IronXL.Excel Guide

Version 2019.5.2

Contents

Introduction

<-- Text in introduction part -->

And you can visit https://ironsoftware.com/csharp/excel/ for more information.

You can download sample project from GitHub
(https://github.com/magedo93/IronSoftware.git)

Document Organization

- Chapter 1 Install IronXL.Excel: this part describes How to install IronXL.Excel to existing project.
- Chapter 2 Basic Operations: this part describes basic operation with excel create or Open workbook, select sheet, select cell save workbook
- Chapter 3 Advanced Sheet Operations: this part describes how to different manipulation capabilities like adding headers or footers, mathematical operations files, and other features.
- Summery
  brief conclusion about what we have learned in this document
- About author
  brief about document author

Chapter 1: Install IronXL.Excel

IronXL.Excel can be installed and used on all of .NET projects type like windows application, ASP.NET MVC and .Net Core Application.

To add IronXL.Excel library to the project we have two ways, from Visual studio editor install using NuGet or command line using package console manager as following: -
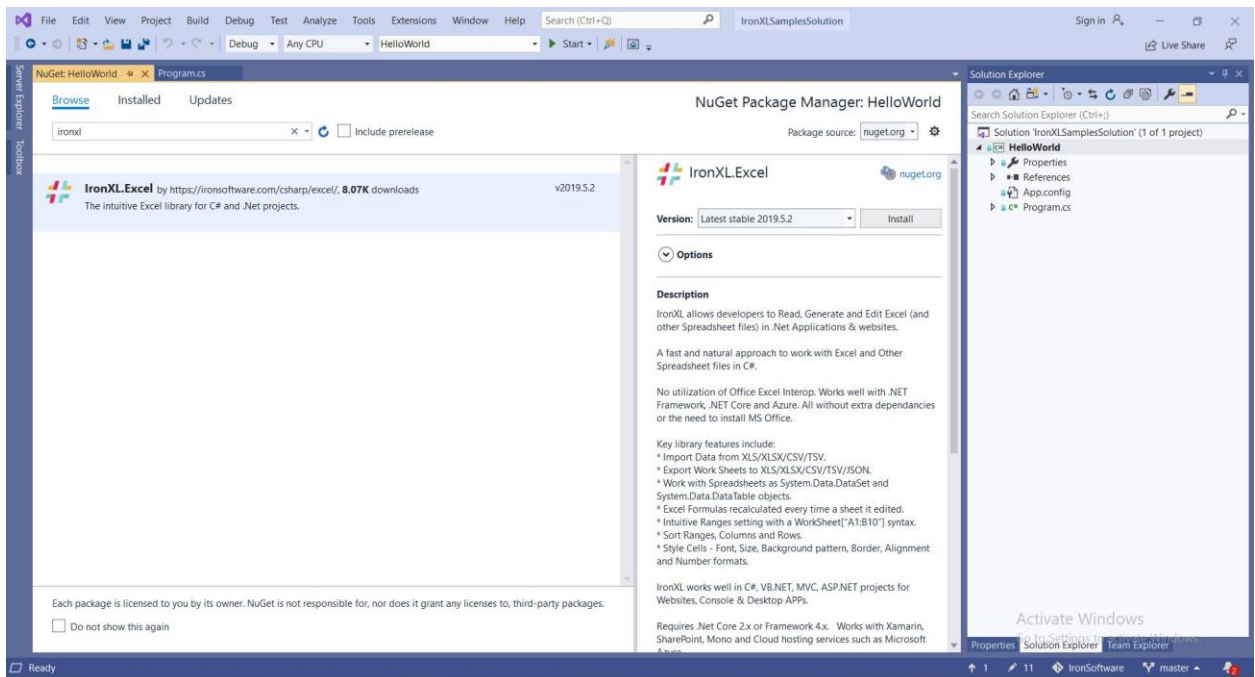
Install using NuGet

