

**Department of Software Engineering**

**Faculty of Computer Science & Information Technology**

**The Superior University, Lahore**

**Name:** MUHAMMAD HAMZA ALI

**Roll No:** SU92-BSAIM-S24-032

**Section:** 3A

**Subject:** ARTIFICIAL INTELLIGENCE(LAB)

**Task No:** Mini Project 1(Lab Task-2)

**Mini Project-1**

**Fizz Buzz Game**

**1. Introduction:**

The Fizz Buzz Game is a simple interactive command-line game designed to engage users in a fun and educational way.

The game challenges players to respond correctly to a series of numbers based on specific divisibility rules.

**2. Features:**

The application provides the following features:

* **Random Number Generation:** The game generates random numbers between 1 and 50.
* **User Input:** Players can input their answers based on the generated number.
* **Fizz Buzz Logic:** The game implements the classic Fizz Buzz rules:
* If the number is divisible by 3, the answer is “Fizz”.
* If the number is divisible by 5, the answer is “Buzz”.
* If the number is divisible by both 3 and 5, the answer is “Fizz Buzz”.
* If the number is not divisible by either, the player should respond with the number itself.
* **Exit Option:** Players can exit the game by typing “exit”.

**3. Implementation Details:**

The application is implemented in Python using a class-based structure.

* **Class FizzBuzz:**
* **Methods:**
* **‘\_\_init\_\_()’:**  Initializes the game and welcomes the player.
* **‘get\_correct\_answer(number)’:** Determines the correct answer based on the Fizz Buzz rules.
* **‘play()’:** Contains the main game loop, generating numbers and processing user input.

**4. User Interaction:**

* The game runs in a continuous loop, generating a new random number and prompting the user for their answer.
* The user can type their answer or exit the game at any time.
* Feedback is provided after each answer, indicating whether the user’s response was correct or incorrect.

**5. Code Structure:**

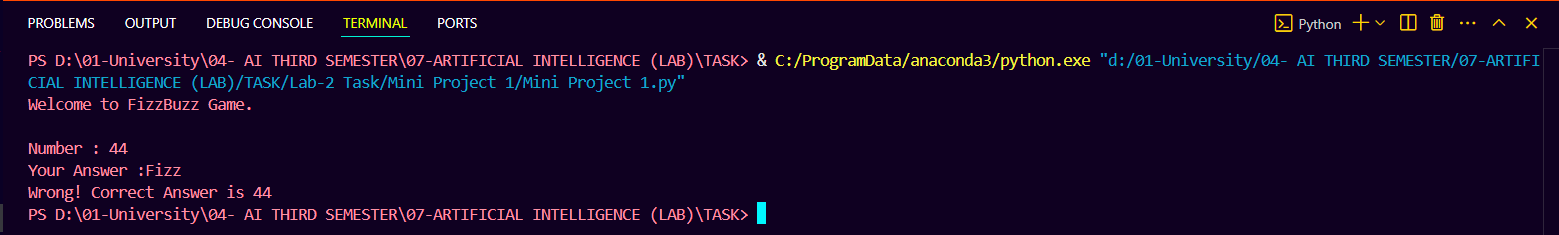
The main components of the code are:

* The main game logic is encapsulated within the **‘play()’** method, which handles the flow of the game.
* The **‘get\_correct\_answer()’** method encapsulates the Fizz Buzz logic, making the code modular and easy to maintain.

**6. Example Usage:**

* When the game starts, a random is displayed (E.g. “Number: 15”).
* The user is prompted to enter their answer (E.g. “Your Answer: Fizz Buzz”).
* The game checks the answer and provides feedback (E.g. “Correct! Next Number.”).

**7. Output:**



**8. Conclusion:**

* The Fizz Buzz game is an engaging way to practice basic arithmetic and divisibility rules.
* It serves as a fun exercise for users of all ages and can be used in educational settings.