Introduction to C# Programming with Visual Studio Code

This tutorial is designed for beginners who want to start learning C# using Visual Studio Code (VS Code), a lightweight and powerful code editor.

---

1. What is C#?

C# (pronounced "C-sharp") is a modern, object-oriented programming language developed by Microsoft. It is widely used for building desktop applications, web services, games (using Unity), and more.

---

2. Setting Up Your Environment

Step 1: Install .NET SDK

- Visit https://dotnet\.microsoft\.com/download
- Download and install the latest version of the .NET SDK for your operating system.

Step 2: Install Visual Studio Code

- Visit https://code\.visualstudio\.com
- Download and install VS Code for macOS.

Step 3: Install C# Extension

- Open VS Code.
- Go to the Extensions view (Ctrl+Shift+X or Cmd+Shift+X on macOS).
- Search for "C#" and install the extension provided by Microsoft.

---

3. Creating Your First C# Project

Step 1: Open Terminal in VS Code

- Use the integrated terminal (Ctrl+ or Cmd+ on macOS).

Step 2: Create a New Console App

```
dotnet new console -o HelloWorld
cd HelloWorld

```

This creates a new folder called HelloWorld with a basic C# console application.

Step 3: Open the Folder in VS Code

- Use File > Open Folder and select the HelloWorld folder.

---

4. Understanding the Code

Open Program.cs. You'll see something like:

```
using System;

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello, World!");
    }
}

```

- using System; imports the System namespace.
- Main is the entry point of the application.
- Console.WriteLine prints text to the console.

---

5. Running Your Application

In the terminal, run:

```
dotnet run

```

You should see:

```
Hello, World!

```

---

6. Next Steps

- Learn about variables, data types, and control structures.
- Explore object-oriented programming concepts.
- Build small projects like a calculator or a to-do list.

---

7. Working with Variables

Variables are used to store data in a program. In C#, you must declare a variable with a type before using it.

Example:

```
int age = 25;
string name = "Ali";
bool isStudent = true;

```

Common Data Types:

- int – whole numbers
- double – decimal numbers

- string – text
- bool – true/false

You can display variable values using Console.WriteLine:

```
Console.WriteLine("Name: " + name);
Console.WriteLine("Age: " + age);

```

8. Using Loops

Loops let you repeat actions multiple times. C# supports several types of loops.

For Loop:

```
for (int i = 0; i < 5; i++)
{
    Console.WriteLine("Iteration: " + i);
}

```

While Loop:

```
int count = 0;
while (count < 5)
{
    Console.WriteLine("Count: " + count);
    count++;
}

```

Foreach Loop (used with collections):

```
string[] fruits = { "Apple", "Banana", "Cherry" };
foreach (string fruit in fruits)
{
    Console.WriteLine(fruit);
}

```

9. Conditional Statements

Conditional statements allow your program to make decisions based on conditions.

If Statement:

```
int score = 85;
if (score >= 60)
{
    Console.WriteLine("You passed!");
}

```

If-Else Statement:

```
int score = 45;
if (score >= 60)
{
    Console.WriteLine("You passed!");
}
else
{
    Console.WriteLine("You failed.");
}

```

Else If Statement:

```
int score = 75;
if (score >= 90)
{
    Console.WriteLine("Excellent!");
}
else if (score >= 60)
{
    Console.WriteLine("Good job!");
}
 else
 {
     Console.WriteLine("Try again.");
 }

```

Resources

- https://learn.microsoft.com/en-us/dotnet/csharp/
- http://www.csharpcourse.com/

Happy coding! 🎉