To add IronXL.Excel library to our project using NuGet we can do it using visualized interface (NuGet Package Manager) or by command using Package Manager Console as following: -
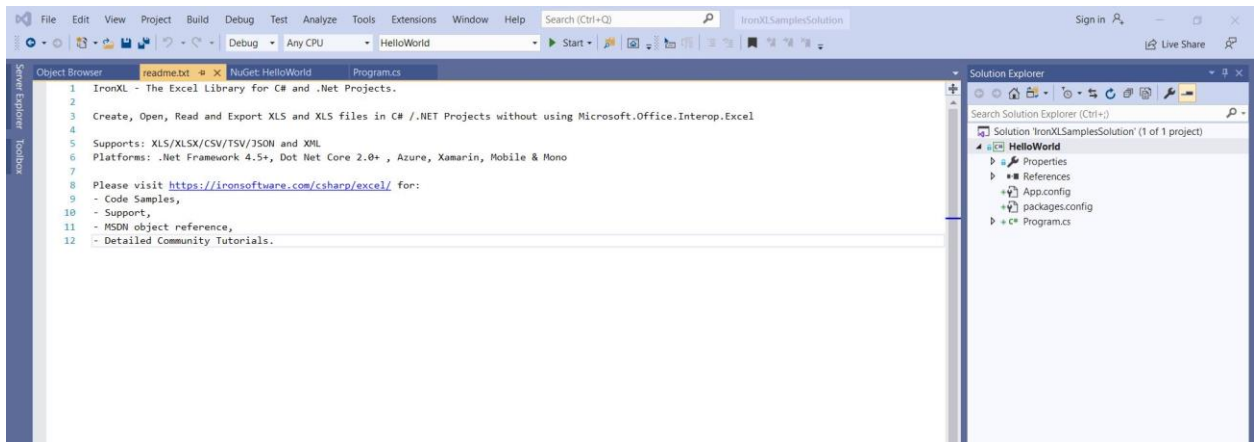
Using NuGet Package Manager

