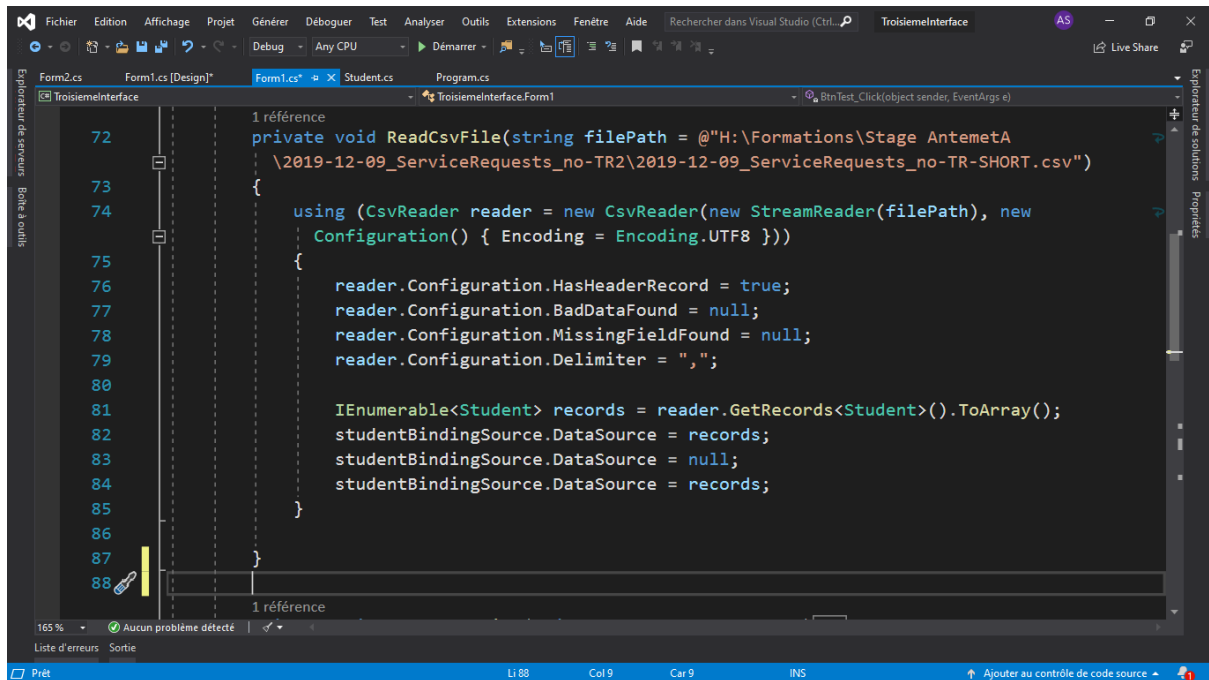


## Présentation de la version finale du code :

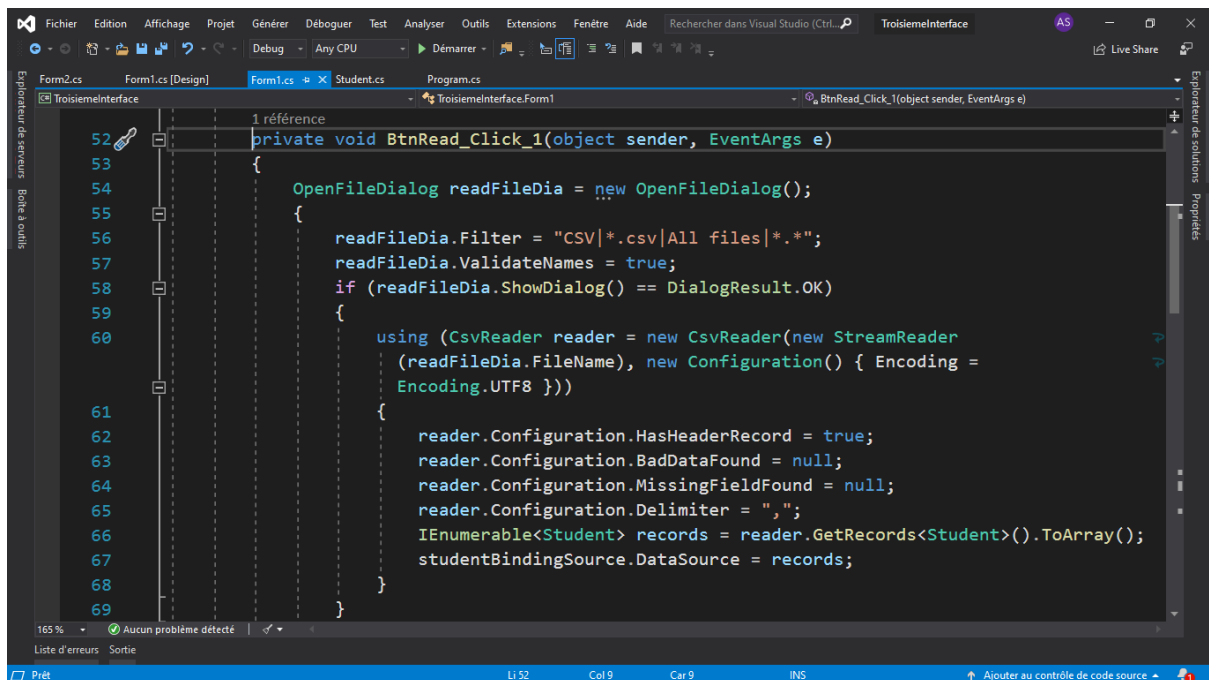
Code qui s'exécute à l'ouverture du logiciel et qui charge directement le fichier choisi :



