**Stone-Paper-Scissors Game Using Python**

**Mini Project Report**

**Institute Name :** Sri Ramakrishna Institute of Technology

**Project Name :** Stone-Paper-Scissors Game Using Python

**Submitted to :** GOWTHAMI

**Submitted by :** HARINI M

**Co-ordinator Name :** THOORIGA

**Date of Submission :** 11/08/2025

**Acknowledgement**

I would like to express my sincere thanks to **Besant Technologies** for giving me the opportunity to attend the 30 Days Python Training Program.  
I am deeply grateful to my trainer, **Mrs. Gowthami Shyam**, for her valuable guidance, clear explanations, and continuous support throughout the course.  
I would also like to extend my heartfelt thanks to **Ms. Thooriga**, our coordinator, for her encouragement, timely updates, and smooth coordination during the training.

Finally, I thank all my classmates and friends for creating a supportive and motivating learning environment during these 30 days. This training has helped me improve my Python skills and build confidence to work on real-world projects.

**Table of Contents**

1. Cover Page
2. Acknowledgement
3. Table of Content
4. Introduction
5. Concepts Used in Project
6. Source Code (Program)
7. Working Procedure of Source Code
8. Output (Sample Results)
9. Conclusion
10. Bibliography(References)
11. Github link

**1. Introduction**

Stone-Paper-Scissors is one of the most popular and simple games played worldwide. It's usually played between two people using hand gestures. In this mini project, I have created a digital version of this classic game using Python programming language.

The game allows a human player to play against the computer. The computer makes random choices, making the game fair and unpredictable. This project helps beginners understand basic Python concepts like user input, random number generation, conditional statements, and loops.

**Game Rules:**

* Stone beats Scissors
* Scissors beats Paper
* Paper beats Stone
* Same choices result in a tie

**2. Objective**

The main objectives of this mini project are:

* To create a simple interactive game using Python
* To understand and implement basic programming concepts
* To learn how to handle user input and validation
* To use Python's random module for computer choices
* To apply conditional statements (if-elif-else) effectively
* To create a user-friendly gaming experience

**3. Concepts Used in Project**

This project demonstrates several important programming concepts:

**1. User Input Handling**

* Taking input from the user
* Converting input to lowercase for consistency
* Input validation

**2. Random Number Generation**

* Using Python's random module
* random.choice() function for selecting computer's move

**3. Conditional Statements**

* if-elif-else statements for game logic
* Multiple condition checking using logical operators

**4. Lists and Data Structures**

* Using lists to store game options
* Membership testing with in operator

**5. String Manipulation**

* Converting strings to lowercase
* String formatting with f-strings

**4. Source Code (Program)**

# Stone-Paper-Scissors Game covering all cases

import random

print("Welcome to Stone-Paper-Scissors Game ")

print("Choices: stone, paper, scissors")

options = ["stone", "paper", "scissors"]

# Get user input

user\_choice = input("Enter your choice: ").lower()

# Validate user input

if user\_choice not in options:

print("Invalid choice! Please type 'stone', 'paper', or 'scissors'.")

else:

# Computer randomly picks a choice

computer\_choice = random.choice(options)

print(f"\nYou chose: {user\_choice}")

print(f"Computer chose: {computer\_choice}\n")

# Decide the winner for all cases

if user\_choice == computer\_choice:

print("It's a tie! ")

elif (user\_choice == "stone" and computer\_choice == "scissors") or \

(user\_choice == "paper" and computer\_choice == "stone") or \

(user\_choice == "scissors" and computer\_choice == "paper"):

print(" Congratulations You win! ")

else:

print(" Oops!! Computer wins! ")

**5. Working Procedure of Source Code**

**Step 1: Import Required Module**

* Import random module to generate computer's choice

**Step 2: Display Game Instructions**

* Show welcome message
* Display available choices to the user

**Step 3: Get User Input**

* Prompt user to enter their choice

**Step 4: Validate User Input**

* Check if user's choice is valid (stone, paper, or scissors)
* Display error message for invalid inputs

**Step 5: Generate Computer's Choice**

* Use random.choice() to select computer's move randomly

**Step 6: Display Both Choices**

* Show what user selected
* Show what computer selected

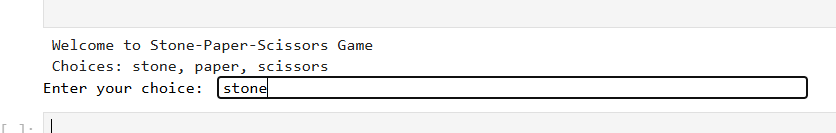
**Step 7: Apply Game Logic**

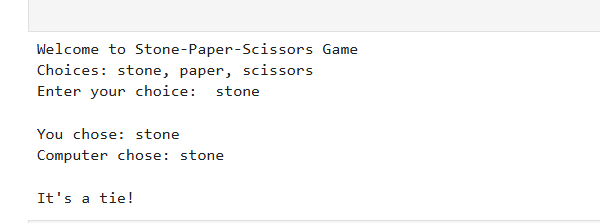
* Compare user's choice with computer's choice
* Apply game rules to determine the winner
* Display the result (win/lose/tie)

**Step 8: End Game**

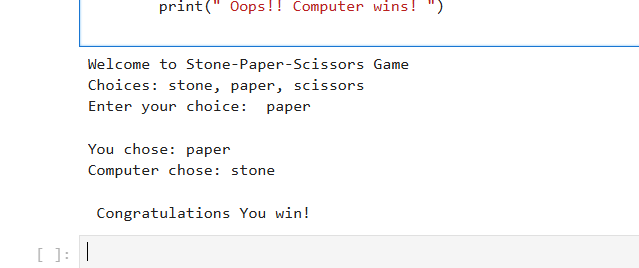
* Show final result with appropriate message

**6. Output (Sample Results)**

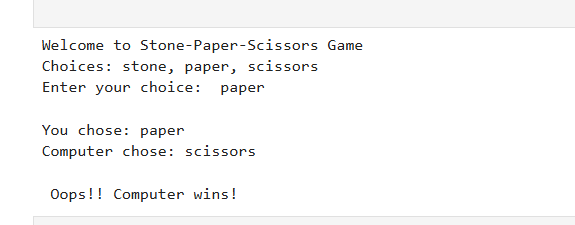
**Output 1:**   
****

**Output 2:**   
****

**Output 3:**

****

Output 4:

****

**7. Conclusion**

This Stone-Paper-Scissors game project successfully demonstrates the implementation of a simple interactive game using Python programming language. Through this project, I learned several important programming concepts:

**Key Learning Outcomes:**

* Understanding of basic Python syntax and structure
* Practical use of conditional statements and logical operators
* Implementation of user input handling and validation
* Application of Python's random module
* Creating user-friendly program interfaces

**Project Benefits:**

* Enhanced problem-solving skills
* Better understanding of game logic implementation
* Improved coding practices and code organization
* Experience with handling user interactions

**8. Bibliography/References**

1. Python Random Module Documentation
2. "Learn Python the Hard Way" by Zed Shaw
3. GeeksforGeeks Python Tutorials
4. W3Schools Python Tutorial
5. Stack Overflow - Python Programming Community

**9.GITHUB LINK :**

https://github.com/Harini27-wq/Python\_mini\_project/