_TextChanged(object sender, EventArgs e)

{

}

private void txtID\_Enter(object sender, EventArgs e)

{

}

private void txtSearch\_Enter(object sender, EventArgs e)

{

if (txtSearch.Text == "Rechercher")

{

txtSearch.Text = "";

txtSearch.ForeColor = Color.WhiteSmoke;

}

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

}

private void Client\_Frm\_Load(object sender, EventArgs e)

{

this.chargerData();

}

public void chargerData()

{

dataGridView1.Rows.Clear();

foreach (BL.Client c in Main\_Frm.ph.Clients1)

{

if (c != null)

{

DataGridViewRow ligne = (DataGridViewRow)dataGridView1.Rows[0].Clone();

ligne.Cells[0].Value = c.Chifa;

ligne.Cells[1].Value = c.Nom;

ligne.Cells[2].Value = c.Prenom;

dataGridView1.Rows.Add(ligne);

}

}

}

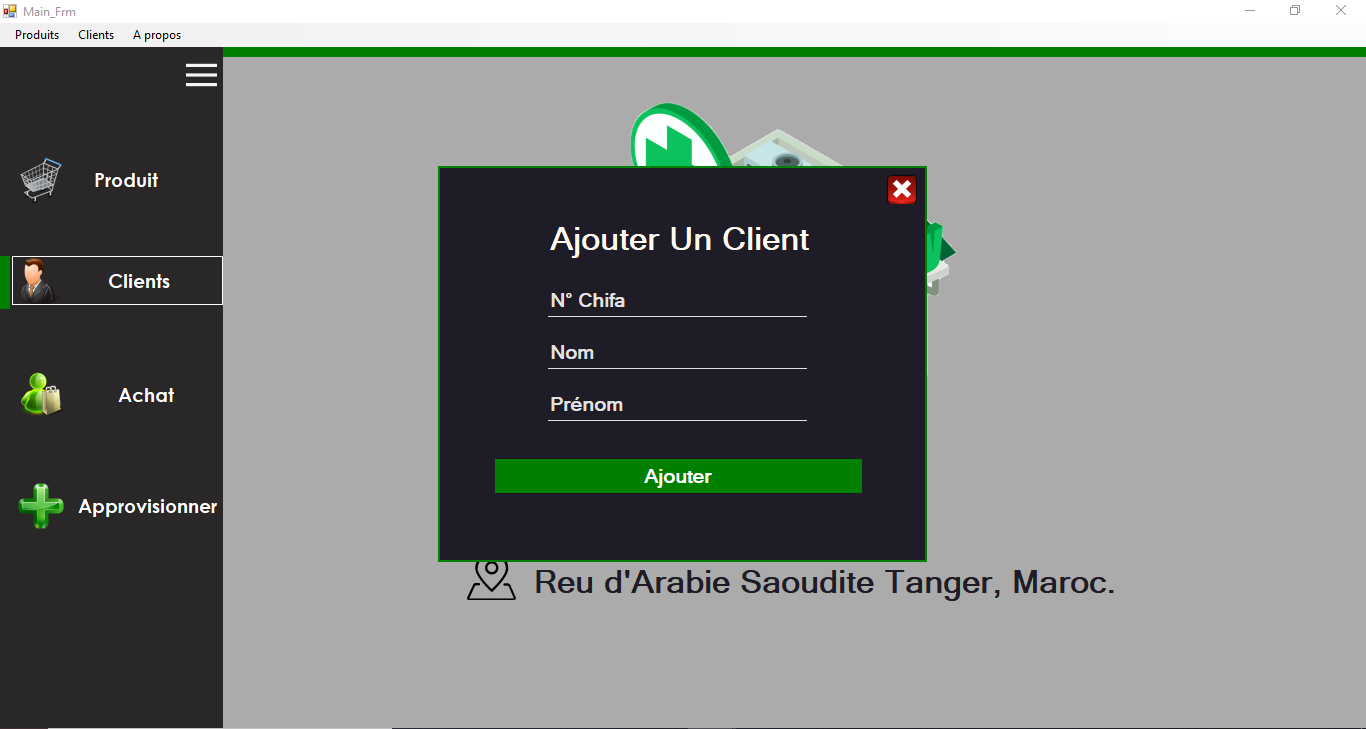
private void txtSearch\_TextChanged(object sender, EventArgs e)

