



# Arquivos do .NET

[.sln - Arquivo Solution](#)

[.csproj - Arquivo do C# Project](#)

[program.cs - Classe Principal](#)

[obj e bin - Subpastas de compilação](#)

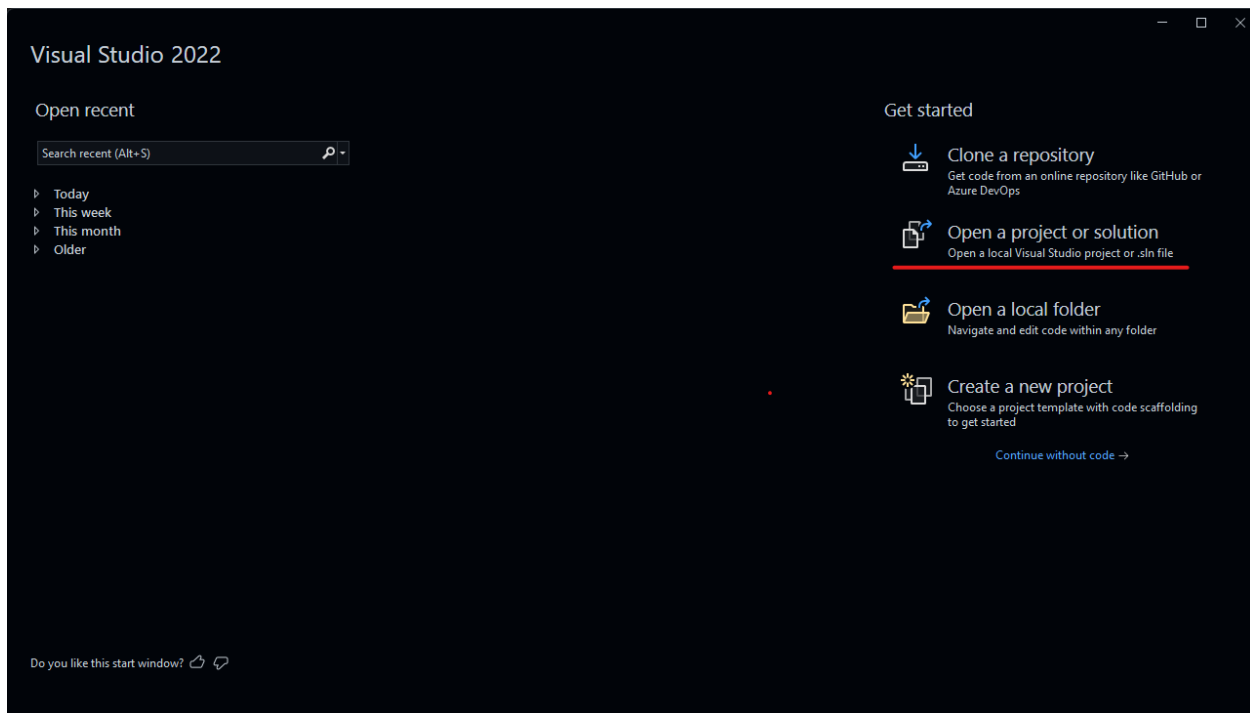
- Em um projeto .NET padrão existem vários arquivos padrões que não envolvem o projeto desenvolvido especificamente.

## .sln - Arquivo Solution

- Todo projeto C# é chamado de **Solução**, por isso ele possui um arquivo com a extensão `.sln` que significa **Solution**.

.git	30/03/2023 15:19	Pasta de arquivos	
.vs	23/03/2023 15:57	Pasta de arquivos	
Sample.Library	23/03/2023 14:57	Pasta de arquivos	
SampleAPI	30/03/2023 14:50	Pasta de arquivos	
SampleLibraryTests	23/03/2023 15:25	Pasta de arquivos	
TestResults	23/03/2023 15:03	Pasta de arquivos	
.gitattributes	23/03/2023 15:42	Git Attributes Source File	3 KB
.gitignore	23/03/2023 15:42	Git Ignore Source File	7 KB
LICENSE.txt	23/03/2023 15:42	Documento de Texto	2 KB
README.md	23/03/2023 15:42	Markdown Source File	1 KB
Sample.Library.sln	27/03/2023 14:12	Visual Studio Solution	3 KB

- É a partir desse arquivo que é aberto o projeto .NET no Visual Studio, como mostra abaixo:



## .csproj - Arquivo do C# Project

- Este arquivo é o arquivo de configuração do projeto .NET, é nele que colocamos as dependências e bibliotecas pego do **Nuget** para o nosso programa.

