

**Unit Code: PRT582**

**Software Unit Testing Report**

**“Guess It Win It” game using TDD**

Submitted by,

Sharon Cheethayil Sasidharan

Student ID: s366502

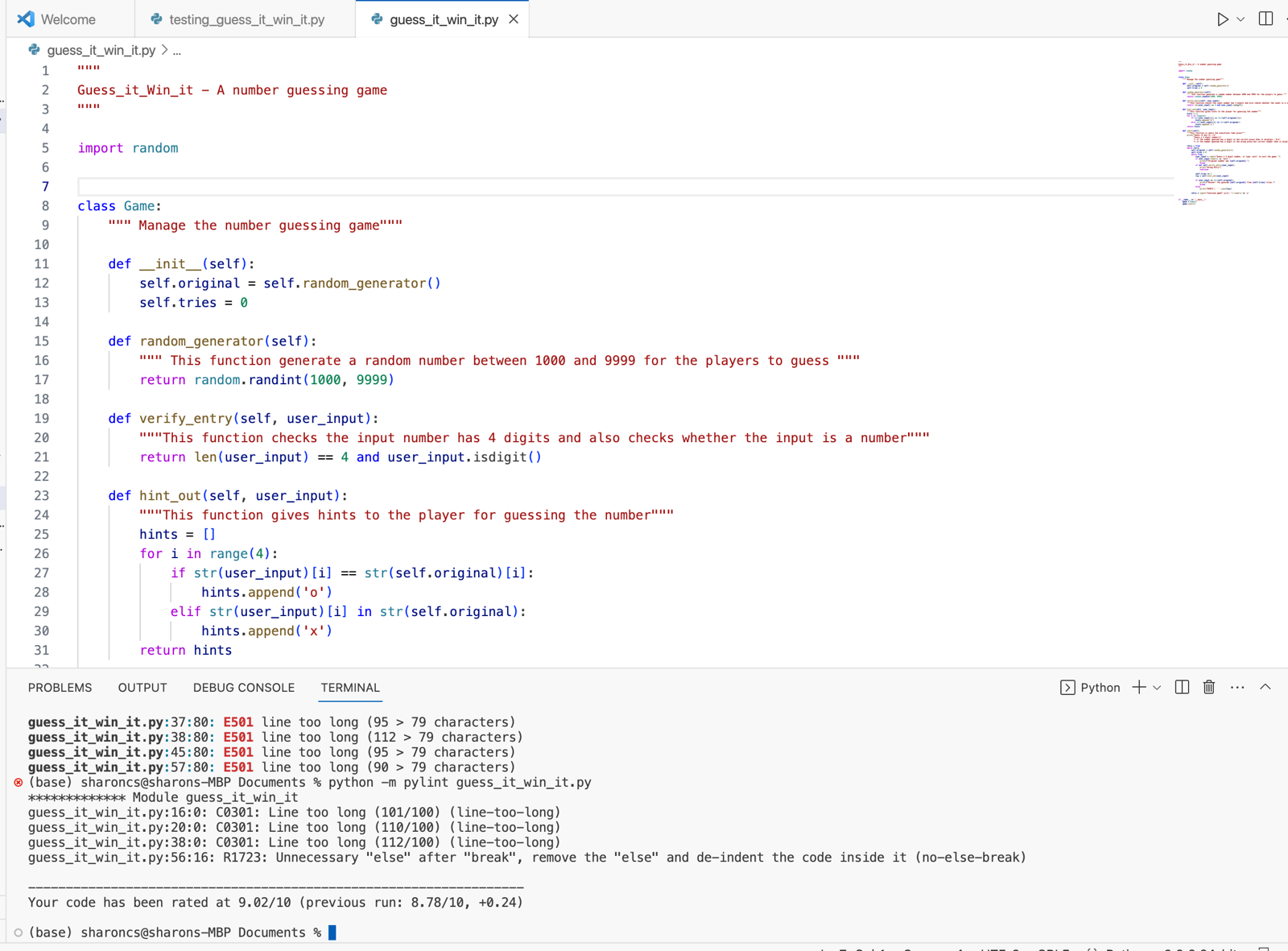
Introduction

The main goal of this project is to create a number guessing game, "Guess it Win it." Using an automated unit testing framework provided by the 'unit test' library, the primary focus is on assuring the accuracy and dependability of the game. In this Project, I used automated tools like Flake8 and Pylint to ensure uniform code quality and readability and I followed Test Driven Development (TDD) for making code that is I write test before the original code so that it describe how the code works.

