# Buzz Game Code Explanation

This document explains the Buzz Game code step by step in simple language. The code is written in Python and allows a user to play a dynamic version of the BuzzFizz game.

## 1. Game Rules

The game follows these rules:  
- If a number is divisible by 3, the answer should be 'Buzz'.  
- If a number is divisible by 5, the answer should be 'Fizz'.  
- If a number is divisible by both 3 and 5, the answer should be 'BuzzFizz'.  
- The user can also add one more custom rule, e.g., 7 = Pop.

## 2. Extra Rule Input

At the start, the program asks the user to enter an extra divisor (like 7). If the user enters 0, no extra rule is added. If the user enters a number (e.g., 10), the program also asks for a word (e.g., 'Python'). So whenever a number is divisible by 10, the program will add 'Python' to the output.

## 3. Main Game Loop

The game starts from number 1 and continues until the user makes a mistake or quits:  
1. For each number n, the program checks divisibility.  
2. It builds a string (val) according to the rules.  
3. If no rules apply, val is just the number itself.  
4. The program then shows the number and asks the user for an answer.

## 4. User Input and Checking

- If the user types 'q', the game ends immediately.  
- If the user types the correct value (matches val), the game continues to the next number.  
- If the user types the wrong value, the program prints the correct answer and ends the game.

## 5. Example Walkthrough

Suppose the user adds the rule: 2 = Hi.  
  
Round 1: n = 1 → not divisible by 2, 3, or 5 → val = '1'.  
Round 2: n = 2 → divisible by 2 → val = 'Hi'.  
Round 3: n = 3 → divisible by 3 → val = 'Buzz'.  
Round 4: n = 5 → divisible by 5 → val = 'Fizz'.  
Round 6: n = 6 → divisible by 2 and 3 → val = 'BuzzHi'.  
If the user gives a wrong answer at any step, the game ends.

## 6. Why the Code Works

The code uses simple concepts:  
- Loops (while True) to keep the game running.  
- If conditions to check divisibility.  
- String concatenation to build the correct answer.  
- Input/output functions to interact with the user.  
  
This makes the game dynamic, because the user can change the extra rule each time they play.

## ScreenShort(output)