The screenshot shows the Visual Studio IDE with the file `Form1.cs` open. The code defines a private method `ReadCsvFile` that takes a `filePath` parameter. It uses a `CsvReader` to read the CSV file and binds the data to a `studentBindingSource`. The status bar at the bottom indicates 'Prêt' and 'Aucun problème détecté'.

```
1 référence
72 private void ReadCsvFile(string filePath = @"H:\Formations\Stage Antemeta
73 \2019-12-09_ServiceRequests_no-TR2\2019-12-09_ServiceRequests_no-TR-SHORT.csv")
74 {
75     using (CsvReader reader = new CsvReader(new StreamReader(filePath), new
76         Configuration() { Encoding = Encoding.UTF8 }))
77     {
78         reader.Configuration.HasHeaderRecord = true;
79         reader.Configuration.BadDataFound = null;
80         reader.Configuration.MissingFieldFound = null;
81         reader.Configuration.Delimiter = ",";
82
83         IEnumerable<Student> records = reader.GetRecords<Student>().ToArray();
84         studentBindingSource.DataSource = records;
85         studentBindingSource.DataSource = null;
86         studentBindingSource.DataSource = records;
87     }
88 }
1 référence
```

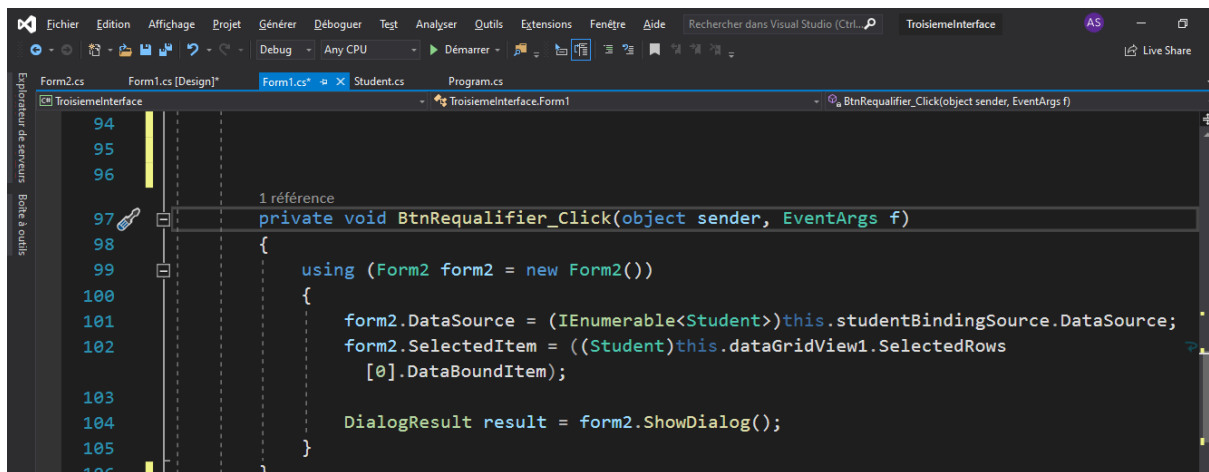
Code du bouton « Read » :