1- Using mouse -> right click on project name -> Select manage NuGet Package
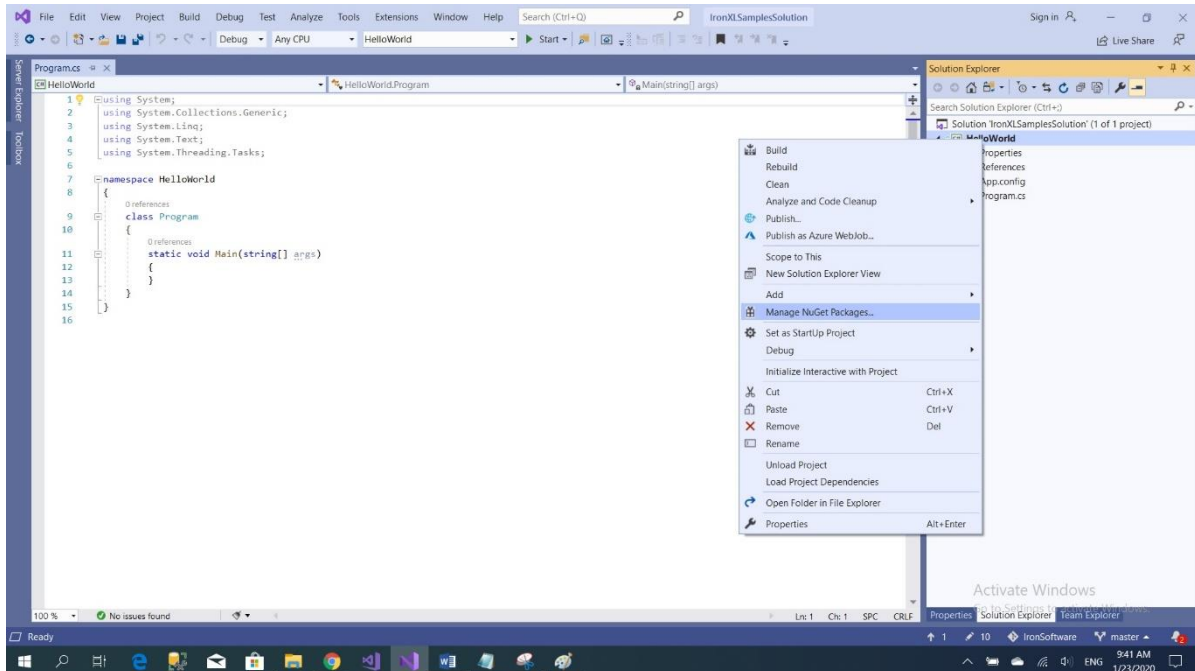
