**HTML CODE:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title>Rock Paper Scissors! Class Project</title>

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

</head>

<body>

<div id="game">

<!--All game functionality happens in here-->

<div id="mainMenu" class="overlay">

<!--Main Menu options go here-->

<div>

<button type="button" onclick="openGame()">

<img src="charmander.jpg" width="200" height="100" alt="Play">

</button>

<br>

<button type="button" onclick="openShop()">

<img src="gastly.png" width="200" height="100" alt="shop">

</button>

<br>

<!--Test for pause function on empty shop-->

<button type="button" onclick="openNav()">

<img src="pika sit.jpg" width="200" height="100" alt="options">

</button>

<br>

<button type="button" class="choice" id="reset">

<img src="megaman.jpg" width="200" height="100" alt="reset">reset </button>

<br>

</div>

</div>

<div id="optionsMenu" class="overlay">

<!--Settings options go here (volume, etc.)-->

<div id="Options">

<a href="javascript:void(0)" class="closebtn" onclick="closeNav()">&times;</a>

<p>Sound FX</p>

<button id="soundSetting" onclick="soundToggle()">I/O</button>

<p>Music</p>

<button id="musicSetting" onclick="musicToggle()">I/O</button>

<p>Etc</p>

<button type="button" class="Menu" onclick="closeNav()">

<p>Back</p>

</button>

</div>

</div>

<div id="pauseMenu" class="overlay">

<!--Pause options go here-->

<a href="javascript:void(0)" class="closebtn" onclick="closePause()">&times;</a>

<button type="button" class="Menu" onclick="closePause()">

<p>Resume Game</p>

</button>

<br>

<button class="menu" onclick="openNav()">Options</button>

<br>

<button type="button" onclick="openMenu()">Main Menu</button>

</div>

<div id="scoresTable" class="overlay">

<!--Scores Table goes here-->

<a href="javascript:void(0)" class="closebtn" onclick="closeScores()">&times;</a>

<fieldset id="entryScore">

<legend>Insert Name</legend>

<input id="plName" type="text" placeholder="Enter Name Here">

<input type="submit" value="Submit Name" id="submitButton">

<button id="refresh" style="background: url(Random.png) no-repeat center;" class="choice">Confirm</button>

<button id="score" style="background: url(Random.png) no-repeat center;" class="choice">View High Scores</button>

</fieldset>

<fieldset id="displayScore">

<legend>RECENT SCORES</legend>

<ol id='list'>

<li>

<a id="link1">No High Score set</a>

</li>

<li>

<a id="link2">No High Score set</a>

</li>

<li>

<a id="link3">No High Score set</a>

</li>

<li>

<a id="link4">No High Score set</a>

</li>

<li>

<a id="link5">No High Score set</a>

</li>

</ol>

<div id="lsOutput"></div>

</fieldset>

<button type="button" class="Menu" onclick="closeScores()">

<p>Back</p>

</button>

</div>

<div id="pointsShop">

<!--Points Shop options go here-->

</div>

<div id="field">

<h2>Computer Choice: <span id="computer-choice"></span>

</h2>

<h2>Your Choice: <span id="user-choice"></span>

</h2>

<h2>Result: <span id="result"></span>

</h2>

<h2>Total Games Played: <span id="totalGames"></span>

</h2>

<h2>Your Score: <span id="theScore"></span>

</h2>

<h2>COM Score: <span id="compScore"></span>

</h2>

<h2>Draws: <span id="drawScore"></span>

</h2>

<!--Where the gameplay happens-->

<div id="rockPaperScissors">

<table>

<!--Rock/Paper/Scissors/Random options go here-->

<tr>

<button id="rock" style="background: url(Rock.png) no-repeat center;" class="choice">rock</button>

<button id="paper" style="background: url(Paper.png) no-repeat center;" class="choice">paper</button>

<button id="scissors" style="background: url(Scissors.png) no-repeat center;" class="choice">scissors</button>

<button id="random" style="background: url(Random.png) no-repeat center;" class="choice">random</button>