The screenshot shows the Visual Studio IDE with the file `Form1.cs` open. The code defines a private method `BtnRead_Click_1` that handles the click event of the 'Read' button. It uses an `OpenFileDialog` to select a file, then uses a `CsvReader` to read the CSV file and binds the data to a `studentBindingSource`. The status bar at the bottom indicates 'Prêt' and 'Aucun problème détecté'.

```
1 référence
52 private void BtnRead_Click_1(object sender, EventArgs e)
53 {
54     OpenFileDialog readFileDia = new OpenFileDialog();
55     {
56         readFileDia.Filter = "CSV|*.csv|All files|*.*";
57         readFileDia.ValidateNames = true;
58         if (readFileDia.ShowDialog() == DialogResult.OK)
59         {
60             using (CsvReader reader = new CsvReader(new StreamReader
61                 (readFileDia.FileName), new Configuration() { Encoding =
62                     Encoding.UTF8 }))
63             {
64                 reader.Configuration.HasHeaderRecord = true;
65                 reader.Configuration.BadDataFound = null;
66                 reader.Configuration.MissingFieldFound = null;
67                 reader.Configuration.Delimiter = ",";
68                 IEnumerable<Student> records = reader.GetRecords<Student>().ToArray();
69                 studentBindingSource.DataSource = records;
70             }
71         }
72     }
73 }
```

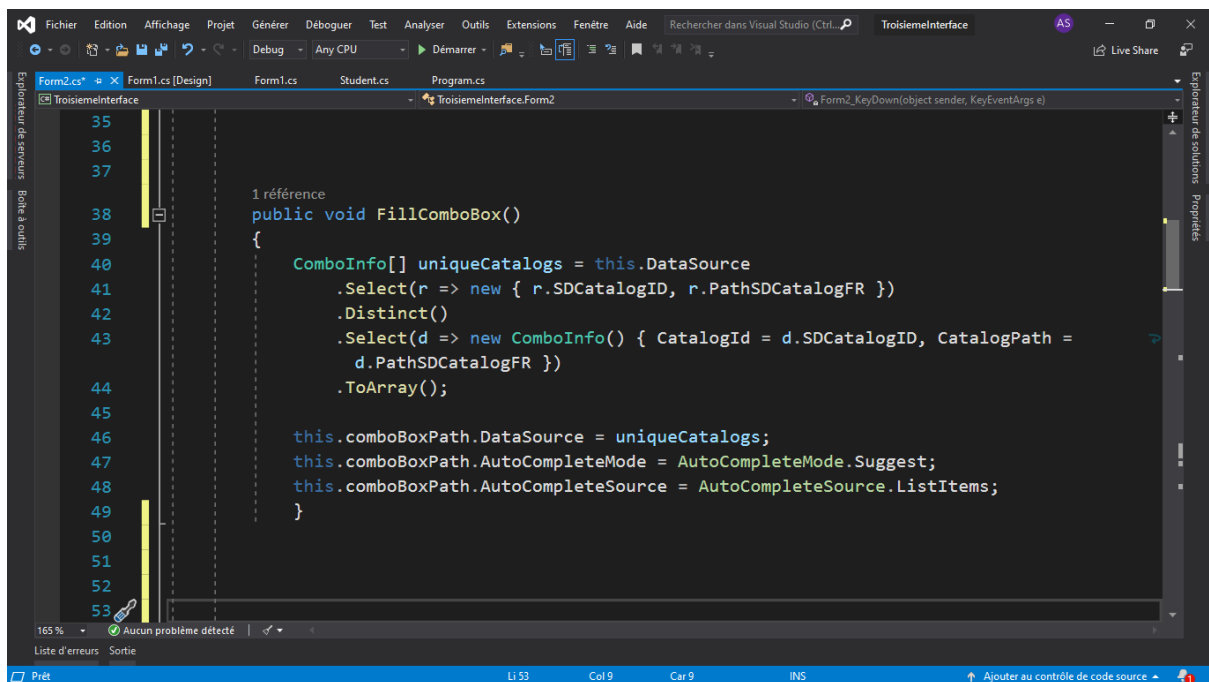
Code du bouton « Requalifier » :



```
1 référence
private void BtnRequalifier_Click(object sender, EventArgs f)
{
    using (Form2 form2 = new Form2())
    {
        form2.DataSource = (IEnumerable<Student>)this.studentBindingSource.DataSource;
        form2.SelectedItem = ((Student)this.dataGridView1.SelectedRows
            [0].DataBoundItem);

        DialogResult result = form2.ShowDialog();
    }
}
```

