PPFI PROJECT ASSIGNMENT

01

Crack the Code: Building Bagels Game in Python

Project Title:

Crack the Code: Building Bagels Game in Python

Objective:

The objective of this assignment is to create Bagels game using Python. In Bagels, a deductive logic game, you must guess a secret three-digit number based on clues. The game offers one of the following hints in response to your guess: “Pico” when your guess has a correct digit in the wrong place, “Fermi” when your guess has a correct digit in the correct place, and “Bagels” if your guess has no correct digits. You have 10 tries to guess the secret number.

The Program in Action:

When you run ***bagels.py***, the output will look like this:

```
Bagels, a deductive logic game.
By Student_Name (Student_Registration_No.)
I am thinking of a 3-digit number. Try to guess what it is.
Here are some clues:
When I say:    That means:
  Pico         One digit is correct but in the wrong position.
  Fermi        One digit is correct and in the right position.
  Bagels       No digit is correct.
I have thought up a number.
 You have 10 guesses to get it.
Guess #1:
> 123
Pico
Guess #2:
> 456
Bagels
Guess #3:
> 178
Pico Pico
--snip--
Guess #7:
> 791
Fermi Fermi
Guess #8:
> 701
You got it!
Do you want to play again? (yes or no)
> no
Thanks for playing!
```

Hints to Prepare the Project:

- Create a function named **generate_secret_number** function to generate a random 3-digit secret number using random.sample.
- Create a function named **provide_hint** function to compare the player's guess with the secret number and provide the appropriate hint.
- Create a **play_game** function which:
    - ✓ Manages the main game loop, allowing the player 10 attempts to guess the secret number.
    - ✓ Calls **generate_secret_number** to get a new secret number for each game.
    - ✓ Validates user input to ensure it is a 3-digit number.
    - ✓ Calls **provide_hint** to give feedback on each guess.
    - ✓ Ends the game if the player guesses the correct number or runs out of attempts.
- Develop the **Main Code**, which executes the game loop and asks the player if they want to play again after each game.

Requirements:

- Implement the Bagels game following the provided instructions.
- Ensure the game provides the appropriate hints ("Pico," "Fermi," or "Bagels") for each guess.
- Allow the player 10 guesses to identify the secret number correctly.
- Implement a replay option, allowing the player to choose whether to play again.

Instructions:

- Create a Python script named bagels.py.
- Implement the Bagels game logic as described.
- Ensure the program provides the appropriate output for each guess and the final result.
- Implement a replay option, asking the player if they want to play again.

Source Code (paste the code below)

Project Report Preparation: (Total = 15 points)

Understanding Bagels Game (1 points):

- Explain the objective of the Bagels game.
- Describe the possible hints the game can provide ("Pico," "Fermi," or "Bagels").
- Why does the program use string values instead of integer values for comparisons?

Game Implementation (5 points):

- Provide a high-level overview of how the Bagels game is implemented in the bagels.py script.
- Explain the role of string values in representing the secret number.
- How is the game ensuring that the player has 10 guesses?

Guessing Mechanism (3 points):

- Describe how the game provides hints for each guess.
- Explain the difference between "Pico," "Fermi," and "Bagels" hints.
- How does the program handle cases where the player's guess has no correct digits?

Replay Option (1 points):

- Explain how the replay option is implemented in the game.
- Describe the user prompt for replay, and how the program responds based on the user's input.

Coding Style and Structure (1 points):

- Discuss the coding style and structure used in the bagels.py script.
- Are there any specific reasons for using string values for numeric digits?

Edge Cases and Testing (3 points):

- Identify potential edge cases in the Bagels game.
- Propose a set of test cases to ensure the correctness of the game implementation.
- Discuss how the game handles unexpected inputs.

Conclusion (1 points):

- Summarize the key components and features of the Bagels game.
- Reflect on the importance of deductive logic in the context of this game.

(Signature of the Faculty)	**(Signature of the Student)**
	Name:
Date:	**Registration No.:**
	Branch:
	Section