{

}

}

}



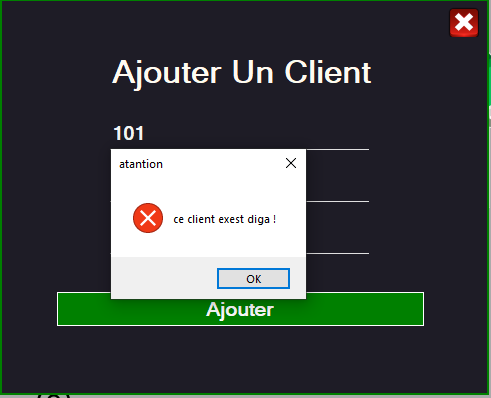
Ad\_Clients

btnAdd

txtPrenom

txtNom

txtChifa



**Code de Ad\_Clients**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Gestion\_Pharmacie\_Console.PL

{

[Serializable]

public partial class Ad\_Clients : Form

{

public Ad\_Clients()

{

InitializeComponent();

}

private void txtChifa\_Enter(object sender, EventArgs e)

{

if (txtChifa.Text == "N° Chifa")

{

txtChifa.Text = "";

txtChifa.ForeColor = Color.WhiteSmoke;

}

}

private void txtNom\_Enter(object sender, EventArgs e)

{

if (txtNom.Text == "Nom")

{

txtNom.Text = "";

txtNom.ForeColor = Color.WhiteSmoke;

}

}

private void txtPrenom\_Enter(object sender, EventArgs e)

{

if (txtPrenom.Text == "Prénom")

{

txtPrenom.Text = "";

txtPrenom.ForeColor = Color.WhiteSmoke;

}

}

private void btnAdd\_Click(object sender, EventArgs e)

{

try

{

int chifa = Convert.ToInt32(txtChifa.Text);

BL.Client prodSearch = Main\_Frm.ph.rechercherClient(chifa);

if (prodSearch == null)

{

string nom = txtNom.Text;

string prenom = txtPrenom.Text;

BL.Client c = new BL.Client(nom, prenom, chifa);

Main\_Frm.ph.AjouterClient(c);

MessageBox.Show("Produit ajouter avec succès !", "Informaton", MessageBoxButtons.OK, MessageBoxIcon.Information);

txtChifa.Text= "N° Chifa";

txtPrenom.Text = "Nom";

txtNom.Text = "Prénom";

}

else

{

MessageBox.Show("ce client exest diga !", "atantion", MessageBoxButtons.OK, MessageBoxIcon.Error);

txtChifa.Focus();

txtChifa.SelectionStart = 0;

txtChifa.SelectionLength = txtChifa.TextLength;

}

}

catch(Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void txtChifa\_TextChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

PL.Client\_Frm frm = new Client\_Frm();

frm.Show();

}

private void panel2\_Paint(object sender, PaintEventArgs e)