</tr>

</table>

</div>

<div id="lizardSpock">

<table>

<!--Rock/Paper/Scissors/Lizard/Spock options go here-->

<tr>

<td>rock</td>

<!--Rock-->

<td>paper</td>

<!--Paper-->

<td>scissors</td>

<!--Scissors-->

<td>lizard</td>

<!--Lizard-->

<td>spock</td>

<!--Spock-->

<td>random</td>

<!--Random-->

</tr>

</table>

</div>

</div>

<a href="javascript:void(0)" id="scoresButton" onclick="toggleScores()">

<div> Scores </div>

</a>

<a href="javascript:void(0)" id="pauseButton" onclick="togglePause()">

<div> Pause </div>

</a>

</div>

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

</body>

</html>

**JS CODE:**   
  
//----------MENU FUNCTIONS------------

/\* Open the options menu \*/

function openNav() {

//document.getElementById("optionsMenu").style.height = "600px";

document.getElementById("optionsMenu").style.display = "block";

}

//Toggle Sound

function soundToggle() {

if (document.getElementById("soundSetting").style.backgroundColor === "orange") {

document.getElementById("soundSetting").style.backgroundColor = "antiquewhite";

//Put in code later to enable all sound effects except music

} else {

document.getElementById("soundSetting").style.backgroundColor = "orange";

//Put in code later to disable all sound effects except music

}

}

//Toggle Music

function musicToggle() {

if (document.getElementById("musicSetting").style.backgroundColor === "orange") {

document.getElementById("musicSetting").style.backgroundColor = "antiquewhite";

//Put in code later to enable all music.

} else {

document.getElementById("musicSetting").style.backgroundColor = "orange";

//Put in code later to disable all music

}

}

/\* Close the options menu \*/

function closeNav() {

//document.getElementById("optionsMenu").style.height = "0px";

document.getElementById("optionsMenu").style.display = "none";

}

/\* Open the pause menu \*/

function togglePause() {

if (document.getElementById("pauseMenu").style.display === "block") {

//document.getElementById("pauseMenu").style.height = "0px";

document.getElementById("pauseMenu").style.display = "none";

} else {

//document.getElementById("pauseMenu").style.height = "600px";

document.getElementById("pauseMenu").style.display = "block";

}

}

/\*

function openPause() {

//document.getElementById("pauseMenu").style.height = "600px";

document.getElementById("pauseMenu").style.display = "block";

}\*/

/\* Close the pause menu\*/

function closePause() {

//document.getElementById("pauseMenu").style.height = "0px";

document.getElementById("pauseMenu").style.display = "none";

}

//Play the game, close the main menu

function openGame() {

//document.getElementById("mainMenu").style.height = "0px";

document.getElementById("mainMenu").style.display = "none";

document.getElementById("pauseButton").style.display = "block";

document.getElementById("scoresButton").style.display = "block";

}

/\* Close \*/

function closeGame() {

document.getElementById("Play").style.height = "0%";

}

//Open the Main Menu

function openMenu() {

document.getElementById("mainMenu").style.display = "block";

document.getElementById("pauseButton").style.display = "none";

document.getElementById("pauseMenu").style.display = "none";

document.getElementById("scoresButton").style.display = "none";

}

//Toggle Scores

function toggleScores() {

if (document.getElementById("scoresTable").style.display === "block") {

document.getElementById("scoresTable").style.height = "0px";

document.getElementById("scoresTable").style.display = "none";

} else {

document.getElementById("scoresTable").style.height = "600px";

document.getElementById("scoresTable").style.display = "block";

}

}

function closeScores() {

document.getElementById("scoresTable").style.height = "0px";

document.getElementById("scoresTable").style.display = "none";

}

//Open the points shop

/\*function toggleShop() {

if (document.getElementById("pointsShop").style.display === "none"){

document.getElementById("pointsShop").style.height = "600px";

document.getElementById().style.display = "block";

}

else {

document.getElementById("pointsShop").style.height = "0px";

document.getElementById().style.display = "none";