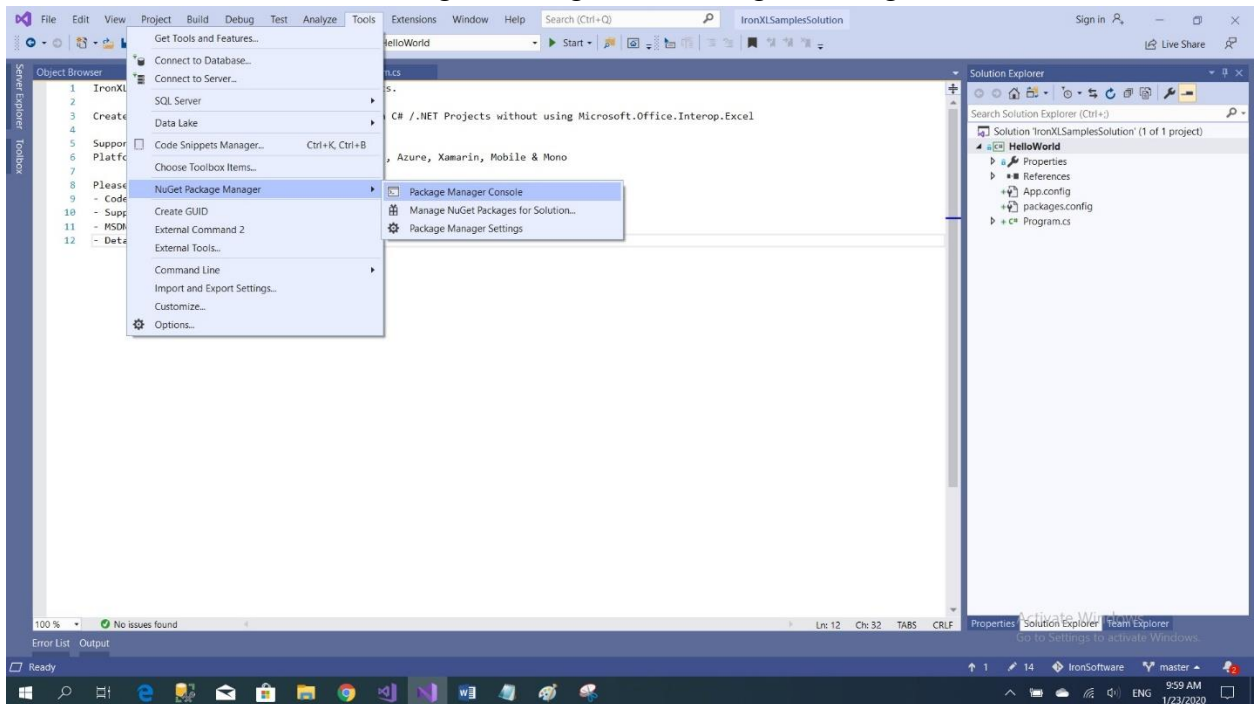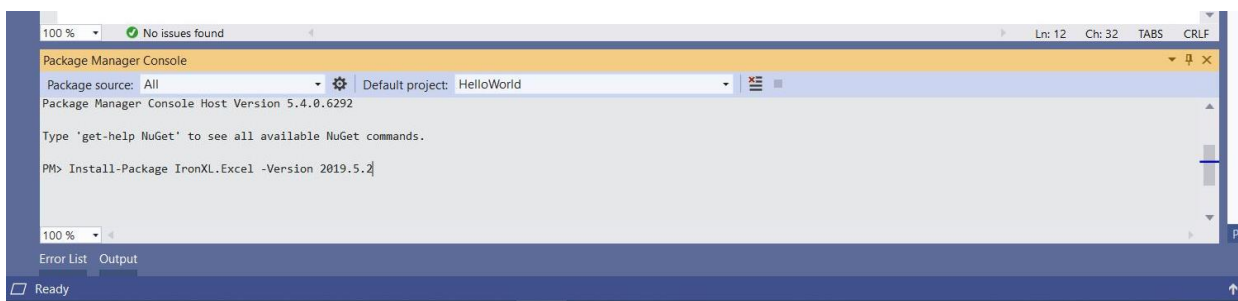2- From brows tab -> search for IronXL.Excel -> Install