{

}

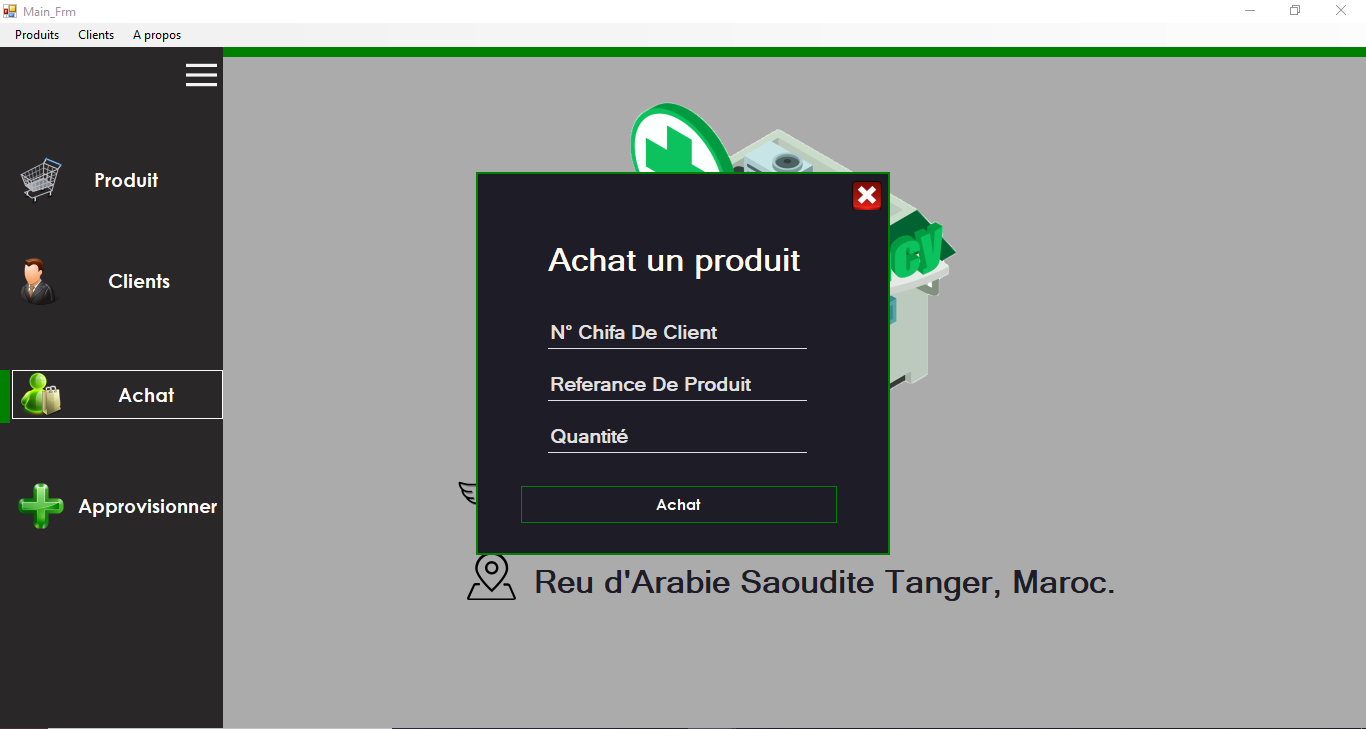
private void Ad\_Clients\_Load(object sender, EventArgs e)

{

}

}

}



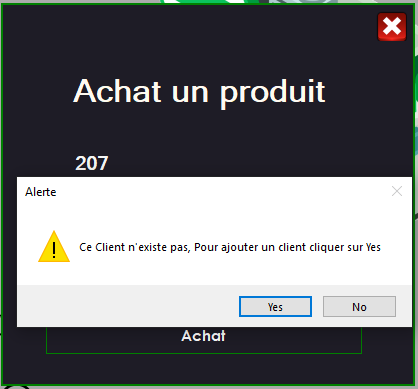
Achet\_Frm

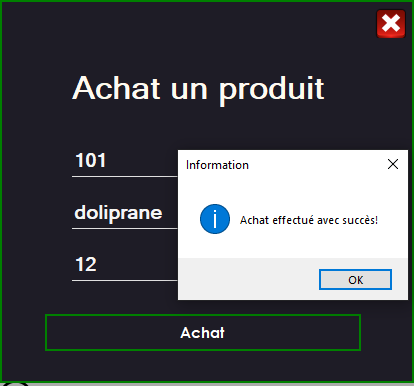
btnConx

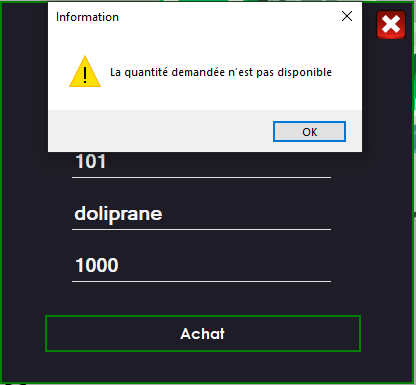
txtQte

txtRef

txtChifa







**Code de Achet\_Frm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Gestion\_Pharmacie\_Console.PL

{

[Serializable]

public partial class Achet\_Frm : Form

{

public Achet\_Frm()