Project requirements and objectives

A clear set of goals and specifications are what are driving this project:

* Create a guessing game with a dynamic 4-digit number that is generated at random between 1000 and 9999. Player can predict the number in several tries and is given an option to quit after guessing.
* Implement a hint system that provides players with feedback after each guess. The system should show accurately located digit as ('o') and correctly predicted but incorrectly placed as ('x').
* Create a framework for automated unit testing to check the accuracy and functioning of full requirements and logic
* Use Flake8 with Pylint to check that code complies with coding standards, fostering readable and consistent programming.



1. This function generates a number between 1000 and 9999 so that the player can guess the number. The Game class, a coordinator of the game's complexities, is the keystone of this file. This class includes a number of crucial steps that together make up the gameplay.The game begins with the call to \_\_init\_\_(self). It starts the game by creating a mysterious four- and initialising the attempt counter.The verify\_entry function makes sure that the input is made up of 4 digits and is numeric.

A screenshot of a computer code

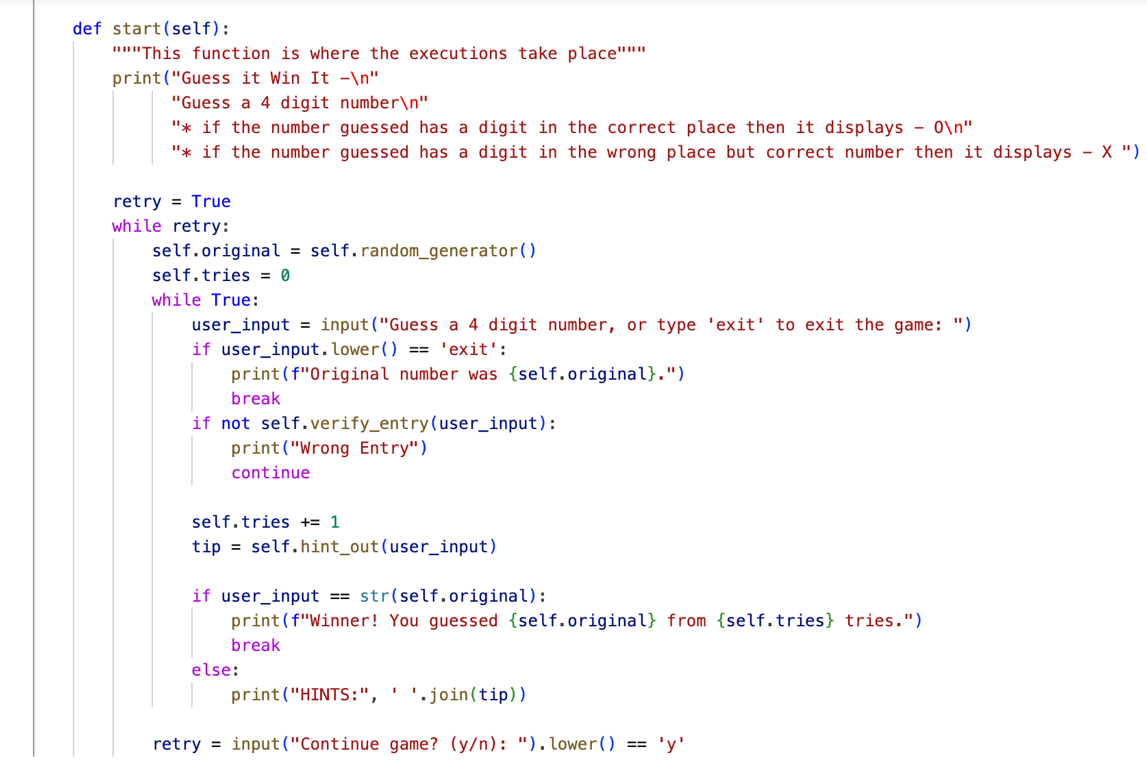
Description automatically generated

1. The core of the gameplay's strategic component is the hint\_out function, which can be accessed via the user\_input. The function gives hints based on player input and leads players to the solution by displaying a 'o' for correctly positioned digits and a 'x' for correctly guessed but misplaced numbers.