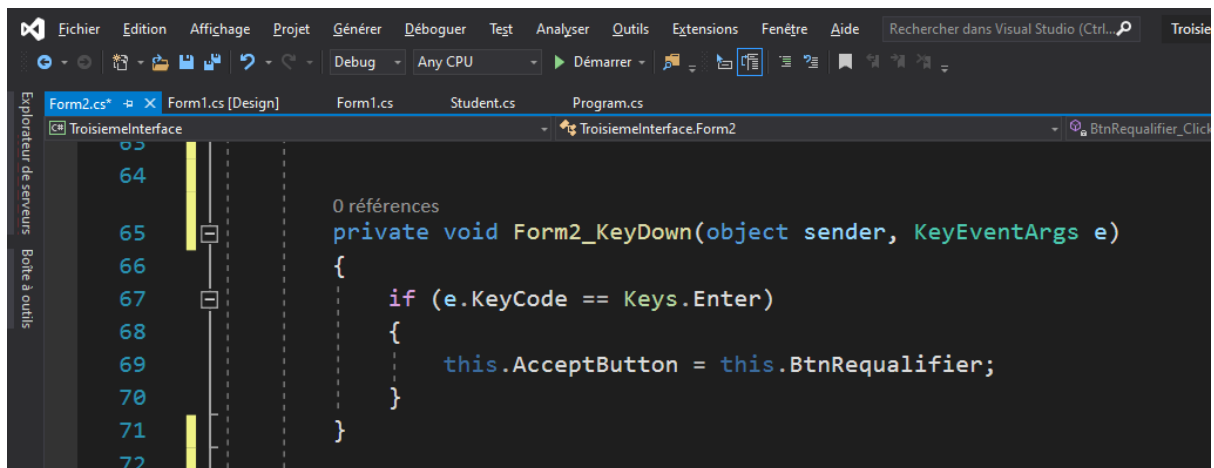
Code qui permet de remplir la combo box et de filtrer ses données :



```
1 référence
public void FillComboBox()
{
    ComboInfo[] uniqueCatalogs = this.DataSource
        .Select(r => new { r.SDCatalogID, r.PathSDCatalogFR })
        .Distinct()
        .Select(d => new ComboInfo() { CatalogId = d.SDCatalogID, CatalogPath =
            d.PathSDCatalogFR })
        .ToArray();

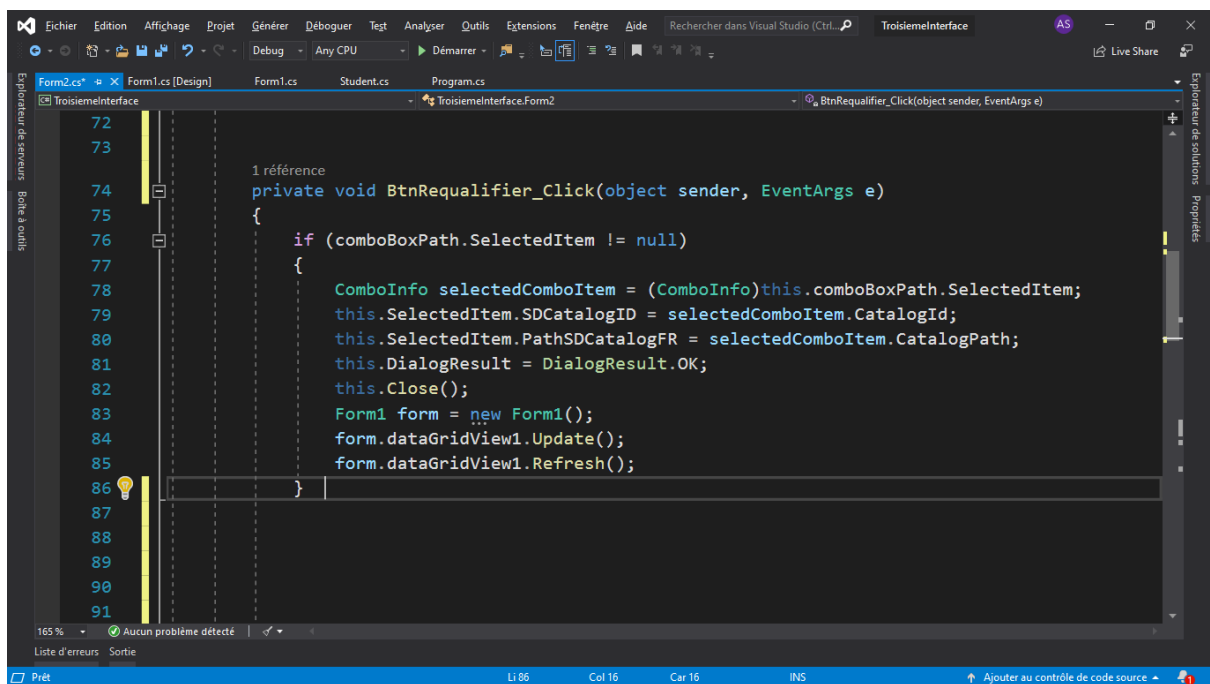
    this.comboBoxPath.DataSource = uniqueCatalogs;
    this.comboBoxPath.AutoCompleteMode = AutoCompleteMode.Suggest;
    this.comboBoxPath.AutoCompleteSource = AutoCompleteSource.ListItems;
}
```

