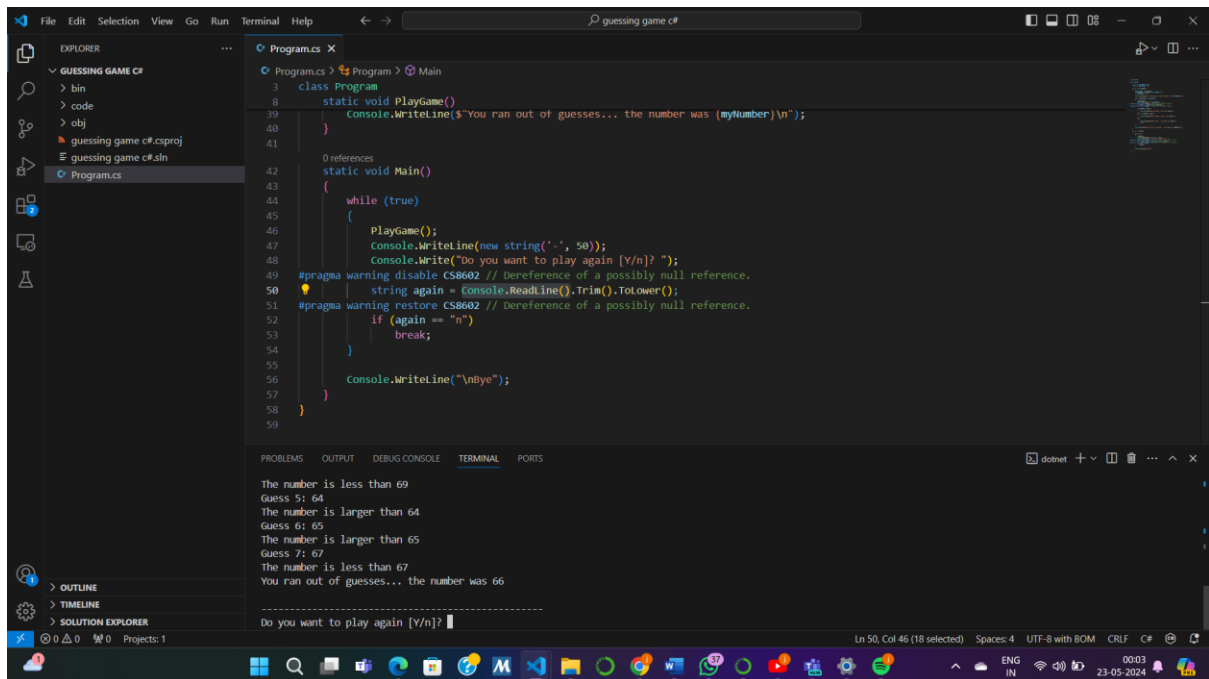


COMPARISONS BETWEEN CSHARP AND PYTHON

Let us look at the codes we provided

C#



The screenshot shows the Visual Studio IDE with a C# project named 'guessing game c#'. The Explorer pane on the left shows the project structure with files like 'bin', 'code', 'obj', 'guessing game c#.csproj', 'guessing game c#.sln', and 'Program.cs'. The main editor displays the code for 'Program.cs', which includes a 'Main' method and a 'PlayGame' static method. The code uses 'Console.WriteLine' for output and 'Console.ReadLine' for input. The terminal at the bottom shows the execution output, including the number of guesses and the final number.

```
using System;

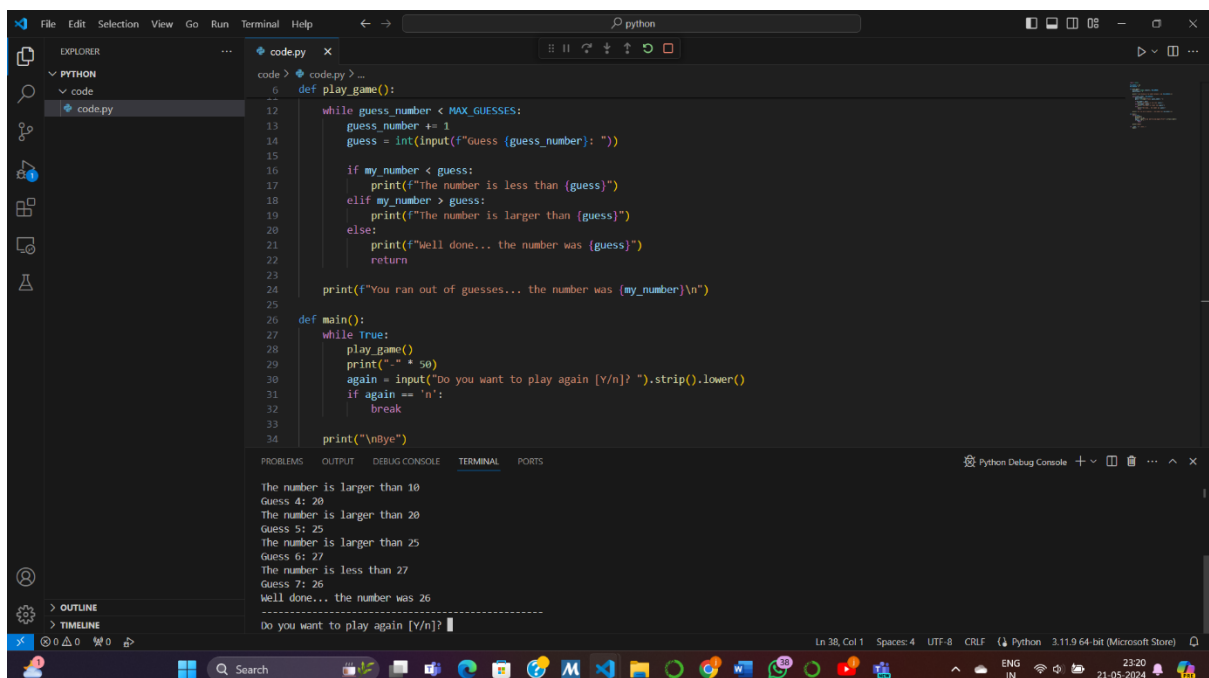
class Program
{
    static void PlayGame()
    {
        Console.WriteLine($"You ran out of guesses... the number was {myNumber}\n");
    }

    static void Main()
    {
        while (true)
        {
            PlayGame();
            Console.WriteLine(new string('-', 50));
            Console.Write("do you want to play again [Y/n]? ");
            string again = Console.ReadLine().Trim().ToLower();
            if (again == "n")
            {
                break;
            }
            Console.WriteLine("\nBye");
        }
    }
}
```

