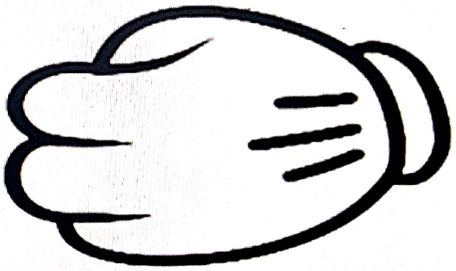
CREATE ROCK, PAPER, SCISSORS GAME IN PYTHON

Rock Paper Scissors

```
###Rock paper scissors game
import random
while True:
    my_answer=input("Choose: rock, paper
or scissors:")
    my_answer=my_answer.lower()
    if my_answer=="quit":
        break
    if my_answer!="rock" and
my_answer!="paper" and
my_answer!="scissors":
        print("Please choose a correct
answer")
        continue

computer_answer=random.choice(["rock","pap
er","scissors"])
    print(f"Computer choice:
{computer_answer}")
    if my_answer==computer_answer:
        print("You tied")
        continue
    elif my_answer=="paper" and
computer_answer=="rock":
        print("You Win")
        break
    elif my_answer=="rock" and
computer_answer=="scissors":
        print("You Win")
        break
    elif my_answer=="scissors" and
computer_answer=="paper":
        print("You Win")
        break
    else:
        print("You lose.Try again")
```

codingal.com/coding-for-kids/blog/rock-paper-scissors-game-in-python/#:~:text=python%20import%20random%



Here's a **detailed explanation** of your **Rock Paper Scissors Game Project** in Python

---

🧠 Project Title: Rock Paper Scissors Game

🎯 Project Goal

To create a simple **command-line game** in Python where the player competes against the computer by choosing between **rock**, **paper**, and **scissors**.

🕹️ How the Game Works

Rock, Paper, Scissors is a classic hand game usually played between two people. The rules are:

- **Rock beats Scissors**
- **Scissors beats Paper**
- **Paper beats Rock**
- If both players choose the same, it's a **tie**

In your program:

- The **user** will input their choice.
- The **computer** will make a random choice.
- The program will compare the choices and declare a winner.

