|  |  |
| --- | --- |
| Name :Ross Paolo Villavicencio | Section : BSIT 3A |
| Assignment No : 2 | Submission Date : Mar. 23, 2024 |
| Assignment Title: Rock, Paper, Scissors | |

**Code :**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Rock Paper Scissors</title>

<link rel="stylesheet" href="styless.css">

</head>

<body>

<div class="container">

<h1>Rock Paper Scissors</h1>

<div class="scoreboard">

<p>You: <span id="player-score">0</span></p>

<p>Computer: <span id="computer-score">0</span></p>

</div>

<div class="picks">

<p>Player Pick: <span id="player-pick"></span></p>

<p>Computer Pick: <span id="computer-pick"></span></p>

</div>

<div class="choices">

<button onclick="play('rock')">👊</button>

<button onclick="play('paper')">✋</button>

<button onclick="play('scissors')">✌</button>

</div>

<div id="result"></div>

</div>

<script src="scripts.js"></script>

</body>

</html>

**JS FILE**

let playerScore = 0;

let computerScore = 0;

function play(playerChoice) {

const choices = ['rock', 'paper', 'scissors'];

const computerChoice = choices[Math.floor(Math.random() \* 3)];

document.getElementById('player-pick').innerText = playerChoice.toUpperCase();

document.getElementById('computer-pick').innerText = computerChoice.toUpperCase();

if (playerChoice === computerChoice) {

showResult('It\'s a TIE');

} else if ((playerChoice === 'rock' && computerChoice === 'scissors') ||

(playerChoice === 'paper' && computerChoice === 'rock') ||

(playerChoice === 'scissors' && computerChoice === 'paper')) {

playerScore++;

showResult('You Win');

} else {

computerScore++;

showResult('You Lose');

}

updateScore();

}

function updateScore() {

document.getElementById('player-score').innerText = playerScore;

document.getElementById('computer-score').innerText = computerScore;

}

function showResult(result) {

const resultElement = document.getElementById('result');

resultElement.innerText = result;

if (result === 'You Win') {

resultElement.style.color = 'green';

} else if (result === 'You Lose') {

resultElement.style.color = 'red';

} else {

resultElement.style.color = 'black'; // Change text color to black for tie

}

}

**CSS FILE**

.container {

text-align: center;

margin-top: 50px;

}

.choices button {

font-size: 100px;

margin: 20px;

border: none;

background: none;

cursor: pointer;

width: 100px;

height: 100px;

}

.choices button img {

font-size: larger;

width: 100px;

height: 100%;

border-radius: 100%;

}

.scoreboard {

margin-bottom: 20px;

}

#result {

margin-top: 20px;

font-size: 24px;

}

.win {

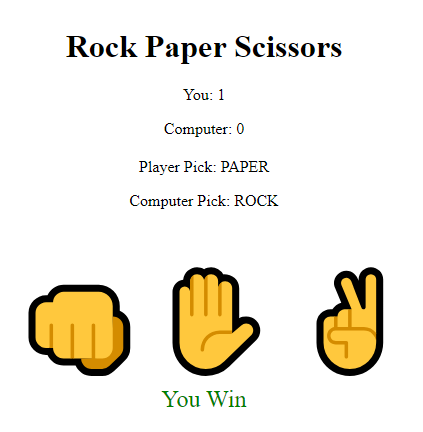
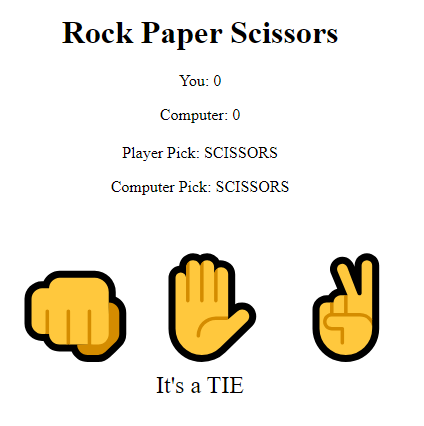
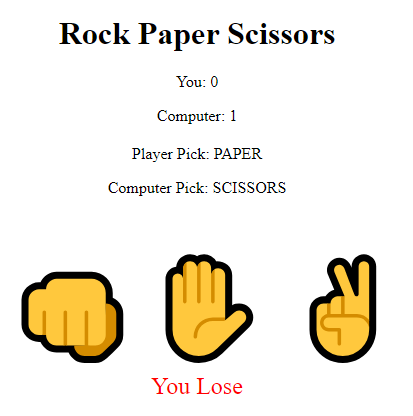
color: green;

}

.lose {

color: red;

}

**SCREENSHOT**

**Dictionary:**

1. **play(playerChoice)** - A function that simulates a round of the rock-paper-scissors game, taking the player's choice as input.

**Parameter** – The choice made by the player ('rock', 'paper', or 'scissors').  
**Functionality:**

* Selects a random choice for the computer from the available options.
* Displays the player's and computer's choices on the HTML page.
* Determines the winner based on the choices and updates the scores accordingly.
* Calls the updateScore() function to update the displayed scores.\

1. **updateScore()** - A function that updates the displayed scores on the HTML page.  
   **Functionality:** Retrieves the player's and computer's scores and updates them on the HTML page.
2. **showResult(result)** - A function that displays the result of each round of the game.  
   **Parameter:** The outcome of the round ('You Win', 'You Lose', or 'It's a TIE').   
   **Functionality:**- Displays the result of the round on the HTML page.

* Changes the color of the displayed result based on the outcome.

1. **playerScore -** A variable that stores the score of the player in the rock-paper-scissors game.  
   **Initial Value:** 0
2. **computerScore** - A variable that stores the score of the computer in the rock-paper-scissors game.  
   **Initial Value:** 0