A screen shot of a computer code

Description automatically generated

1. The flow of the game is determined by the start() . This is the function that determines and validates the user entry. This component is crucial because in this component we check the users desire to continue the game or quit the game and also wrong entries are determined.



A screenshot of a computer

Description automatically generated

I created 4 test cases that check the random number creation, datatypes of variables, the hint generation function and a function to validate and verify the user inputs

test\_hint\_out\_testcase: The objective of this test is to validate that the hint\_out function generates hints that only contain the characters 'o' and 'x' in accordance with the game's rules.

test\_random\_generation\_testcase: Carefully examines the random\_generator function's consistency in generating valid 4-digit values within the prescribed range.

test\_datatype\_testcase: The focus of this test is on how well-coordinated the data structures are inside the Game class. A functioning system needs data type validation to work in a good way.

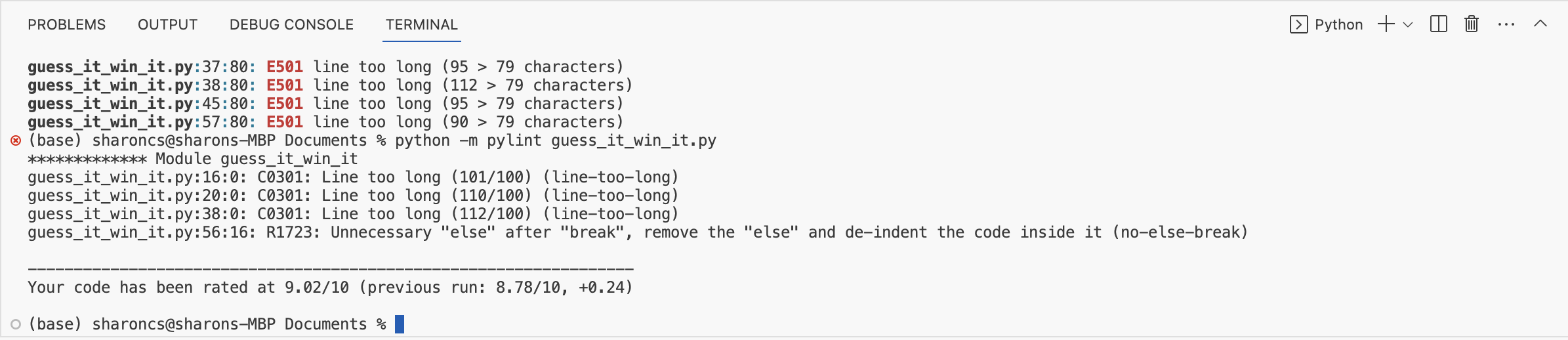
test\_input\_testCase: This test carefully examines the verify\_entry function, evaluating its capability to precisely assess both valid and incorrect user inputs.