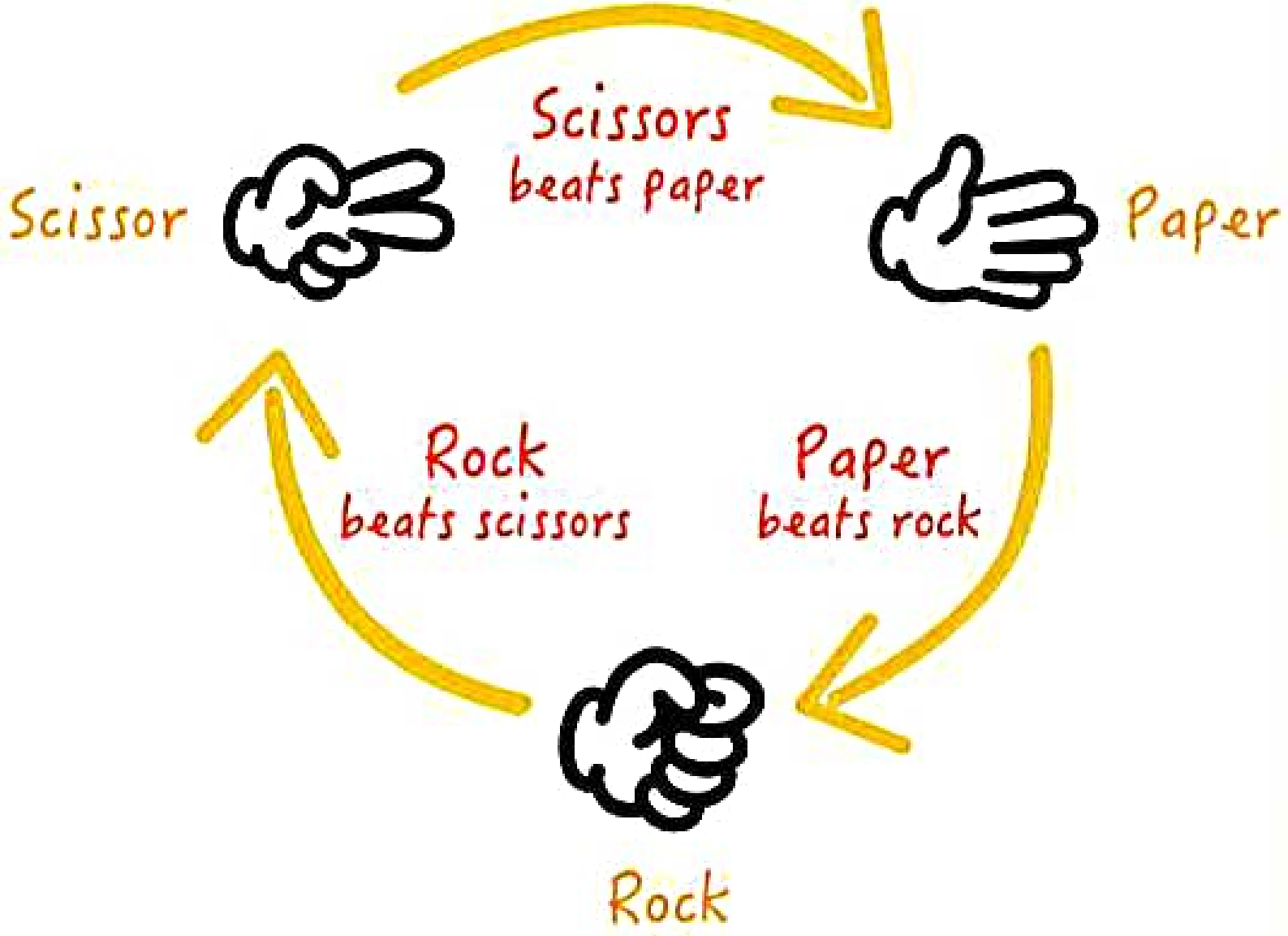
🔧 Technologies Used

Technology	Purpose
Python	Core programming language
`random` module	To make the computer choose randomly

📂 Project Structure

```
Rock-Paper-Scissors-Game/
├── rps.py           # Main game code
├── README.md        # Project details and instructions
└── .gitignore       # Ignores unnecessary files like __pycache__
```

Game rules


Scissors
beats paper
Scissor
Paper
Rock
beats scissors
Paper
beats rock
Rock

💻 How the Code Works – Step-by-Step

1. **Import Random Module**
   To allow the computer to make a random choice.

   ```
   import random
   ```

2. **Define Possible Choices**

   ```
   choices = ["rock", "paper", "scissors"]
   ```

3. **Get User Input**

   ```
   user_choice = input("Enter rock, paper or scissors: ").lower()
   ```

4. **Computer Makes a Choice**

   ```
   computer_choice = random.choice(choices)
   ```

5. **Print Choices**

   ```
   print(f"You chose {user_choice}, computer chose {computer_choice}")
   ```

6. **Determine the Winner**
   Use conditional `if-elif-else` statements.

   ```
   if user_choice == computer_choice:
       print("It's a tie!")
   elif (user_choice == "rock" and computer_choice == "scissors") or \
        (user_choice == "scissors" and computer_choice == "paper") or \
        (user_choice == "paper" and computer_choice == "rock"):
       print("You win!")
   else:
       print("Computer wins!")
   ```

7. **Optional: Add a Loop**
   Let the user play again without restarting the program.

