## **PROJECT: ROCK PAPER SCISSORS**

### **Introduction**

We’re going to make a simple implementation of grade-school classic “rock paper scissors”. If you don’t know what that is, check the [Wikipedia article](https://en.wikipedia.org/wiki/Rock%E2%80%93paper%E2%80%93scissors) or [this](https://www.wikihow.com/Play-Rock,-Paper,-Scissors) ridiculous step-by-step. For the moment we’re just going to play the game from the browser console, but we will revisit this project in a later lesson and add a Graphical User Interface with buttons and text, so don’t forget to keep the code on GitHub! You might notice some ‘Live Preview’ links in the student solutions that have a GUI - this is coming in a later lesson. When you get there don’t forget to come back and add your link!

### **Quick Exercises Before Starting**

1. Identify three ways to include JavaScript in a page.
2. Test it out! Write console.log("Hello World"); in JavaScript and check to see if it displays in the browser’s console.
3. If you haven’t used it already, this is where the [VSCode Live Server extension](https://www.theodinproject.com/courses/foundations/lessons/text-editors#additional-set-up-live-server-extension-for-vscod) becomes handy!

Finally, this is your first JavaScript program built from scratch, so don’t forget the previous lesson on problem solving. Plan your solution out before writing any code, and test each piece as you build to ensure it is working before moving on to the next!

### **Assignment**

1. Start a new Git repo for your project.
2. Create a blank HTML document with a script tag (Hint: it is best practice to link an external .js file). This game is going to be played completely from the console, so don’t worry about putting anything else in there.
3. Your game is going to play against the computer, so begin with a function called computerPlay that will randomly return either ‘Rock’, ‘Paper’ or ‘Scissors’. We’ll use this function in the game to make the computer’s play. *Tip: use the console to make sure this is returning the expected output before moving to the next step!*
4. Write a function that plays a single round of Rock Paper Scissors. The function should take two parameters - the playerSelection and computerSelection - and then return a string that declares the winner of the round like so: "You Lose! Paper beats Rock"
   1. Make your function’s playerSelection parameter case-insensitive (so users can input rock, ROCK, RocK or any other variation).
5. Important note: you want to return the results of this function call, *not* console.log() them. To test this function console.log the results:  
   function playRound(playerSelection, computerSelection) { // your code here! } const playerSelection = "rock"; const computerSelection = computerPlay(); console.log(playRound(playerSelection, computerSelection));
6. Write a NEW function called game(). Use the previous function *inside* of this one to play a 5 round game that keeps score and reports a winner or loser at the end.
   1. You have not officially learned how to “loop” over code to repeat function calls… if you already know about loops from somewhere else (or if you feel like doing some more learning) feel free to use them. If not, don’t worry! Just call your playRound function 5 times in a row. Loops are covered in the next lesson.
   2. At this point you should be using console.log() to display the results of each round and the winner at the end.
   3. Use prompt() to get input from the user. [Read the docs here if you need to.](https://developer.mozilla.org/en-US/docs/Web/API/Window/prompt)
   4. Feel free to rework your previous functions if you need to. Specifically, you might want to change the return value to something more useful.
   5. Feel free to create more “helper” functions if you think it would be useful.

### **Examples**

* <https://lookingcoolonavespa.github.io/rock-paper-scissors/>
* <https://michalosman.github.io/rock-paper-scissors/>
* <https://rlmoser99.github.io/rock-paper-scissors/>
* <https://mooniidev.github.io/rock-paper-scissors-game/>