{

InitializeComponent();

}

private void Achet\_Frm\_Load(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void txtChifa\_TextChanged(object sender, EventArgs e)

{

}

private void btnConx\_Click(object sender, EventArgs e)

{

string Reference = txtRef.Text;

BL.Produit prodSearch = Main\_Frm.ph.rechercherProduit(Reference);

int chifa = Convert.ToInt32(txtChifa.Text);

BL.Client clientSearch = Main\_Frm.ph.rechercherClient(chifa);

if (prodSearch != null && clientSearch!=null)

{

int qte = Convert.ToInt32(txtQte.Text);

if (Main\_Frm.ph.QuantiteOK(prodSearch, qte))

{

Main\_Frm.ph.Achat(clientSearch, prodSearch, qte);

MessageBox.Show("Achat effectué avec succès!", "Information", MessageBoxButtons.OK, MessageBoxIcon.Information);

txtChifa.Text = "N° Chifa De Client";

txtRef.Text = "Referance De Produit";

txtQte.Text = "Quantité";

}

else

{

MessageBox.Show("La quantité demandée n'est pas disponible","Information",MessageBoxButtons.OK,MessageBoxIcon.Warning);

}

}

}

private void txtRef\_Validated(object sender, EventArgs e)

{

try

{

string Reference = txtRef.Text;

BL.Produit prodSearch = Main\_Frm.ph.rechercherProduit(Reference);

if (prodSearch == null)

{

DialogResult result = MessageBox.Show("Ce produit n'existe pas, pour annuler cette opération cliquer sur Yes", "Alerte", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);

if(result== DialogResult.Yes)

{

this.Hide();

}

txtRef.Focus();

txtRef.SelectionStart = 0;

txtRef.SelectionLength = txtRef.TextLength;

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void txtChifa\_Enter(object sender, EventArgs e)

{

if (txtChifa.Text == "N° Chifa De Client")

{

txtChifa.Text = "";

txtChifa.ForeColor = Color.WhiteSmoke;

}

}

private void txtRef\_Enter(object sender, EventArgs e)

{

if (txtRef.Text == "Referance De Produit")

{

txtRef.Text = "";

txtRef.ForeColor = Color.WhiteSmoke;

}

}

private void txtQte\_TextChanged(object sender, EventArgs e)

{

}

private void txtChifa\_Validated(object sender, EventArgs e)

{

try

{

int chifa = Convert.ToInt32(txtChifa.Text);

BL.Client clientSearch = Main\_Frm.ph.rechercherClient(chifa);

if (clientSearch == null)

{

DialogResult result = MessageBox.Show("Ce Client n'existe pas, Pour ajouter un client cliquer sur Yes", "Alerte", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);

if (result == DialogResult.Yes)

{

PL.Ad\_Clients frm = new Ad\_Clients();

frm.Show();

this.Hide();

}

txtChifa.Focus();

txtChifa.SelectionStart = 0;

txtChifa.SelectionLength = txtRef.TextLength;

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

}

private void txtQte\_Enter(object sender, EventArgs e)

{

if (txtQte.Text == "Quantité")

{

txtQte.Text = "";

txtQte.ForeColor = Color.WhiteSmoke;

}

}

private void txtRef\_TextChanged(object sender, EventArgs e)