✅ Features You Can Add (Bonus Ideas)

- Keep track of the score (Player vs Computer)
- Allow the user to quit any time by typing `exit`
- Add a graphical interface using `tkinter` or `pygame` (advanced)
- Show emojis (🪨📄✂️) to make it more fun

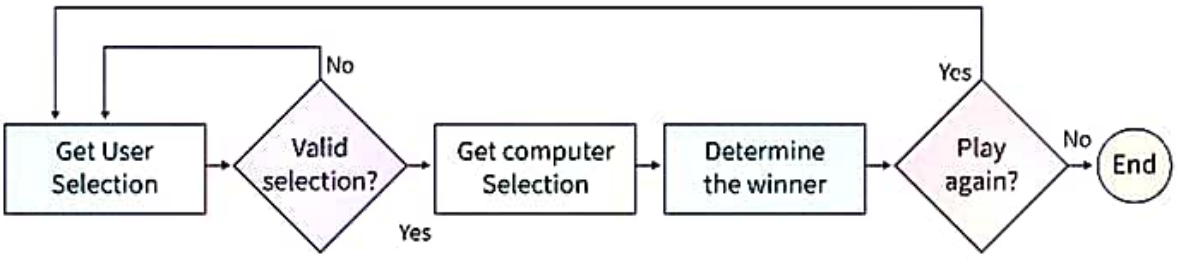
📸 Sample Output

```
Enter rock, paper or scissors: rock
You chose rock, computer chose scissors
You win!
```

📚 Learning Outcomes

By completing this project, you'll learn:

- How to use `if-else` logic
- How to work with user input
- How to use Python's `random` module
- How to structure a simple program_____


Get User Selection
Valid selection?
No
Yes
Get computer Selection
Determine the winner
Play again?
Yes
No
End

_____ ##Here's a **detailed explanation of the output** for your **Rock Paper Scissors game**, including what each part of the output means and how it is generated from the code.

- ### 🖥️ Sample Output (with full explanation)

```
Enter rock, paper or scissors: rock
You chose rock, computer chose scissors.
You win!
```

💡 Explanation of Each Line

1️⃣ `Enter rock, paper or scissors: rock`

- This line is **displayed by the program** using `input()`.
- It prompts the **user to enter their choice**.
- The user types `rock` (input is case-insensitive if `.lower()` is used).

Code:

```
user_choice = input("Enter rock, paper or scissors: ").lower()
```

2️⃣ `You chose rock, computer chose scissors.`

- After the user inputs their choice, the **computer picks randomly** using the `random.choice()` function.
- Then, the program prints both the user's and computer's choices.

Code:

```
computer_choice = random.choice(choices)
print(f"You chose {user_choice}, computer chose {computer_choice}.")
```

- Here, let's assume the **computer randomly picked `scissors`**.
- The output becomes:
  ✅ `"You chose rock, computer chose scissors."`

3️⃣ `You win!`

- The program uses **if-elif-else conditions** to determine the result.
- It compares `user_choice` (`rock`) and `computer_choice` (`scissors`).
- According to the rules:
  - **Rock beats Scissors**, so the user wins.

Code:

```
if user_choice == computer_choice:
    print("It's a tie!")
elif (user_choice == "rock" and computer_choice == "scissors") or \
     (user_choice == "scissors" and computer_choice == "paper") or \
     (user_choice == "paper" and computer_choice == "rock"):
    print("You win!")
else:
    print("Computer wins!")
```

🎯 Other Possible Outputs

Let's look at all possible outcomes with different inputs:

Case 1: Tie

```
Enter rock, paper or scissors: paper
You chose paper, computer chose paper.
It's a tie!
```

Case 2: You win

```
Enter rock, paper or scissors: scissors
You chose scissors, computer chose paper.
You win!
```

Case 3: Computer wins

```
Enter rock, paper or scissors: paper
You chose paper, computer chose scissors.
Computer wins!
```

⚠️ Invalid Input (Optional Feature)

If you add input validation, you can also show:

```
Enter rock, paper or scissors: pen
Invalid input. Please enter rock, paper, or scissors only.
```

Code (optional check):

```
if user_choice not in choices:
    print("Invalid input. Please enter rock, paper, or scissors only.")
```