- Podemos acessar ele abrindo esse arquivo na IDE ou pelo nome do projeto dentro do visual studio.

[https://drive.google.com/file/d/1DgzKBk-14Fy7MwLaPI9v\\_zOSTATZcKcx/view?usp=sharing](https://drive.google.com/file/d/1DgzKBk-14Fy7MwLaPI9v_zOSTATZcKcx/view?usp=sharing)

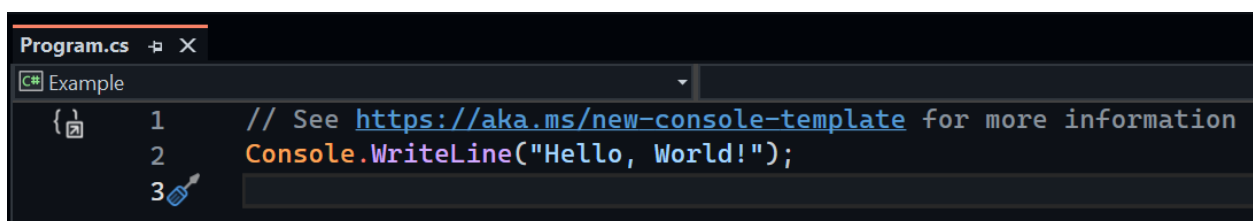
Acessando o .csproj pelo diretório do projeto no Visual Studio Code

[https://drive.google.com/file/d/1Zn6cw25jiP8XyoNZwO-G\\_tIRoxxXbguO/view?usp=sharing](https://drive.google.com/file/d/1Zn6cw25jiP8XyoNZwO-G_tIRoxxXbguO/view?usp=sharing)

Acessando o .csproj pelo Visual Studio, só clicando no nome do projeto.

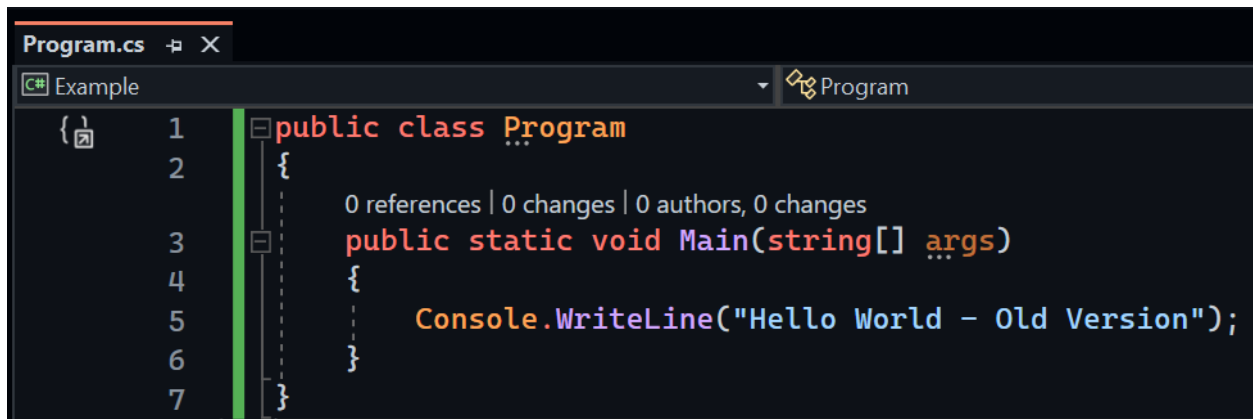
## program.cs - Classe Principal

- Este é o arquivo com a linguagem C# mesmo, como se fosse o App do Spring Boot, nele colocamos o código do C# que queremos rodar.
- Existem duas formas de colocar código nesse arquivo, um deles é o padrão novo que não tem dados de uma classe padrão e a outra é seguindo o padrão de Programação Orientado a Objetos.



```
Program.cs  [icon] X
C# Example
1 // See https://aka.ms/new-console-template for more information
2 Console.WriteLine("Hello, World!");
3 [icon]
```

Padrão novo



```
Program.cs [X]
C# Example [X] Program
1 public class Program
2 {
3     0 references | 0 changes | 0 authors, 0 changes
4     public static void Main(string[] args)
5     {
6         Console.WriteLine("Hello World - Old Version");
7     }
8 }
```

Padrão antigo, onde é mais didático

## obj e bin - Subpastas de compilação

Estas duas Subpastas são criados pelo Visual Studio para salvar as alterações e compilações feitas por ele, por isso não são importantes de mexermos nelas.