3- And we are Done

Using NuGet Package Console manager

1- From tools -> NuGet Package Manager -> Package Manager Console

2- Run command -> Install-Package IronXL.Excel -Version 2019.5.2

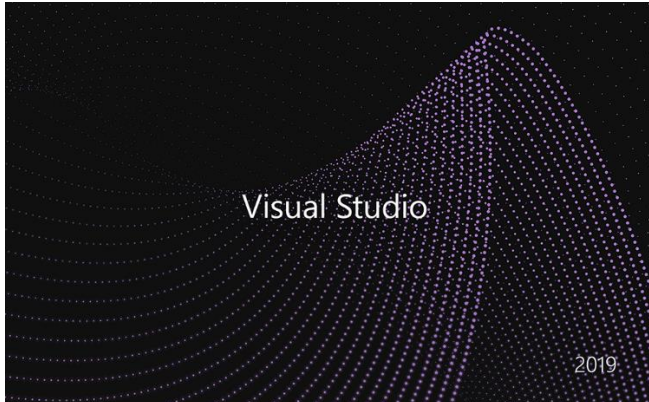# Chapter 2: Basic Operations

Open Workbook or (CSV, Json , XML) Files as workbook

Sample: HelloWorld console application

Follow coming steps to create HelloWorld Project

1- Open visual studio

2- Choose Create new project

3- Choose Console App (.NET framewrok)
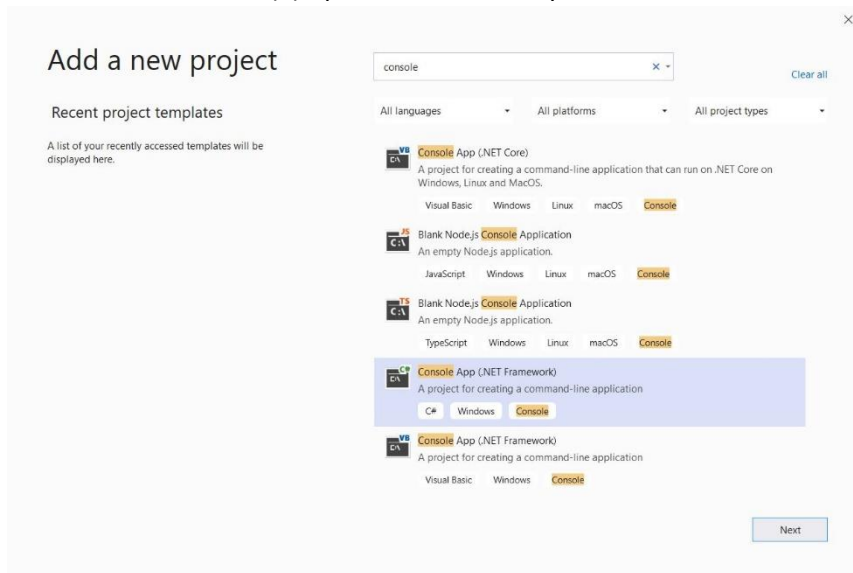
4- Give our sample name "HelloWorld" and click create
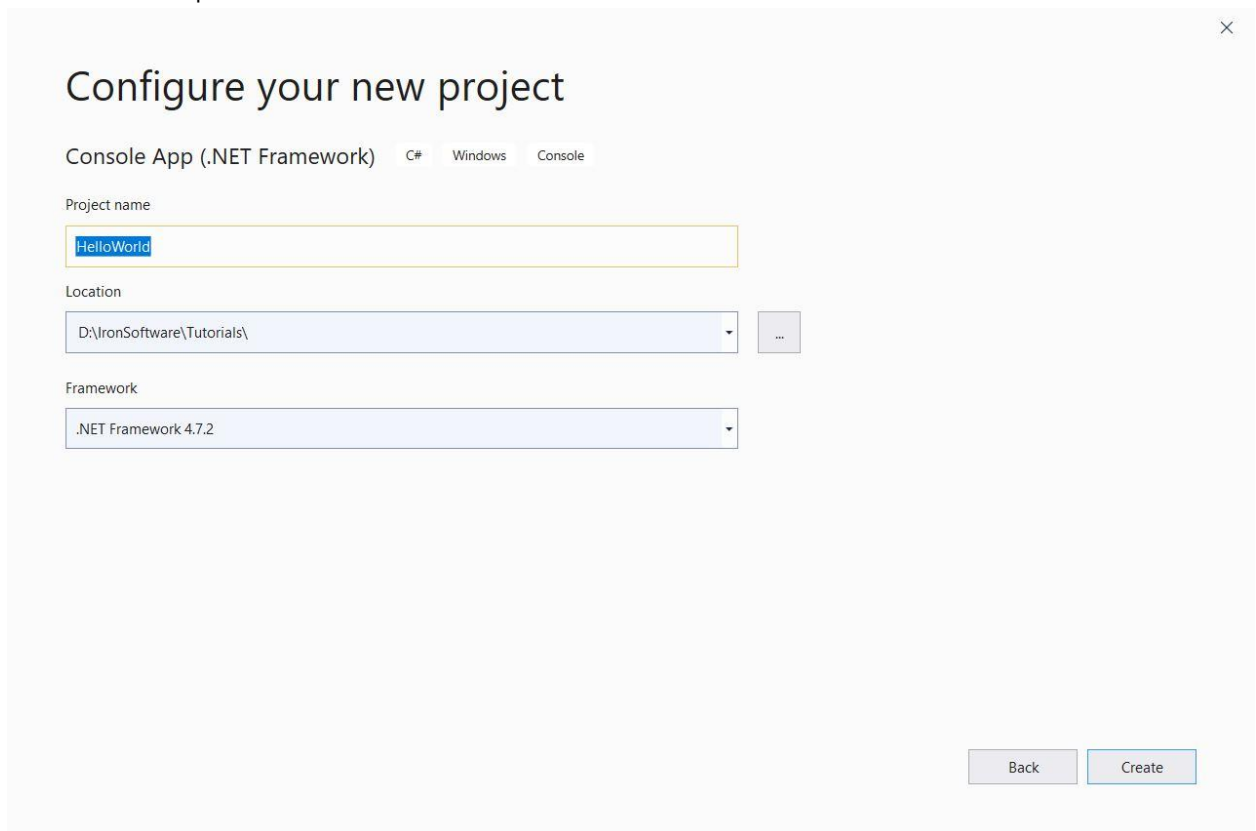
5- Now we have console application created