{

}

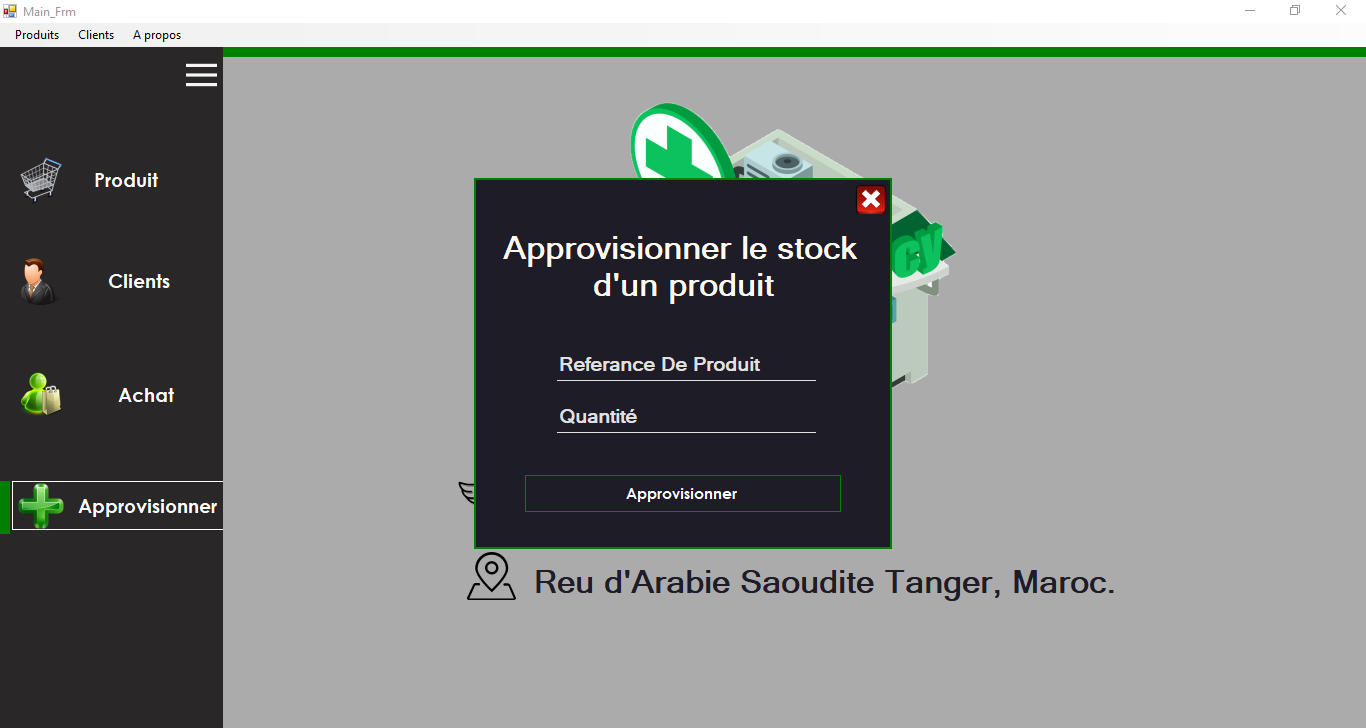
private void txtQte\_Validated(object sender, EventArgs e)

{

}

}

}



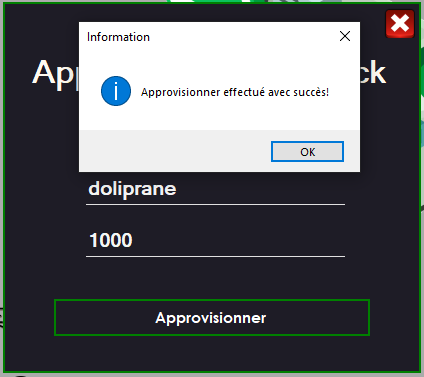
Approvisionner\_Frm

btnConx

txtQte

txtRef





**Code de Approvisionner\_Frm**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Gestion\_Pharmacie\_Console.PL

{

public partial class Approvisionner\_Frm : Form

{

public Approvisionner\_Frm()

{

InitializeComponent();

}

private void txtRef\_Enter(object sender, EventArgs e)

{

if (txtRef.Text == "Referance De Produit")

{

txtRef.Text = "";

txtRef.ForeColor = Color.WhiteSmoke;

}

}

private void txtQte\_Enter(object sender, EventArgs e)

{

if (txtQte.Text == "Quantité")

{

txtQte.Text = "";

txtQte.ForeColor = Color.WhiteSmoke;

}

}

private void txtRef\_Validated(object sender, EventArgs e)

{

try

{

string Reference = txtRef.Text;

BL.Produit prodSearch = Main\_Frm.ph.rechercherProduit(Reference);

if (prodSearch == null)