Code qui permet de requalifier les tickets en appuyant sur « Entrée » :



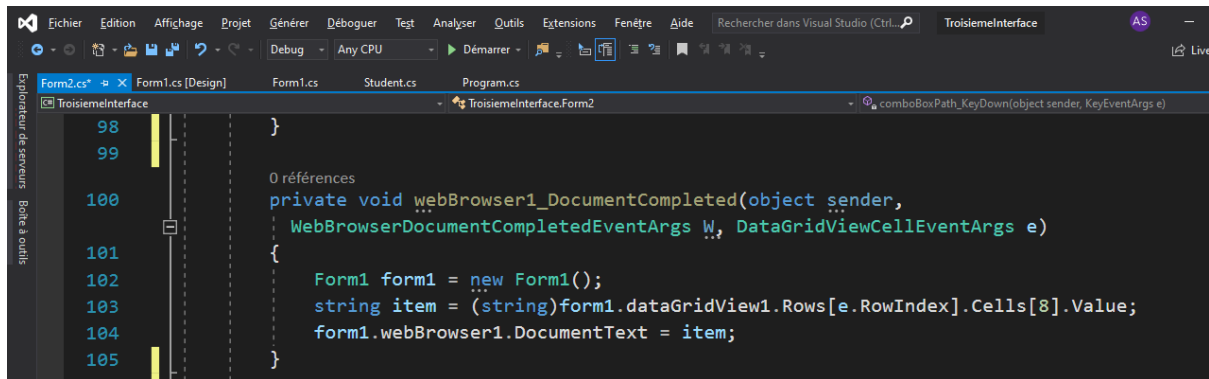
```
0 références
private void Form2_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        this.AcceptButton = this.BtnRequalifier;
    }
}
```

Code qui permet la requalification des tickets en prenant les informations dans la ligne choisie dans le DataGridView :



