Workshop on Python (Day 7)

By Suriya G
Organized by Suresh Sir, UPNM

TABLE OF CONTENTS

🎮 1. Guess the Number Game

🎯 Concepts Used:

- if, elif, else
- while loop
- input(), int(), random module

💡 Game Idea:

Computer randomly picks a number (1 to 100). Player has to guess it. After each guess, the program says if the guess is **too high**, **too low**, or **correct**.

💡 Hints

- Use random.randint() to pick a number.
- Use while loop to repeat until correct.
- Use if, elif, else to compare guesses.

📜 Procedure

- Import the random module.
- Generate a random number between 1 and 100.
- Use a while loop to repeatedly ask for guesses.
- Compare the guess:
- Too low → print "Too low!"
- Too high → print "Too high!"
- Correct → print success message & break loop

Solution

```
import random

number = random.randint(1, 100)
guess = None

while guess != number:
    guess = int(input("Guess a number (1–100): "))
    if guess < number:
        print("Too low!")
    elif guess > number:
        print("Too high!")
    else:
        print("🎉 Correct! You guessed it!")
```

🎮 2. Rock, Paper, Scissors

🎯 Concepts Used:

•if-elif-else
•while loop
•Functions

💡 Game Idea:

User plays rock-paper-scissors against the computer. First to 3 wins!

💡 Hints
- Use random.choice() for computer move.
- Use if-elif to compare user vs computer.
- Keep track of scores using variables.
- Wrap computer choice in a function for better code structure.

📜 Procedure
- Create a function get_computer_choice() that returns random choice.
- Use a while loop until either player reaches score 3.
- Compare moves:
- Rock beats Scissors
- Paper beats Rock
- Scissors beats Paper
- Update scores and print result.

Solution

```
import random

def get_computer_choice():
    return random.choice(['rock', 'paper', 'scissors'])

user_score = 0
computer_score = 0

while user_score < 3 and computer_score < 3:
    user = input("Rock, Paper, or Scissors? ").lower()
    computer = get_computer_choice()

    print("Computer chose:", computer)

    if user == computer:
        print("It's a tie!")
    elif (user == 'rock' and computer == 'scissors') or \
         (user == 'paper' and computer == 'rock') or \
         (user == 'scissors' and computer == 'paper'):
        print("You win this round!")
        user_score += 1
    else:
        print("Computer wins this round!")
        computer_score += 1

print("🏁 Final Scores -> You:", user_score, "| Computer:",
computer_score)
```

🎮 3. Even or Odd Game

🎯 Concepts Used:

- if-else
- functions
- while loop
- modulo (%)

💡 Game Idea:

User enters a number. Computer checks if it’s even or odd. The game continues until the user types exit.

💡 Hints
- Use % operator: if number % 2 == 0 → Even
- Use while True for infinite loop
- Use break to exit when input is 'exit'
- Use .isdigit() to validate input

📜 Procedure
- Define a function check_even_odd() to return "Even" or "Odd".
- Use a loop to ask the user for input continuously.
- If input is 'exit', stop the game.
- If input is a number, check if it is even or odd.
- Print the result.

Solution

```
def check_even_odd(num):
    if num % 2 == 0:
        return "Even"
    else:
        return "Odd"

while True:
    choice = input("Enter a number (or type 'exit' to stop): ")
    if choice.lower() == 'exit':
        print("Game over. Thanks for playing!")
        break
    if choice.isdigit():
        number = int(choice)
        result = check_even_odd(number)
        print(f"{number} is {result}")
    else:
        print("Please enter a valid number.")
```