}

}\*/

//----------Rock Paper Scissors Javascript---------------

//Matt Gates v01-with highscore functionality

const computerChoiceDisplay = document.getElementById('computer-choice');

const userChoiceDisplay = document.getElementById('user-choice');

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

const possibleChoices = document.querySelectorAll('.choice');

const inpName = document.getElementById("plName");

const btInsert = document.getElementById("submitButton");

const lsOutput = document.getElementById("lsOutput");

let userChoice;

let computerChoice;

let result;

let score = 0;

let highScore = 0;

let compScore = 0;

let drawScore = 0;

let totalGames = 0;

let scoreDisplay = 1;

let check = 0;

/\*Matt Gates: Addition for High Score

window.onload = function(){

scoreFromBrowser = localStorage.getItem("highScore");

if(scoreFromBrowser != undefined){

highScore = scoreFromBrowser;

}

document.getElementById("highScore").innerHTML = highScore;

}\*/

possibleChoices.forEach(possibleChoice => possibleChoice.addEventListener('click', (e) => {

userChoice = e.target.id

userChoiceDisplay.innerHTML = userChoice

generateComputerChoice()

if (userChoice === "random") {

const randomNumber = Math.floor(Math.random() \* 3) + 1

if (randomNumber === 1) {

userChoice = 'rock'

}

if (randomNumber === 2) {

userChoice = 'scissors'

}

if (randomNumber === 3) {

userChoice = 'paper'

}

}

userChoiceDisplay.innerHTML = userChoice

getResult()

let yourScore = score.toString();

document.getElementById('theScore').innerHTML = yourScore;

scoringSystem();

/\*Matt Gates: Addition for High Score

let yourHighScore = highScore.toString();

document.getElementById('highScore').innerHTML = yourHighScore;

\*/

}))

function generateComputerChoice() {

const randomNumber = Math.floor(Math.random() \* 3) + 1 // or you can use possibleChoices.length

if (randomNumber === 1) {

computerChoice = 'rock'

}

if (randomNumber === 2) {

computerChoice = 'scissors'

}

if (randomNumber === 3) {

computerChoice = 'paper'

}

computerChoiceDisplay.innerHTML = computerChoice

}

function getResult() {

if (computerChoice === userChoice) {

result = 'its a draw!'

check = 0;

}

if (computerChoice === 'rock' && userChoice === "paper") {

result = 'you win!'

check = 1;

}

if (computerChoice === 'rock' && userChoice === "scissors") {

result = 'you lose!'

check = 0;

}

if (computerChoice === 'paper' && userChoice === "scissors") {

result = 'you win!'

check = 1;

}

if (computerChoice === 'paper' && userChoice === "rock") {

result = 'you lose!'

check = 0;

}

if (computerChoice === 'scissors' && userChoice === "rock") {

result = 'you win!'

check = 1;

}

if (computerChoice === 'scissors' && userChoice === "paper") {

result = 'you lose!'

check = 0;

}

if (userChoice === 'refresh') {

result = "Refreshing page";

window.alert("Name is confirmed. Click to view high scores.");

SortLocalStorage()

let refresh1Element = document.getElementById("link1");

let refresh = "Refreshing"

refresh1Element.innerHTML = refresh;

let refresh2Element = document.getElementById("link2");

refresh2Element.innerHTML = refresh;

let refresh3Element = document.getElementById("link3");

refresh3Element.innerHTML = refresh;

let refresh4Element = document.getElementById("link4");

refresh4Element.innerHTML = refresh;

let refresh5Element = document.getElementById("link5");

refresh5Element.innerHTML = refresh;

}

if (userChoice === 'score') {

result = "Submitting";

window.alert("Current High scores. Scores will only save if you submit and confirm your name details.");

SortLocalStorage()

const storageScoreArray = new Array();

for (let i = 0; i < sortArray.length; i++) {

storageScoreArray[i] = sortArray[i].split(";");

