🎮 ROCK-PAPER-SCISSORS
Name: Mikkili Koushik
Type: Micro IT Project

✅ Introduction
Rock-Paper-Scissors is a classic hand game often used as a decision-making tool. As a programming project, it provides a great opportunity to understand random number generation, input handling, and conditional logic. This Python-based game allows a human player to compete against the computer in an infinite loop until they choose to stop.

🎯 Objective of the Project
Understand basic Python programming constructs like conditionals, loops, and functions.

Implement interaction between the user and a computer.

Learn how to use randomness to simulate AI decisions.

Develop a simple yet functional terminal-based game.

🧰 Tools and Technologies Used
Programming Language: Python

Library: random (to simulate AI moves)

Platform: Any system with Python installed (console-based)

🧩 System Design and Architecture
Game Flow:

The game continuously runs until the player types "stop".

In each round:

The computer (AI) selects randomly between rock, paper, or scissors.

The player inputs their choice.

The program compares both choices to determine the winner or if it's a tie.

User Interface:

Console-based text UI.

Prompts user for input and displays results after every round.

💻 Implementation Details
1. AI Logic:

A list ["rock", "paper", "scissors"] is used.

random.choice() selects a random move for the computer.

2. Key Conditions:

If both choices match → Tie

If player’s move beats the AI → You Win

If AI’s move beats the player → You Lose

If player types "stop" → Game ends

3. Structure:

python
Copy code

```
while True:
    random_choice = random.choice(AI)
    user = input(...)
    # Compare logic
```

🛠️ Challenges Faced
Handling unexpected or invalid user inputs.

Ensuring smooth user interaction in a continuous loop.

Avoiding repeated code using clean conditional logic.

📚 Learning Outcomes
Gained hands-on experience with loops, input/output, and decision-making logic.

Understood how to use Python's random module effectively.

Practiced creating real-time interactive applications in the terminal.

Learned basic game logic implementation and flow control.

✅ Conclusion
This project helped reinforce core programming concepts in Python through a fun and engaging game. Although simple, it was a meaningful step in learning interactive application development and logical thinking.

🚀 Future Scope
Add a score tracker for both player and AI.

Allow best-of-n rounds instead of infinite play.

Introduce GUI using Tkinter or Pygame.

Handle invalid inputs gracefully with warnings.

Add a countdown timer for each move.