

Understanding Command Line Arguments in C#

Command line arguments in C# allow you to pass information to your program when it starts. This can be useful for configuring how your program runs without requiring user input during execution.

Step 1: Modify Your Program to Accept Arguments

1. Open Your Existing Project:

- Open the project you created in Visual Studio 2022. If you don't have a project yet, follow the previous steps to create one.

2. Modify the `Main` Method:

- The `Main` method already has a parameter `string[] args` which is used to accept command line arguments. You can access these arguments within your program.

```
using System;
```

```
namespace HelloWorld
```

```
{
```

```
    class Program
```

```
    {
```

```
        static void Main(string[] args)
```

```
        {
```

```
            if (args.Length > 0)
```

```
            {
```

```
                Console.WriteLine("Arguments passed to the program:");
```

```
                foreach (string arg in args)
```

```
                {
```

```
                    Console.WriteLine(arg);
```

```
                }
```

```
            }
```

```
        else
```

```
        {
```

```
            Console.WriteLine("No arguments were passed.");
```

```
        }
```

```
    }
```

```
}
```

```
}
```

Step 2: Run Your Program with Arguments

1. Using Visual Studio:

- Right-click on your project in the Solution Explorer and select 'Properties'.
- Go to the 'Debug' tab.
- In the 'Application arguments' box, enter the arguments you want to pass to your program, separated by spaces. For example: `arg1 arg2 arg3`.
- Click 'Save' and then run your program. You should see the arguments printed in the console.

2. Using Command Prompt or Terminal:

- Open Command Prompt or Terminal.
- Navigate to the directory where your compiled executable is located (usually in the `bin\Debug\net6.0` or similar folder).
- Run your program with arguments by typing the name of the executable followed by the arguments. For example:
``sh
HelloWorld.exe arg1 arg2 arg3
``
- Press Enter to execute the program. You should see the arguments printed in the console.

Step 3: Parsing and Using Command Line Arguments

1. Basic Parsing:

- You can parse and use the command line arguments as needed in your program. For example, you might want to check for specific flags or options:

using System;

```
namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            foreach (string arg in args)
            {
                switch (arg.ToLower())
                {
                    case "-h":
                    case "--help":
                        Console.WriteLine("Help: Use this program with the following options...");
                        break;
                    case "-v":
                    case "--version":
```

```

        Console.WriteLine("Version 1.0.0");
        break;
    default:
        Console.WriteLine($"Unknown argument: {arg}");
        break;
    }
}
}
}
}
}

```

Example: Command Line Arguments in a C# Program

```

using System;

namespace HelloWorld
{
    class Program
    {
        static void Main(string[] args)
        {
            if (args.Length > 0)
            {
                Console.WriteLine("Arguments passed to the program:");
                foreach (string arg in args)
                {
                    Console.WriteLine(arg);
                }

                // Parsing specific arguments
                foreach (string arg in args)
                {
                    switch (arg.ToLower())
                    {
                        case "-h":
                        case "--help":
                            Console.WriteLine("Help: Use this program with the following options...");
                            break;
                        case "-v":
                        case "--version":
                            Console.WriteLine("Version 1.0.0");
                            break;
                    }
                }
            }
        }
    }
}

```

```
        default:
            Console.WriteLine($"Unknown argument: {arg}");
            break;
        }
    }
}
else
{
    Console.WriteLine("No arguments were passed.");
}
}
}
```