A screenshot of a computer code

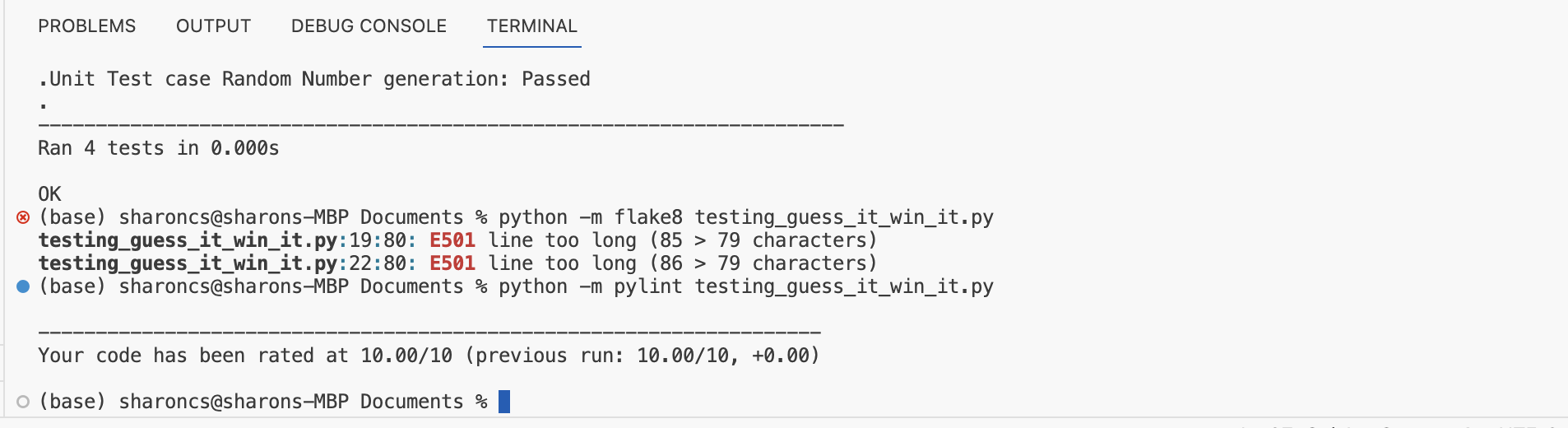
Description automatically generated

QUALITY CHECK:

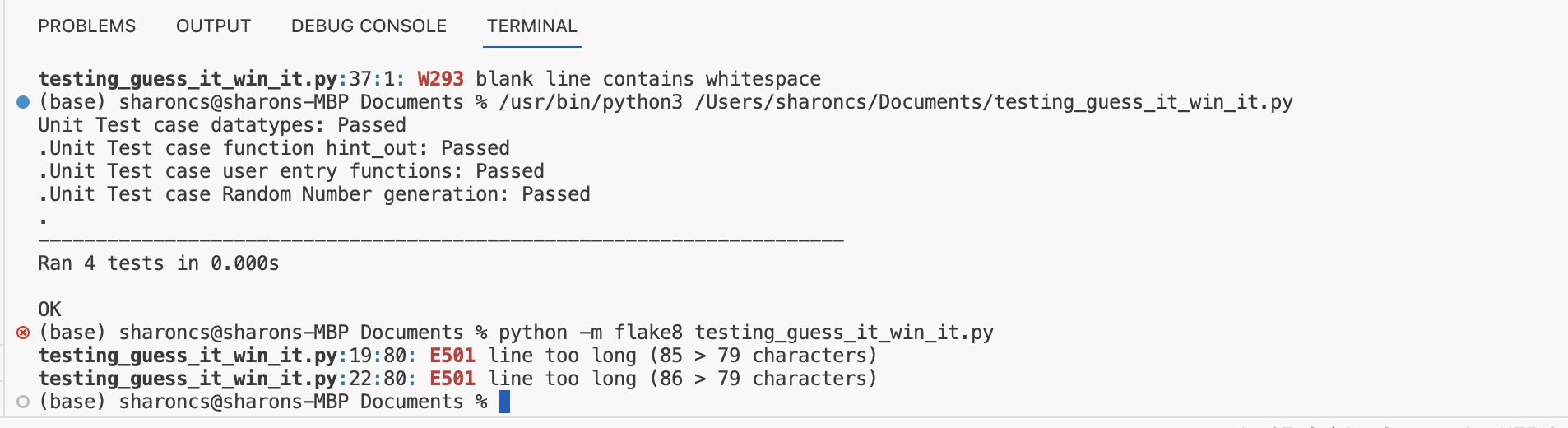
Flake8 and Pylint, two automated code quality evaluation tools, are used to examine and validate the code.



