**Hands-on .NetCore 3.0**

**Writing Files**

Table of Contents

[Scaffolding Console App Basics 2](#_Toc23627116)

[WritingFiles Console 2](#_Toc23627117)

[Creating the console app 2](#_Toc23627118)

[Simple Write 2](#_Toc23627119)

[Advance Write 3](#_Toc23627120)

[File object 4](#_Toc23627121)

[Using StreamWriter to write the file 4](#_Toc23627122)

# Scaffolding Console App Basics

We have already run the commands multiple times, here is the info

* dotnet new console
* dotnet restore [pulls in the dependencies needed by the application]
* dotnet run [compiles and run the application]
* dotnet build [compiles the application]
* dotnet publish [packages up the files for reuse]

Take a look at this document for more details:

<https://itplate.blogspot.com/2019/11/scaffolding-applications-with-net-cli.html>

# WritingFiles Console

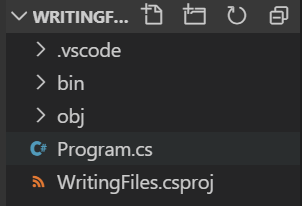
## Creating the console app

Run command **dotnet new console -o WritingFiles**

It has done the restore for us as well

Open the app with VS Code by

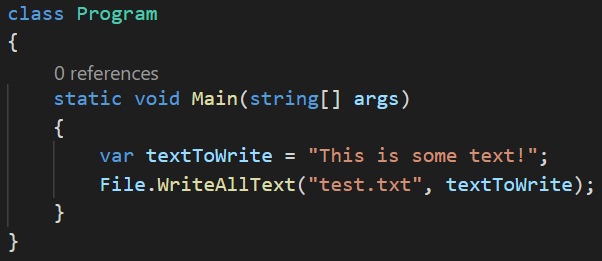
1. **cd WritingFiles**
2. and then typing **code .** [code space dot]



## Simple Write

Make sure to put a reference to **using System.IO;**

We’ll be writing a file in the same folder.



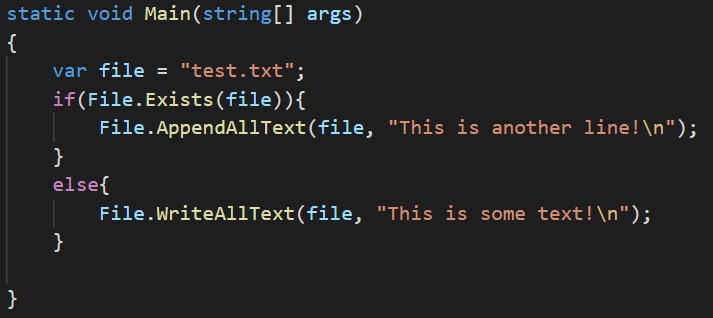
Then do dotnet run, after this has completed look in your working folder and open the file.

|  |  |
| --- | --- |
|  |  |

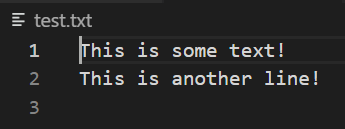
If you change the text and re run it, it will erase the content of your file.

## Advance Write

We’ll first check the file for existence and then will wither write or append to it. Make sure to add \n to your text so that each sting is added on to the new line.

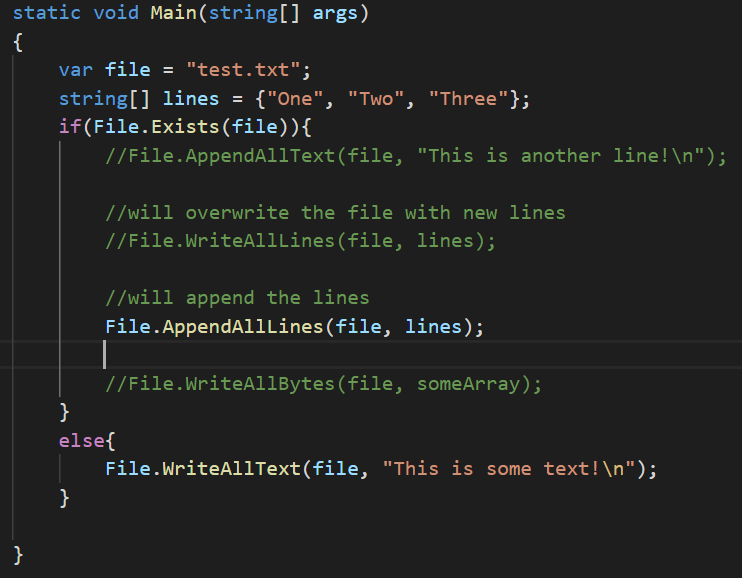


Then run the app using **dotnet run** and take a look at the file. You can view the file directly in VS Code as well.



## File object

We can use File object to write arrays, bytestream etc



## Using StreamWriter to write the file

This is a little home work for the group. There are a tons of resources available to online.