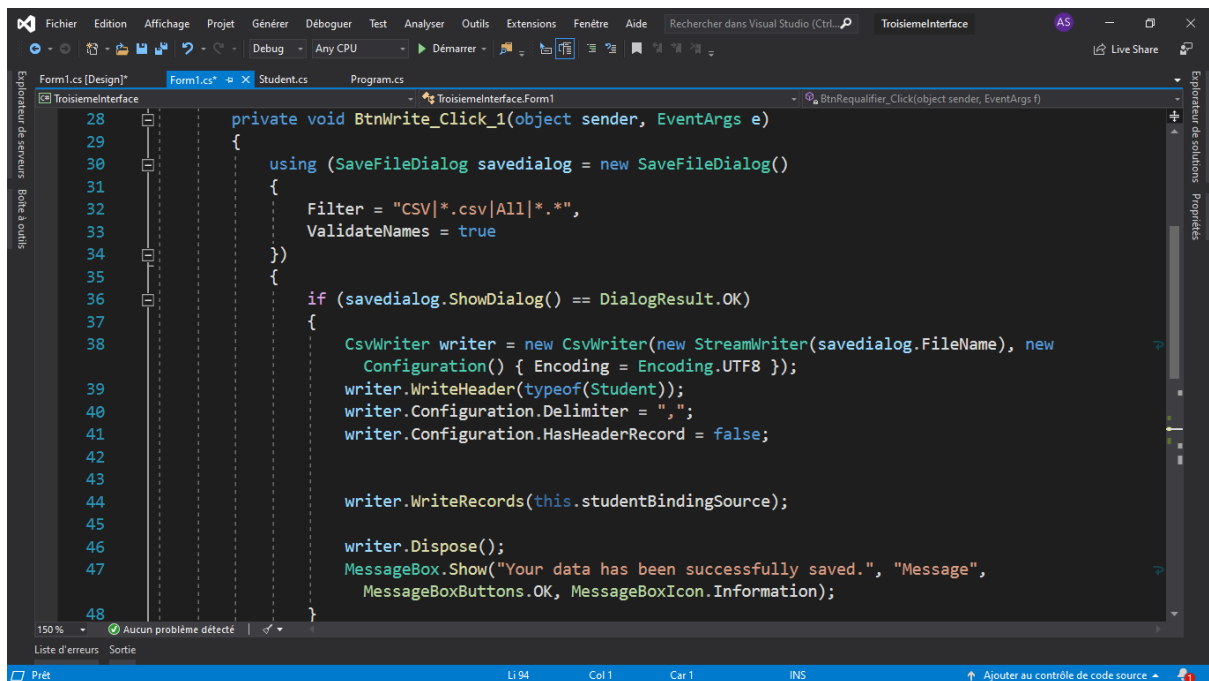
```
1 référence
private void BtnRequalifier_Click(object sender, EventArgs e)
{
    if (comboBoxPath.SelectedItem != null)
    {
        ComboInfo selectedComboItem = (ComboInfo)this.comboBoxPath.SelectedItem;
        this.SelectedItem.SDCatalogID = selectedComboItem.CatalogId;
        this.SelectedItem.PathSDCatalogFR = selectedComboItem.CatalogPath;
        this.DialogResult = DialogResult.OK;
        this.Close();
        Form1 form = new Form1();
        form.dataGridView1.Update();
        form.dataGridView1.Refresh();
    }
}
```

Code qui met à jour le DataGridView de la Form1 :



```
98 }
99
100 0 références
101 private void webBrowser1_DocumentCompleted(object sender,
102     WebBrowserDocumentCompletedEventArgs W, DataGridViewCellEventArgs e)
103 {
104     Form1 form1 = new Form1();
105     string item = (string)form1.dataGridView1.Rows[e.RowIndex].Cells[8].Value;
106     form1.webBrowser1.DocumentText = item;
107 }
```

Code qui permet d'enregistrer le nouveau fichier CSV et affiche que le fichier a bien été enregistré :



```
28 private void BtnWrite_Click_1(object sender, EventArgs e)
29 {
30     using (SaveFileDialog savedialog = new SaveFileDialog())
31     {
32         Filter = "CSV|*.csv|All|*.*",
33         ValidateNames = true
34     })
35     {
36         if (savedialog.ShowDialog() == DialogResult.OK)
37         {
38             CsvWriter writer = new CsvWriter(new StreamWriter(savedialog.FileName), new
39                 Configuration() { Encoding = Encoding.UTF8 });
40             writer.WriteHeader(typeof(Student));
41             writer.Configuration.Delimiter = ",";
42             writer.Configuration.HasHeaderRecord = false;
43
44             writer.WriteRecords(this.studentBindingSource);
45
46             writer.Dispose();
47             MessageBox.Show("Your data has been successfully saved.", "Message",
48                 MessageBoxButtons.OK, MessageBoxIcon.Information);
49         }
50     }
```