

COMPARISONS BETWEEN CSHARP AND PYTHON

Let us look at the codes we provided

C#

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project named "GUESSING GAME CS" containing "bin", "code", "obj", "guessing game.csproj", and "guessing game.cs.sln".
- Code Editor:** Displays "Program.cs" with the following C# code:

```
3 class Program
4 {
5     static void PlayGame()
6     {
7         Console.WriteLine($"You ran out of guesses... the number was {myNumber}\n");
8     }
9     static void Main()
10    {
11        while (true)
12        {
13            PlayGame();
14            Console.WriteLine(new string('-', 50));
15            Console.Write("Do you want to play again [Y/n]? ");
16            #pragma warning disable CS8602 // Dereference of a possibly null reference.
17            string again = Console.ReadLine().Trim().ToLower();
18            #pragma warning restore CS8602 // Dereference of a possibly null reference.
19            if (again == "n")
20            {
21                break;
22            }
23            Console.WriteLine("\nBye");
24        }
25    }
26 }
```
- Terminal:** Shows the output of the program:

```
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
```
- Status Bar:** Shows "Ln 50, Col 46 (18 selected) Spaces: 4 UTF-8 with BOM CRLF C# dotnet" and the date "23-05-2024".

Python

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project named "PYTHON" containing "code" and "code.py".
- Code Editor:** Displays "code.py" with the following Python code:

```
6 def play_game():
7     while guess_number < MAX_GUESSES:
8         guess_number += 1
9         guess = int(input(f"Guess {guess_number}: "))
10        if my_number < guess:
11            print(f"The number is less than {guess}")
12        elif my_number > guess:
13            print(f"The number is larger than {guess}")
14        else:
15            print(f"Well done... the number was {guess}")
16            return
17
18        print(f"You ran out of guesses... the number was {my_number}\n")
19
20 def main():
21     while True:
22         play_game()
23         print("-" * 50)
24         again = input("Do you want to play again [Y/n]? ").strip().lower()
25         if again == 'n':
26             break
27
28         print("\nBye")
```
- Terminal:** Shows the output of the program:

```
The number is larger than 10
Guess 4: 29
The number is larger than 29
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
```
- Status Bar:** Shows "Ln 38, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)" and the date "21-05-2024".

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.