}

let storage1Refresh = document.getElementById("link1");

let storage1 = inpName.value + " : " + score;

storage1Refresh.innerHTML = storage1;

let storage2Refresh = document.getElementById("link2");

let storage2 = storageScoreArray[1][1]

storage2Refresh.innerHTML = storage2;

let storage3Refresh = document.getElementById("link3");

let storage3 = storageScoreArray[2][1]

storage3Refresh.innerHTML = storage3;

let storage4Refresh = document.getElementById("link4");

let storage4 = storageScoreArray[3][1]

storage4Refresh.innerHTML = storage4;

let storage5Refresh = document.getElementById("link5");

let storage5 = storageScoreArray[4][1]

storage5Refresh.innerHTML = storage5;

console.log(localStorage);

}

resultDisplay.innerHTML = result

if (check === 1) {

score = score + 1

}

///Matt Gates: Addition for High Score

resultDisplay.innerHTML = result

if (score > highScore) {

highScore = score;

}

}

//localStorage.setItem("highScore", highScore);

//document.getElementById("highScore").innerHTML="High Score: " + highScore;

function scoringSystem() {

totalGames++;

switch (result) {

case 'you win!':

score++;

break;

case 'you lose!':

compScore++;

score--;

break;

default:

drawScore++;

}

/\*Idea for bonuses\*/

if (totalGames === 10 && score > compScore) {

window.alert("Bonus 5 points for being ahead after 10 games!");

score += 5;

} else if (totalGames === 10 && score < compScore) {

window.alert("Subtracting 5 points for being behind after 10 games!");

score -= 5;

}

/\*Updating the values in the HTML document\*/

document.getElementById('theScore').innerHTML = score;

document.getElementById('compScore').innerHTML = compScore;

document.getElementById('drawScore').innerHTML = drawScore;

document.getElementById('totalGames').innerHTML = totalGames;

}

/\*\*Anon function to store player name & score \*/

btInsert.onclick = function() {

const dateID = Date.now();

const key = dateID;

const value = inpName.value + " : " + document.getElementById('theScore').innerHTML;

/\* console.log(key);

console.log(value);\*/

if (key && value) {

localStorage.setItem(key, value);

//btInsert.onclick = getElementById("lsOutput").reload();

//onclick=localstorage.clear();

//document.getElementById("theScore").innerHTML;

}

};

let sortArray = SortLocalStorage();

function SortLocalStorage() {

if (localStorage.length > 0) {

var localStorageArray = new Array();

for (i = 0; i < localStorage.length; i++) {

localStorageArray[i] = localStorage.key(i) + ";" + localStorage.getItem(localStorage.key(i));

}

}

var sortedArray = localStorageArray.sort();

sortedArray.reverse();

return sortedArray;

}

const scoreArray = new Array();

for (let i = 0; i < sortArray.length; i++) {

scoreArray[i] = sortArray[i].split(";");

}

let link1Element = document.getElementById("link1");

let link1 = scoreArray[0][1]

link1Element.innerHTML = link1;

let link2Element = document.getElementById("link2");

let link2 = scoreArray[1][1]

link2Element.innerHTML = link2;

let link3Element = document.getElementById("link3");

let link3 = scoreArray[2][1]

link3Element.innerHTML = link3;

let link4Element = document.getElementById("link4");

let link4 = scoreArray[3][1]

link4Element.innerHTML = link4;

let link5Element = document.getElementById("link5");

let link5 = scoreArray[4][1]

link5Element.innerHTML = link5;

/\*\*For loop to display all recent scores that are stored into local storage

for(let i = 0; i < 5; i++){

if (i > scoreArray.length){

lsOutput.innerHTML += scoreDisplay + " No Score" + `<br>`;

}

else{

const key = scoreArray[i][1];

lsOutput.innerHTML += scoreDisplay + " " + `${key} <br>`;

}

//document.getElementById("plName").innerHTML = inpName

scoreDisplay++;

} \*/

//${value}

/\*

const key = localStorage.key(i);

const value = localStorage.getItem(key);\*/

function reset() {

document.getElementById("reset").onclick = localStorage.clear()

}