6- Add IronXL.Excel => click install

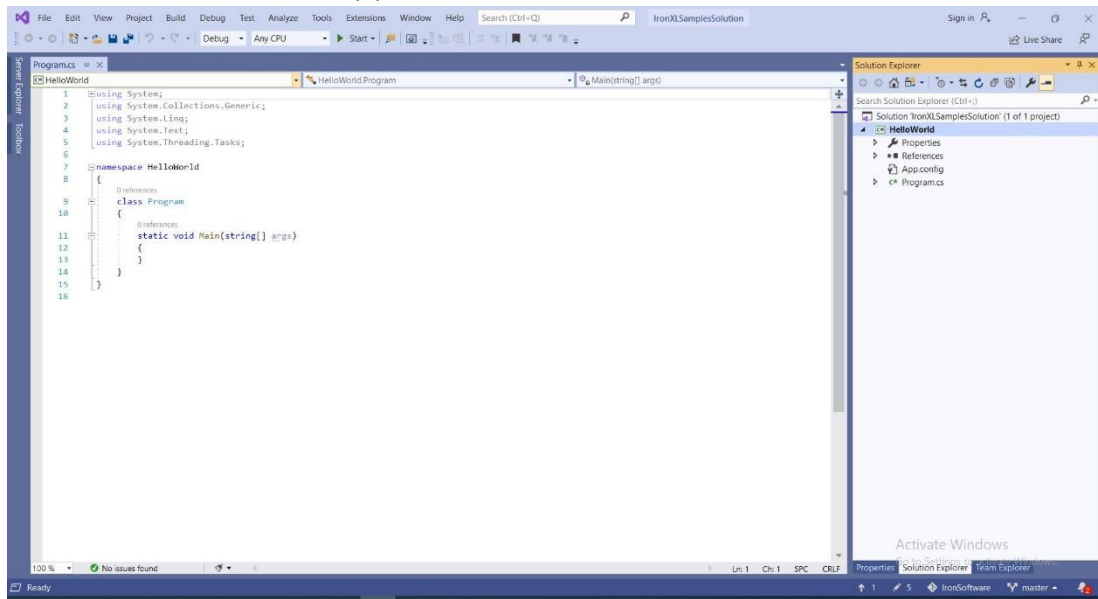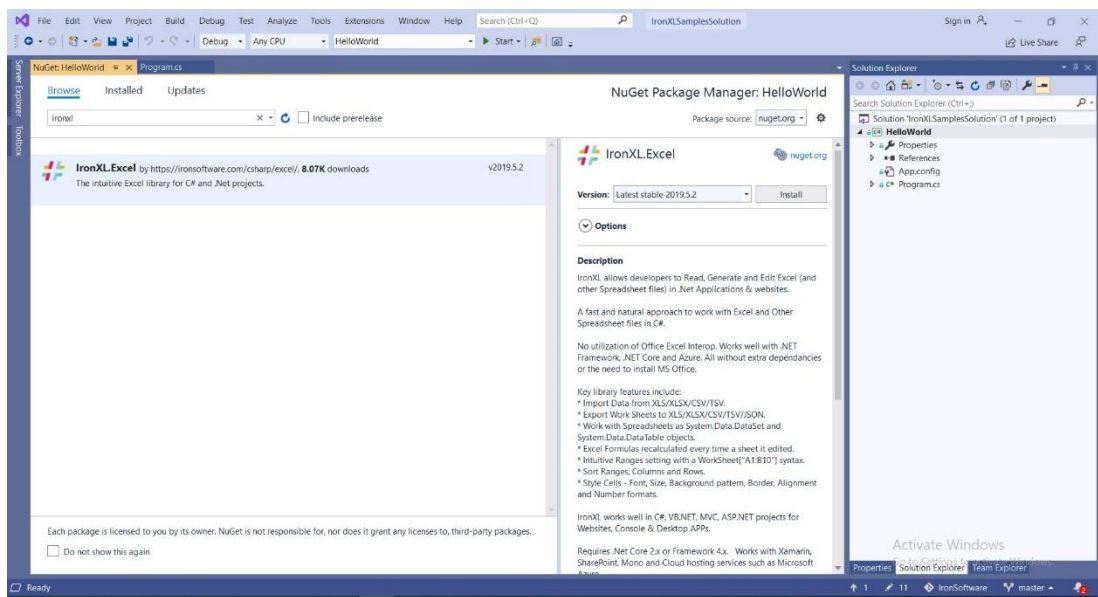7- Add our first few lines that reads 1ˢᵗ cell in 1ˢᵗ sheet in excel file and print it

```csharp
static void Main(string[] args)
{
 var workbook = IronXL.WorkBook.Load(@"D:\IronSoftware\Files\HelloWorld.xlsx");
 var sheet = workbook.WorkSheets.First();
 var cell = sheet["A1"].StringValue;
 Console.WriteLine(cell);
}
```

