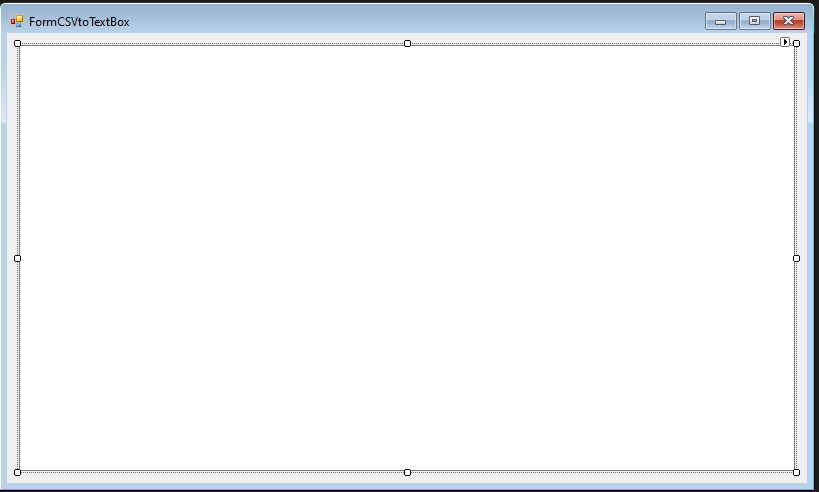
**Interprétation d’un fichier CSV**

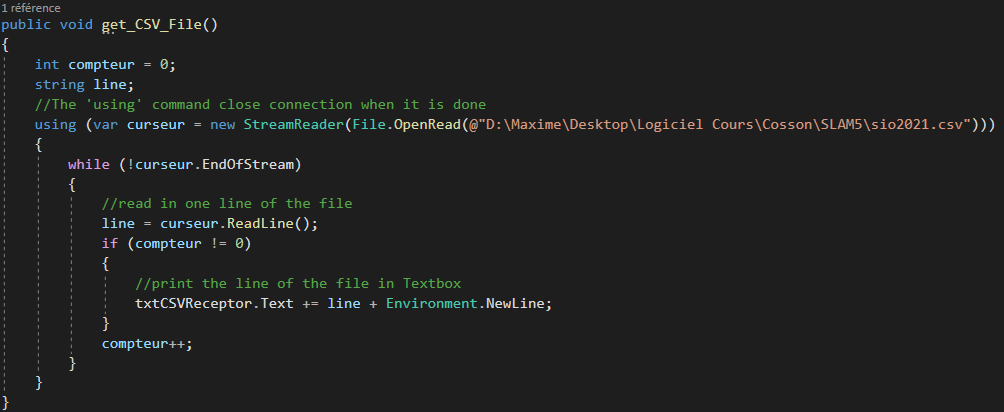
**Affichage dans textBox :**

Après création du Windows Form Application :

<https://stackoverflow.com/questions/14505233/how-can-i-read-csv-file-into-textboxes>



Code permettant de lire un ficher CSV (Excel) et de l’afficher :



Amélioration : (Attention modifier la police en « Courrier new » car police a caractère identique)



**Création de la classe « Etudiant » :**

class Etudiant

{

private string \_nom;

private string \_prenom;

private string \_adresseMail;

public string Nom { get => \_nom; set => \_nom = value; }

public string Prenom { get => \_prenom; set => \_prenom = value; }

public string AdresseMail { get => \_adresseMail; set => \_adresseMail = value; }

public Etudiant(string unNom = "", string unPrenom ="", string uneAdresseMail="")

{

this.\_nom = unNom;

this.\_prenom = unPrenom;

this.\_adresseMail = uneAdresseMail;

}

public Etudiant(string uneLigneCSV)

{

string[] champs = uneLigneCSV.Split(';');

this.\_nom = champs[0];

this.\_prenom = champs[1];

this.\_adresseMail = champs[2];

}

override

public string ToString()

{

const string format = "{0,-30} {1,-30} {2,-30}";

return string.Format(format, this.\_nom , this.\_prenom , this.\_adresseMail);

}

}

Ici l’utilisation d’un format est utile pour définir un type d’affichage précis

(Ici le format est défini de manière simple : { [n°de l’argument], [marge] }  
si la marge est Positive l’écriture se positionne sur la droite, à l’inverse à gauche si négative)

**Affichage dans DataGridView :**

(PS : Ici l’utilisation d’un DataGridNavigator et d’un objet BindingSource dans le formulaire n’est pas nécessaire, c’est pour cela que je me suis permis de faire un objet Binding source directement dans le code en supprimant le DataGridNavigator et le BindingSource de celui-ci)

public partial class FormDataGridView : Form

{

private List<Etudiant> \_etudiants;

private BindingSource \_maSource;

public FormDataGridView()

{

InitializeComponent();

\_etudiants = new List<Etudiant>();

\_maSource = new BindingSource();

}

public void get\_CSV\_File()

{

int compteur = 0;

string line;

//The 'using' command close connection when it is done

using (var curseur = new StreamReader(File.OpenRead(@"D:\Maxime\Desktop\Logiciel Cours\Cosson\SLAM5\sio2021.csv")))

{

while (!curseur.EndOfStream)

{

//read in one line of the file

line = curseur.ReadLine();

Etudiant monEtudiant = new Etudiant(line);

if (compteur != 0)

{

\_etudiants.Add(monEtudiant);

}

compteur++;

}

}

}

private void FormDataGridView\_Load(object sender, EventArgs e)

{

get\_CSV\_File();

dataGridView1.DataSource = \_maSource;

\_maSource.DataSource = \_etudiants;

}

}

**Résultat :**