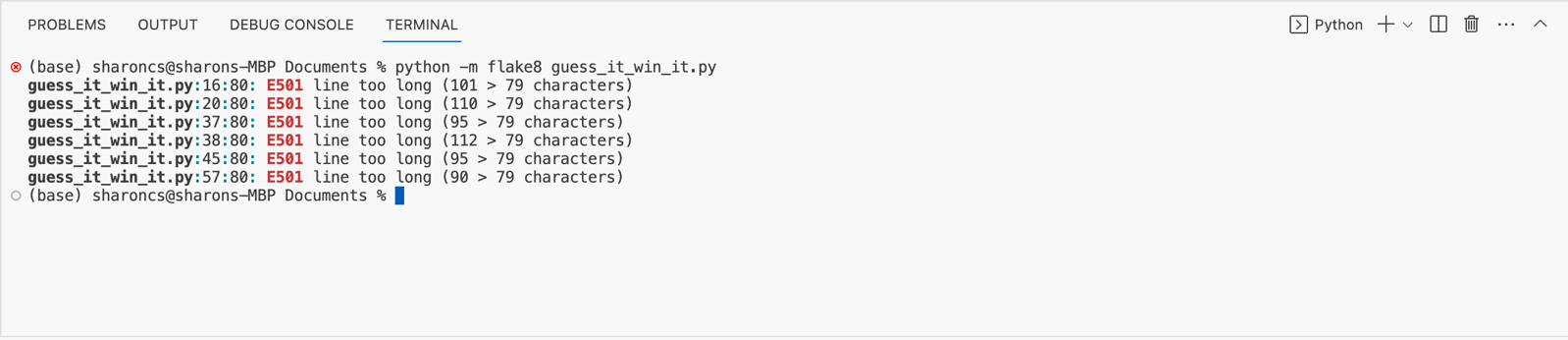
Got 9.02/10 rating in pylint for the game code



Got 10/10 rating in pylint for testing\_guess\_it\_win\_it.py



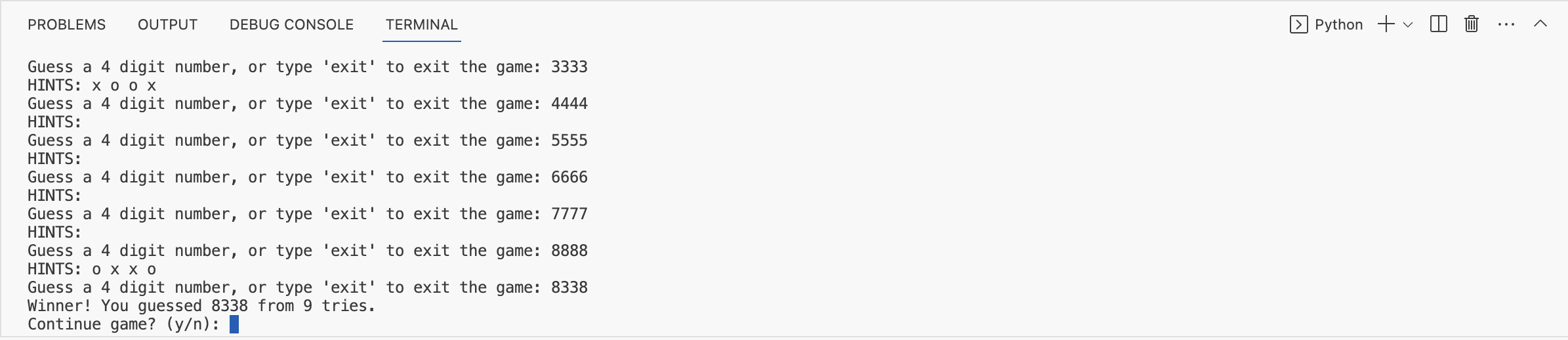
Flake8 output for testing\_guess\_it\_win\_it.py