{

DialogResult result = MessageBox.Show("Ce produit n'existe pas, pour annuler cette opération cliquer sur Yes", "Alerte", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);

if (result == DialogResult.Yes)

{

this.Hide();

}

txtRef.Focus();

txtRef.SelectionStart = 0;

txtRef.SelectionLength = txtRef.TextLength;

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void btnConx\_Click(object sender, EventArgs e)

{

string Reference = txtRef.Text;

BL.Produit prodSearch = Main\_Frm.ph.rechercherProduit(Reference);

int qte = Convert.ToInt32(txtQte.Text);

if (prodSearch != null)

{

Main\_Frm.ph.Approvisionner(prodSearch, qte);

MessageBox.Show("Approvisionner effectué avec succès!", "Information", MessageBoxButtons.OK, MessageBoxIcon.Information);

txtRef.Text = "Referance De Produit";

txtQte.Text = "Quantité";

}

}

private void button1\_Click(object sender, EventArgs e)

{

this.Close();

}

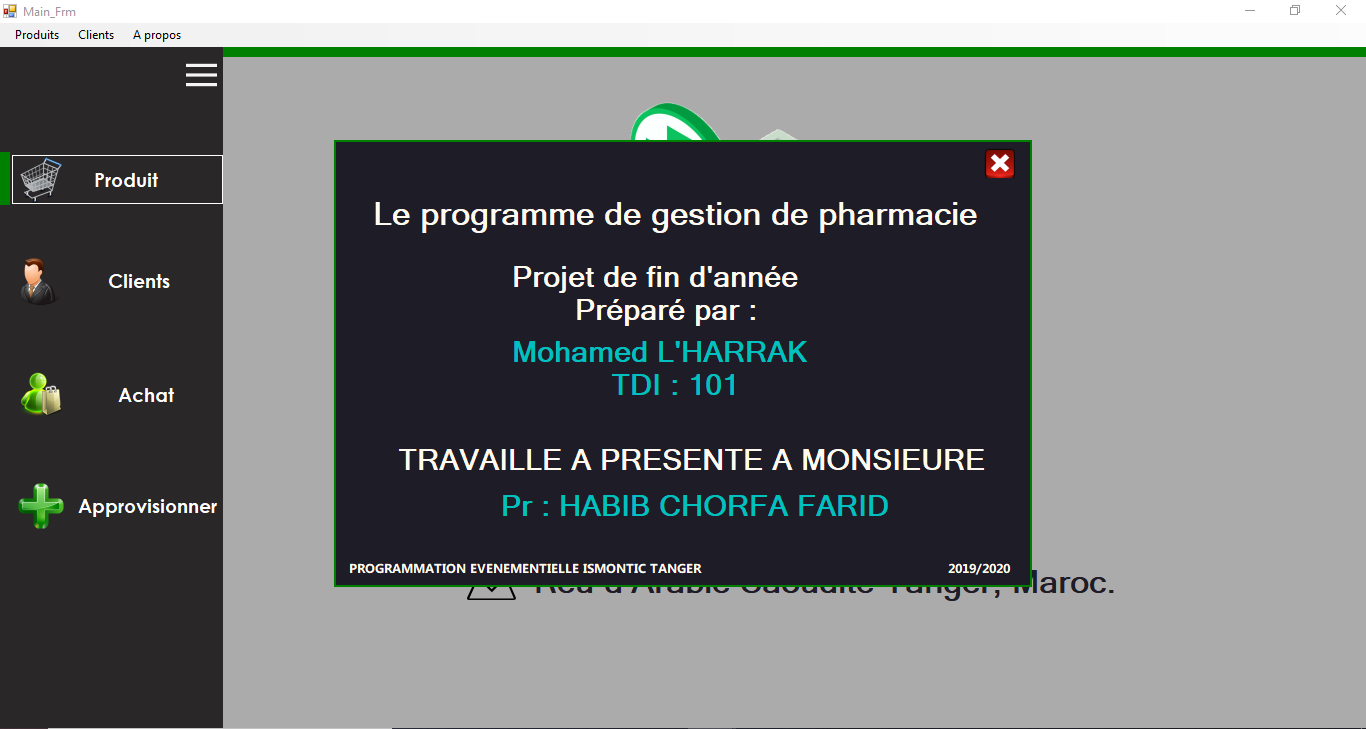
private void Approvisionner\_Frm\_Load(object sender, EventArgs e)

