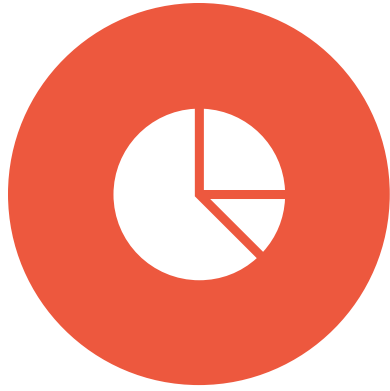




.NET 6 Where is the Main

ARCTECH INFO PRIVATE
LIMITED

Main function is missing in .NET 6



MAIN FUNCTION & ITS
USE



MAIN FUNCTION
BEFORE .NET 6



.NET 6
ENHANCEMENTS

MAIN FUNCTION & ITS USE

In most coding platforms and languages, the `main` function serves as the starting point for program execution.

It usually controls program execution by directing the calls to other functions in the program.

A program usually stops executing at the end of `main`.

Command Line arguments can be passed to a program via parameters to `main` functions

Syntax:

C/C++: `int main(int argc, char* argv[]) {}`

C# (upto .NET 5): `public void Main(string[] args);`

Main function before .NET 6

.NET FRAMEWORK, .NET CORE & .NET 5

```
using System;

namespace MainDemo
{
    0 references
    internal class Program
    {
        0 references
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World");
        }
    }
}
```

.NET 6

Just like in previous versions, .NET 6 by convention includes a Program.cs file.

However unlike older version which had a lot of boilerplate code, .NET 6 is neat and concise



```
Program.cs
1 Console.WriteLine("Hello, World!");
```

Top level statements

Usually, C# code can only be written inside methods within classes

However, with top level statements, introduced in C#9, you can begin writing simple code without all the boilerplate

A project can have only one file with top-level statements.

- Putting top-level statements in more than one file in a project results in the following compiler error:
- CS8802 Only one compilation unit can have top-level statements.
- Any other file in the project must include a class

Top-level statements can reference the `implicit args` parameter to access command-line arguments.