Terminal Output:

```
The number is less than 69
Guess 5: 64
The number is larger than 64
Guess 6: 65
The number is larger than 65
Guess 7: 67
The number is less than 67
You ran out of guesses... the number was 66
-----
do you want to play again [Y/n]?
```

Python



The screenshot shows the Visual Studio IDE with a Python project named 'python'. The Explorer pane on the left shows the project structure with files like 'code.py'. The main editor displays the code for 'code.py', which includes a 'play_game' function and a 'main' function. The code uses 'input' for user input and 'print' for output. The terminal at the bottom shows the execution output, including the number of guesses and the final number.

```
def play_game():
    while guess_number < MAX_GUESSES:
        guess_number += 1
        guess = int(input(f"Guess {guess_number}: "))
        if my_number < guess:
            print(f"The number is less than {guess}")
        elif my_number > guess:
            print(f"The number is larger than {guess}")
        else:
            print(f"Well done... the number was {guess}")
            return
    print(f"You ran out of guesses... the number was {my_number}\n")

def main():
    while True:
        play_game()
        print("-" * 50)
        again = input("do you want to play again [Y/n]? ").strip().lower()
        if again == 'n':
            break
    print("\nBye")
```

Terminal Output:

```
The number is larger than 30
Guess 4: 20
The number is larger than 20
Guess 5: 25
The number is larger than 25
Guess 6: 27
The number is less than 27
Guess 7: 26
Well done... the number was 26
-----
do you want to play again [Y/n]?
```

WE CAN COMPARE THESE CODES ON THE BASIS OF A FEW PARAMETERS :

Imports and Constants

Python: Uses import random for random number generation.

C#: Uses using System; for basic functionality and Random class for random number generation.

Function Definitions

Python: Functions are defined using def keyword. Example: def play_game():.

C#: Methods are defined using static void. Example: static void PlayGame().

Random Number Generation

Python: random.randint(1, MAX_NUMBER).

C#: rand.Next(1, MAX_NUMBER + 1).

Input and Output

Python: Uses input() for reading user input and print() for output.

C#: Uses Console.ReadLine() for input and Console.WriteLine() for output.

String Formatting

Python: Uses f-strings for formatting. Example: print(f"I am thinking of a number between 1 and {MAX_NUMBER}\n").

C#: Uses \$ before the string for interpolation. Example: Console.WriteLine(\$"I am thinking of a number between 1 and {MAX_NUMBER}\n");.

Conditional Statements

Python: Uses if, elif, and else for conditionals.

C#: Uses if, else if, and else.

Looping

Python: Uses while loop. Example: while guess_number < MAX_GUESSES:.

C#: Uses while loop in the same way. Example: while (guessNumber < MAX_GUESSES).

Main Function

Python: Checks if __name__ == "__main__": to run the main function.

C#: Uses static void Main() as the entry point.

Running the application

We generally use 'dotnet run' to execute the csharp code. While we can use f5 shortcut to run the python code.

Summary

Both the Python and C# versions of the game are structurally similar but differ in syntax and some built-in functions. Python's code is generally more concise and easier to read due to its high-level nature, while C# is more verbose and requires explicit type declarations and method structures.

