# fizzbuzz Full Source Code

import random  
def buzfuz():  
 number = random.randint(1,100)  
 while True:  
 print(number)  
 print("tell me its fizz if divide by 3 or buzz if divide by 5 or if it divide by 3 and 5 then write fizzbuzz either none " )  
   
 user = input("buzz or fizz or fizzbuzz or none")  
   
 if number % 5 == 0 and number % 3 == 0:  
 correct = "fizzbuzz"  
 elif number % 3 == 0 :  
 correct = "fizz"  
 elif number % 5 == 0:  
 correct = "buzz"  
 else:  
 correct = "none"   
   
 if user == correct:  
 print("Correct! It is", correct)  
 else:  
 print("Wrong! Correct answer was:", correct)  
 break  
   
   
 new\_number = random.randint(1,100)  
 print(new\_number)  
 total = number + new\_number  
 print(f"old number {number} + new number {new\_number} = {total}")  
 number = total  
   
   
buzfuz()

# Step-by-step Explanation

1. Import the `random` module so the program can generate random numbers.

2. Define the function `buzfuz()` which contains the game logic.

3. Generate a random integer between 1 and 100 and store it in `number`.

4. Enter an infinite loop (`while True`) so the game continues until the user makes a mistake.

5. Print the current `number` and a message telling the user how to respond (fizz/buzz/fizzbuzz/none).

6. Read the user's answer using `input()` and store it in `user`.

7. Determine the correct label (`correct`) for `number` by checking divisibility:  
 - If divisible by both 3 and 5 -> `fizzbuzz`.  
 - Else if divisible by 3 -> `fizz`.  
 -Else if divisible by 5 -> `buzz`.  
 - Otherwise -> `none`.

8. Compare the user's answer to `correct`:  
 - If they match, print a confirmation and continue the game.  
 - If they don't match, print the correct answer and `break` out of the loop (ending the game).

9. If the user was correct, generate a new random integer between 1 and 100 (`new\_number`).

10. Print `new\_number`, compute the sum `total = number + new\_number`, show the addition to the user, and set `number = total` so the next round uses the summed value.

11. When the loop ends (user answered incorrectly), the function returns and the program finishes.

12. Finally, call `buzfuz()` to start the game.