8- Also we can load CSV file , create new text file and add to its list of names and ages as follow then save it as CSVList.csv
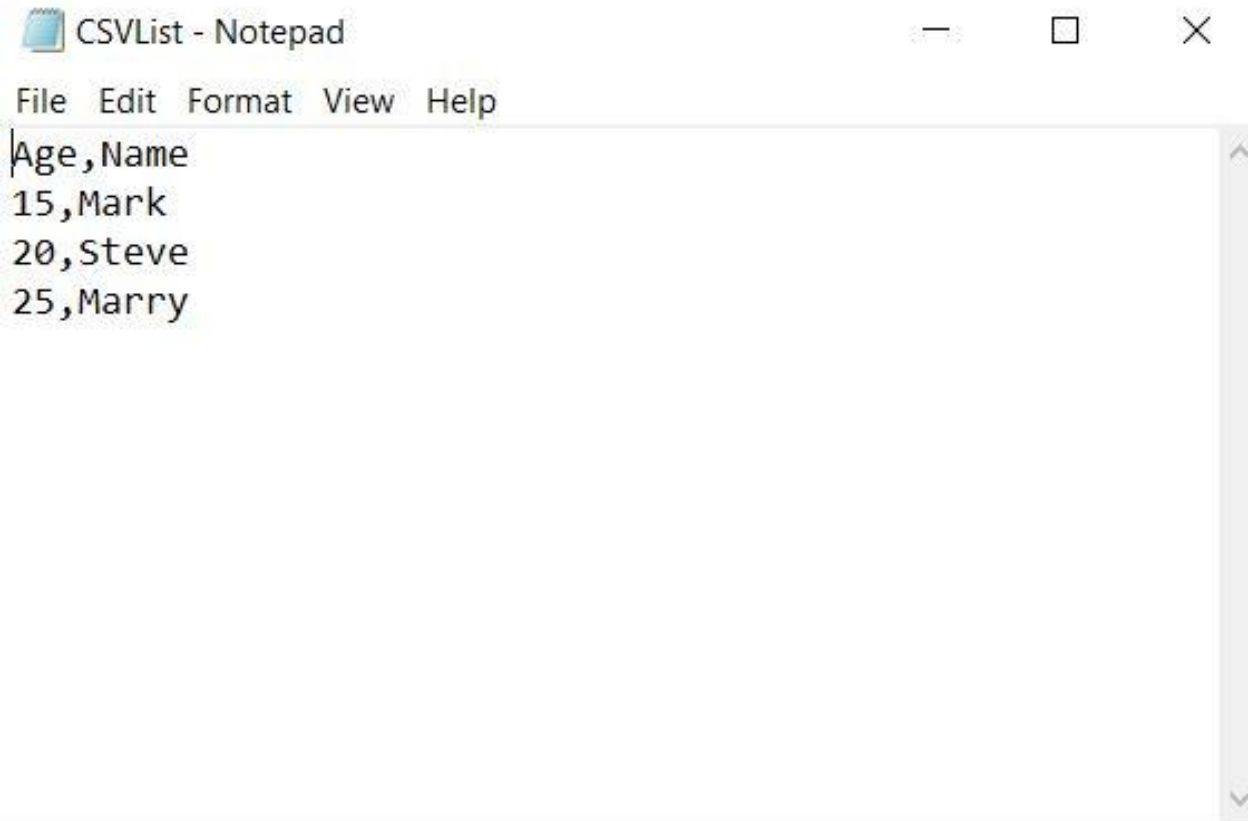
9- Code snippet should be like this

```
static void Main(string[] args)
{
 var workbook = IronXL.WorkBook.Load(@"D:\IronSoftware\Files\CSVList.csv");
 var sheet = workbook.WorkSheets.First();
 var cell = sheet["A1"].StringValue;
 Console.WriteLine(cell);
}
```

<-- I'll continue to talk about how to load Json, XML and also different options will loading -->

Chapter 3: Advanced Sheet Operations

<-- I'll talk about how to manipulate loop add header or footer into the sheet, and how to execute common functions like Sum, average, etc. -->

Summary

Author

Ahmed Aboelmagd is a Full-stack experienced and certified Microsoft technology specialist with 12+ year's experience in IT and Software development, delivered 15+ successful project in many size scales from small to an enterprise for industries like (tourism, educational, manufacturing, etc.)