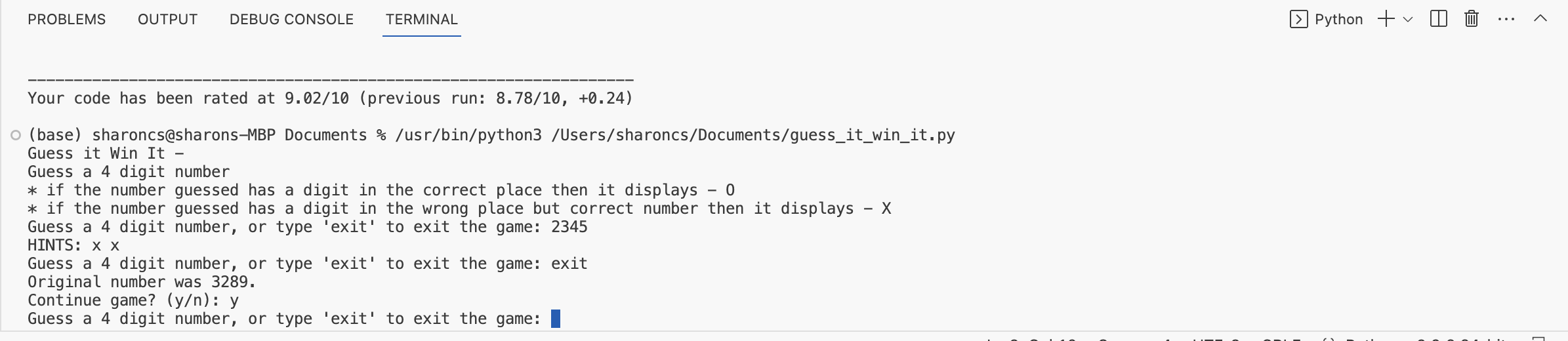


Flake8 output for guess\_it\_win\_it.py

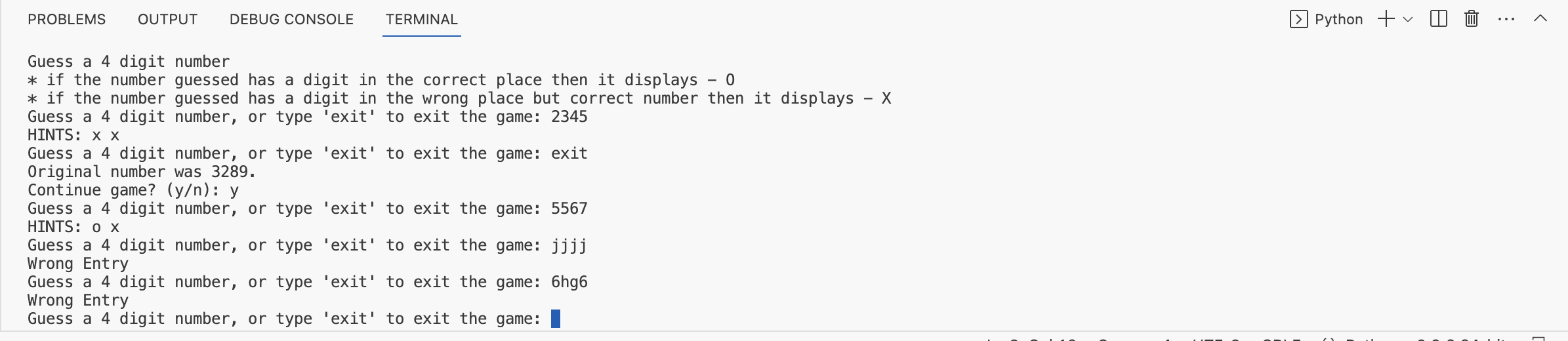
Output for different cases

1.



2.

3.



Conclusion

In conclusion, the creation of the "Guess\_it\_Win\_it" game combines creative functions with systematic testing procedures. The attraction of the game is its ability to entertain players while also exercising players' thinking skills through logic. Here I used the TDD method for development so that I planned before the creation. With the careful application of Flake8 and pylint, code quality is maintained in the background.Overall by using these best practises, the project achieves an ideal balance between creativity, reliability, and sustainability.

GitHub Repository Link: