

```
1 using System;
2 using System.IO;
3 using System.Collections;
4 using System.Collections.Generic;
5 using System.Runtime.Serialization.Formatters.Binary;
6 using System.Windows.Forms;
7
8 namespace GestionCabinetMedical
9 {
10     [Serializable]
11     public partial class FormPatient : Form
12     {
13         Form1 f = new Form1();
14         CabinetMedical cm = new CabinetMedical();
15
16         private static int nbr;
17
18         public FormPatient()
19         {
20             InitializeComponent();
21         }
22         public FormPatient(Form1 f):this()
23         {
24             this.f=f;
25         }
26         private void FormPatient_Load(object sender, EventArgs e)
27         {
28             this.ImporterDataFromFile();
29             this.Vider();
30         }
31
32         private void BtnAjouter_Click(object sender, EventArgs e)
33         {
34             try
35             {
36                 if ((this.txtNom.Text.Length < 3) || this.txtNom.Text.Length > 15)
37                 {
38                     this.lblnote.Visible = true;
39                     //lblnote contient le message suivant : "les champs
40                     //contains "*" sont obligatoire"
41                     throw new Exception("Le nom qui vous avez entre est
42                     invalide");
43                 }
44                 string nom = this.txtNom.Text;
45
46                 if ((this.txtPrenom.Text.Length < 3) ||
47                     this.txtPrenom.Text.Length > 15)
48                 {
49                     this.lblnote.Visible = true;
50                     throw new Exception("Le prenom qui vous avez entre est
51                     invalide");
52                 }
53                 string prenom = this.txtPrenom.Text;
54
55                 if (this.dtpDateNaissance.Value.Date > DateTime.Now.Date)
```

```
52         {
53             this.lblnote.Visible = true;
54             throw new Exception("Le date qui vous avez entre est  ➤
invaide");
55         }
56         DateTime dateNaissance = this.dtpDateNaissance.Value.Date;
57
58         string sexe;
59         if (this.rbm.Checked == false && this.rbf.Checked == false)
60         {
61             this.lblnote.Visible = true;
62             throw new Exception("Please select your gender");
63         }
64         if (this.rbm.Checked == true)
65         {sexe = rbm.Text;}
66         else sexe = rbf.Text;
67
68         string adresse = this.txtAdresse.Text;
69
70         if (txtTelephone.Text.Length < 5 || txtTelephone.Text.Length > ➤
15)
71         {
72             this.lblnote.Visible = true;
73             throw new Exception("telephone incorrect");
74         }
75         string tele = txtTelephone.Text;
76
77         string email = this.txtEmail.Text;
78
79         nbr = this.f.Cbm.ListPatient.Count;
80         int codePatient = nbr++;
81
82         Patient pat = new Patient(codePatient, nom, prenom, ➤
dateNaissance, sexe, adresse, tele, email);
83
84         foreach (Patient p in this.f.Cbm.ListPatient)
85         {
86             if (pat.Equals(p)) throw new Exception("Patient est deja ➤
exist");
87         }
88
89         this.f.Cbm.ListPatient.Add(pat);
90         this.ChargerData();
91         this.Vider();
92     }
93     catch (Exception ex)
94     {
95         MessageBox.Show(ex.Message, "Error", ➤
MessageBoxButtons.OK, MessageBoxIcon.Information);
96     }
97 }
98 private void TxtTelephone_KeyPress(object sender, KeyPressEventArgs e)
99 {
100     if ((char.IsDigit(e.KeyChar)) || (char.IsControl(e.KeyChar))) ➤
101         e.Handled = false;
102     else e.Handled = true;
```

```
102     }
103     private void TxtNom_KeyPress(object sender, KeyPressEventArgs e)
104     {
105         if ((char.IsLetter(e.KeyChar)) || (char.IsControl(e.KeyChar))) ➤
106             e.Handled = false;
107         else e.Handled = true;
108     }
109     private void TxtPrenom_KeyPress(object sender, KeyPressEventArgs e)
110     {
111         if ((char.IsLetter(e.KeyChar)) || (char.IsControl(e.KeyChar))) ➤
112             e.Handled = false;
113         else e.Handled = true;
114     }
115     private void ChargerData()
116     {
117         this.dgPatient.Rows.Clear();
118         this.cmbCodePatient.Items.Clear();
119         foreach (Patient p in this.f.Cbm.ListPatient)
120         {
121             this.cmbCodePatient.Items.Add(p.CodePatient);
122             this.dgPatient.Rows.Add(p.CodePatient, p.Nom, p.Prenom, ➤
123                                     p.DateNaissance.ToShortDateString(), p.Sexe, p.Adresse, ➤
124                                     p.Tel, p.Email);
125         }
126     }
127     private void Vider()
128     {
129         txtNom.Text = null;
130         txtPrenom.Text = null;
131         txtAdresse.Text = null;
132         txtEmail.Text = null;
133         txtTelephone.Text = null;
134         rbf.Checked = false;
135         rbm.Checked = false;
136         cmbCodePatient.Text = null;
137         dtpDateNaissance.Value = DateTime.Now.Date;
138         lblnote.Visible = false;
139         dgPatient.ClearSelection();
140     }
141     private void BtnSupprimer_Click(object sender, EventArgs e)
142     {
143         DialogResult res = MessageBox.Show("Voulez vous vraiment ➤
144             supprimer ce patient ?", "Confirmation", ➤
145             MessageBoxButtons.OKCancel, MessageBoxIcon.Question);
146         if (res == DialogResult.OK)
147         {
148             try
149             {
150                 //supprimer par codePatient
151                 int indice = int.Parse ➤
152                     (cmbCodePatient.SelectedItem.ToString());
153                 Patient p = this.f.Cbm.TrouverPatient(indice);
154                 this.f.Cbm.ListPatient.Remove(p);
```

```
151         if(p.RendezVous!=null) this.f.Cbm.ListRendezVous.Remove
152         (p.RendezVous);
153         this.ChargerData();
154         this.Vider();
155
156         //une autre metode - supprimer par datatagrid
157         //int indice = dgPatient.CurrentRow.Index;
158         //for (int i = 0; i < this.f.Cbm.ListPatient.Count; i++)
159         //{
160             if (this.f.Cbm.ListPatient[i].CodePatient.ToString()
161             == dgPatient.Rows[indice].Cells[0].Value.ToString())
162             {
163                 RendezVous rv = this.f.Cbm.TrouverRendezVous
164                 (this.f.Cbm.ListPatient[i].CodePatient);
165                 if (rv!=null) this.f.Cbm.ListRendezVous.Remove
166                 (rv);
167                 this.f.Cbm.ListPatient.RemoveAt(i);
168             }
169         }
170     }
171 }
172 private void BtnExporter_Click(object sender, EventArgs e)
173 {
174     this.SaveDataInFile();
175 }
176 private void BtnImporter_Click(object sender, EventArgs e)
177 {
178     this.ImporterDataFromFile();
179 }
180 public void ImporterDataFromFile()
181 {
182     if (File.Exists("DataBase.bin"))
183     {
184         BinaryFormatter f = new BinaryFormatter();
185         FileStream fs = new FileStream("DataBase.bin", FileMode.Open,
186         FileAccess.Read);
187         this.f.Cbm.ListPatient = (List<Patient>)f.Deserialize(fs);
188         ChargerData();
189         fs.Close();
190     }
191 }
192 public void SaveDataInFile()
193 {
194     BinaryFormatter f = new BinaryFormatter();
195     FileStream fs = new FileStream("DataBase.bin",
196     FileMode.OpenOrCreate, FileAccess.Write);
197     f.Serialize(fs, this.f.Cbm.ListPatient);
198     fs.Close();
199 }
200 private void BtnFermer_Click(object sender, EventArgs e)
```

```
201     {
202         this.SaveDataInFile();
203         this.Close();
204     }
205
206     private void DgPatient_Click(object sender, EventArgs e)
207     {
208         try
209         {
210             int indice = dgPatient.CurrentRow.Index;
211             this.cmbCodePatient.SelectedItem = dgPatient.Rows[indice].Cells[0].Value;
212             this.txtNom.Text = dgPatient.Rows[indice].Cells[1].Value.ToString();
213             this.txtPrenom.Text = dgPatient.Rows[indice].Cells[2].Value.ToString();
214             this.dtpDateNaissance.Text = dgPatient.Rows[indice].Cells[3].Value.ToString();
215             if (dgPatient.Rows[indice].Cells[4].Value.ToString() == "M")
216             { this.rbM.Checked = true; }
217             else { this.rbF.Checked = true; }
218             this.txtAdresse.Text = dgPatient.Rows[indice].Cells[5].Value.ToString();
219             this.txtTelephone.Text = dgPatient.Rows[indice].Cells[6].Value.ToString();
220             this.txtEmail.Text = dgPatient.Rows[indice].Cells[7].Value.ToString();
221         }
222         catch (Exception ex) { }
223     }
224
225     private void CmbCodePatient_SelectedIndexChanged(object sender, EventArgs e)
226     {
227         int indice = cmbCodePatient.SelectedIndex;
228         if(indice!=-1)
229         {
230             dgPatient.ClearSelection();
231             dgPatient.Rows[indice].Selected = true;
232             this.txtNom.Text = dgPatient.Rows[indice].Cells[1].Value.ToString();
233             this.txtPrenom.Text = dgPatient.Rows[indice].Cells[2].Value.ToString();
234             this.dtpDateNaissance.Text = dgPatient.Rows[indice].Cells[3].Value.ToString();
235             if (dgPatient.Rows[indice].Cells[4].Value.ToString() == "M")
236             { this.rbM.Checked = true; }
237             else { this.rbF.Checked = true; }
238             this.txtAdresse.Text = dgPatient.Rows[indice].Cells[5].Value.ToString();
239             this.txtTelephone.Text = dgPatient.Rows[indice].Cells[6].Value.ToString();
240             this.txtEmail.Text = dgPatient.Rows[indice].Cells[7].Value.ToString();
241         }
242     }
```

```
241
242     private void BtnModifier_Click_1(object sender, EventArgs e)
243     {
244         try
245         {
246             DialogResult res = MessageBox.Show("Voulez vous vraiment  
modifier ce patient ?", "Confirmation",  
MessageBoxButtons.OKCancel, MessageBoxIcon.Question);
247             if (res == DialogResult.OK)
248             {
249
250                 int codePatient = int.Parse  
(cmbCodePatient.SelectedItem.ToString());
251
252                 if ((this.txtNom.Text.Length < 3) ||  
this.txtNom.Text.Length > 15)
253                 {
254                     this.lblnote.Visible = true;
255                     throw new Exception("Le nom qui vous avez entre est  
invaide");
256                 }
257                 string nom = this.txtNom.Text;
258
259                 if ((this.txtPrenom.Text.Length < 3) ||  
this.txtPrenom.Text.Length > 15)
260                 {
261                     this.lblnote.Visible = true;
262                     throw new Exception("Le prenom qui vous avez entre est  
invaide");
263                 }
264                 string prenom = this.txtPrenom.Text;
265
266                 if (this.dtpDateNaissance.Value.Date > DateTime.Now.Date)
267                 {
268                     this.lblnote.Visible = true;
269                     throw new Exception("Le date qui vous avez entre est  
invaide");
270                 }
271                 DateTime dateNaissance = this.dtpDateNaissance.Value.Date;
272
273                 string sexe;
274                 if (this.rbM.Checked == false && this.rbF.Checked ==  
false)
275                 {
276                     this.lblnote.Visible = true;
277                     throw new Exception("Please select your gender");
278                 }
279                 if (this.rbM.Checked == true)
280                 { sexe = rbM.Text; }
281                 else sexe = rbF.Text;
282
283                 string adresse = this.txtAdresse.Text;
284
285                 if (txtTelephone.Text.Length < 5 ||  
txtTelephone.Text.Length > 15)
286                 {
```

```
287         this.lblnote.Visible = true;
288         throw new Exception("telephone incorrect");
289     }
290     string tele = txtTelephone.Text;
291
292     string email = this.txtEmail.Text;
293
294     this.f.Cbm.ListPatient.Remove(this.f.Cbm.TrouverPatient  ➤
        (codePatient));
295
296     Patient pat = new Patient(codePatient, nom, prenom,  ➤
        dateNaissance, sexe, adresse, tele, email);
297
298     this.f.Cbm.ListPatient.Add(pat);
299     this.ChargerData();
300     this.Vider();
301     }
302 }
303 catch (Exception ex)
304 {
305     MessageBox.Show(ex.Message, "Error", MessageBoxButtons.OK,  ➤
        MessageBoxIcon.Information);
306 }
307 }
308 }
309 }
310
```