AI GAME FOR ROCK PAPER AND SCISSORS

A report on AI game for rock, paper and scissors

Submitted by- Prakhar Rai

Roll No. – 202401100400137

Library ID- 2428CSEAIML1114

Date – 10/03/2025

Institute – KIET Group Of Institutions

INTRODUCTION

It is a simple Rock-Paper-Scissors game in which the user competes against an AI. The AI moves randomly, and the basic rules decide the outcome. The basic events are imparted core into the powers of Python programming like user input handling, randomizing, and resolving the using conditionals as in this game.

Method

Implementation would be very simple as pitting one human opponent up against the computer's choice. The procedures heading into playing a game can be lined in order as such:

• User Input Processing: The program prompts the user to type his or her choice (rock, paper, or scissors). For an invalid input, the program will prompt the user again to input.

• AI move picking: AI picks one of the three options (rock, paper, or scissors) randomly using the python random.choice function.

• 3. Game Logic- Here are the rules that can lead to a dictionary named winning_combos is defined, mapping each move to the move it defeats.

- **Rock** beats **Scissors**
- **Paper** beats **Rock**
- **Scissors** beats **Paper**
- **If both are equal it's a tie**

• 4. Result display: The game presents the verdict (Win, Lose, Tie) based on comparison. The result is displayed in a more human friendly manner as we are producing a delay before giving result and we are also using emojis.

• 5. Game Loop: The game will keep running continuously unless the player wants to end it by typing quit. If the player decides to quit, the program finishes with a termination message.

Below is the Python implementation of the Rock-Paper-Scissors game. This script allows the user to play against an AI that randomly selects its move.

```python
import random
import time  # For adding slight delays to make the game feel more natural


# Define possible choices
choices = ["rock", "paper", "scissors"]


def get_ai_move():
    """ AI randomly picks rock, paper, or scissors """
    return random.choice(choices)


def determine_winner(player_move, ai_move):
    """ Determines the winner based on the choices """
    if player_move == ai_move:
        return "It's a tie! We both chose the same."

    winning_combos = {
        "rock": "scissors",
        "paper": "rock",
        "scissors": "paper"
    }

    if winning_combos[player_move] == ai_move:
        return "You win! 🎉"
    return "AI wins! 🤘 Better luck next time."


# Game introduction
```

```python
print("Welcome to Rock-Paper-Scissors! 🪨✋✌️")
print("Type 'quit' anytime to exit.\n")


# Main game loop
while True:
    # Get player input
    player_move = input("Your move (rock, paper, or scissors): ").lower()


    if player_move == "quit":
        print("\nThanks for playing! See you next time. 👋")
        break


    if player_move not in choices:
        print("Oops! That's not a valid move. Try again.\n")
        continue


    # AI makes a move
    print("\nThinking...")
    time.sleep(1)  # Add a slight delay for realism
    ai_move = get_ai_move()


    # Show AI's move
    print(f"AI chose: {ai_move.capitalize()}")


    # Determine and show the winner
    time.sleep(0.5)
    print(determine_winner(player_move, ai_move), "\n")
```
Below is a sample execution of the game:

```
Welcome to Rock-Paper-Scissors! ✊ ✋ ✌
Type 'quit' anytime to exit.

Your move (rock, paper, or scissors): rock

Thinking...
AI chose: Paper
AI wins! 🤖 Better luck next time.

Your move (rock, paper, or scissors): scissors

Thinking...
AI chose: Scissors
It's a tie! We both chose the same.

Your move (rock, paper, or scissors): quit

Thanks for playing! See you next time. 👋
```

References –

Python Official Documentation: https://docs.python.org/3/library/random.html

W3Schools Python Tutorial: https://www.w3schools.com/python/

GeeksforGeeks Python Programming: https://www.geeksforgeeks.org/python-programming-examples/

Real Python (Python Real-World Applications): https://realpython.com/

Stack Overflow (for coding discussions and troubleshooting): https://stackoverflow.com/

All these references helped me to successfully complete my project with proper logic building and troubleshooting some common coding issues.