{

}

}

}



**Code de class produit**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Gestion\_Pharmacie\_Console.BL

{

[Serializable]

public class Produit

{

private string reference;

private double prix;

private int qte;

public string Reference

{

get

{

return reference;

}

set

{

reference = value;

}

}

public double Prix

{

get

{

return prix;

}

set

{

prix = value;

}

}

public int Qte

{

get

{

return qte;

}

set

{

qte = value;

}

}

public Produit(string reference, double prix, int qte)

{

this.reference = reference;

this.prix = prix;

this.qte = qte;

}

}

}

**Code de class Medicament**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Gestion\_Pharmacie\_Console.BL

{

[Serializable]

public class Medicament : Produit

{

private bool generique;

private bool ordonnance;

public string gen = "Non", ord = "Non";

public Medicament(string reference, double prix, int qte, bool generique, bool ordonnance) : base(reference, prix, qte)

{

this.generique = generique;

this.ordonnance = ordonnance;

if (this.generique)

gen = "Oui";

if (this.ordonnance)

ord = "Oui";

}

public bool Generique

{

get

{

return generique;

}

set

{

generique = value;

}

}

public bool Ordonnance

{

get

{

return ordonnance;

}

set

{

ordonnance = value;

}

}

}

}

**Code de class ProdParPharm**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Gestion\_Pharmacie\_Console.BL

{

[Serializable]

public class ProdParPharm : Produit

{

private string type;

public ProdParPharm(string reference, double prix, int qte, string type) : base(reference, prix, qte)

{

this.type = type;

}

public string Type

{

get

{

return type;

}

set

{

type = value;

}

}

}

}

**Code de class Client**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Gestion\_Pharmacie\_Console.BL

{

[Serializable]

public class Client

{

private string nom;

private string prenom;

private int chifa;

public string Nom

{

get

{

return nom;

}

set

{

nom = value;

}

}

public string Prenom

{

get

{

return prenom;

}

set

{

prenom = value;

}

}

public int Chifa

{

get

{

return chifa;

}

set

{

chifa = value;

}

}

public Client(string nom, string prenom, int chifa)

{

this.nom = nom;

this.prenom = prenom;

this.chifa = chifa;

}

}

}

**Code de class Pharmacie**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Gestion\_Pharmacie\_Console.BL

{

[Serializable]

public class Pharmacie

{

private string nom;

private string adresse;

private List<Client> clients=new List<Client>();

private List<Produit> produits = new List<Produit>();

public string Nom

{

get

{

return nom;

}

set

{

nom = value;

}

}

public string Adresse

{

get

{

return adresse;

}

set

{

adresse = value;

}

}

public List<Client> Clients1 { get => clients; set => clients = value; }

public List<Produit> Produits1 { get => produits; set => produits = value; }

public Pharmacie(string nom, string adresse)

{

this.nom = nom;

this.adresse = adresse;

this.Clients1 = new List<Client>();

this.Produits1 = new List<Produit>();

}

public void Achat(Client c, Produit p, int qte)

{

Produit posProduit = rechercherProduit(p.Reference);

if (posProduit != null)

{

posProduit.Qte -= qte;

}

}

public Client rechercherClient(int chifa)

{

foreach (Client c in this.clients)

if (c.Chifa == chifa)

return c;

return null;

}

public Produit rechercherProduit(string reference)

{

foreach (Produit p in this.produits)

if (p.Reference == reference)

return p;

return null;

}

public void Approvisionner(Produit p, int qte)

{

Produit posProduit = rechercherProduit(p.Reference);

if (posProduit != null)

posProduit.Qte += qte;

}

public void AjouterClient(Client c)

{

this.clients.Add(c);

}

public void AjouterProduit(Produit p)

{

this.Produits1.Add(p);

}

public bool QuantiteOK(Produit p, int qteDemande)

{

if (p.Qte >= qteDemande)

return true;

return false;

}

}

}