Lesson 2: Rock, Paper, Scissors

In this lesson, you will play against the computer in a game of Rock, Paper, Scissors. We will use the random module and nested ifs to make the game.

Part 1 – Computer and Player Choice.

Create a new file like before and save it as **YourName LastInitial Lesson 2 - RPS.py** in your folder.

Let's start the program by importing the random module. By importing a module, we can use functions the functions inside of it. We will use the function random.randint(int a, int b), that can make random numbers. Here, we'll set a variable called comp_choice to a random number between 1-3.

```
1  import random
2
3  print("I challenge you to a battle of rock, paper, scissors")
4  comp_choice = random.randint(1, 3)
```

Right now, comp_choice is set to a number. We could leave it like that, but then our code won't be as clear later on. To help us out with this, we'll use a simple conditional to change the value of comp_choice to one of 3 strings: "rock", "paper", or "scissors".

```
4  comp_choice = random.randint(1, 3)
5  if (comp_choice == 1):
6      comp_choice = "rock"
7  elif (comp_choice == 2):
8      comp_choice = "paper"
9  else:
10     comp_choice == "scissors"
```

Now that the computer had made its choice, let's ask for the player's choice.

```
10     comp_choice == "scissors"
11
12 print("I have made my selection. Now it is your turn")
13 player_choice = input("Tell me: rock, paper or scissors?")
14 print("You chose", player_choice, "I chose", comp_choice)
```

Test your code. After you input your choice, the computer should print out your choice and a random computer choice. Make sure the computer choice is actually making random choices.

Part 2 – Checking For A Winner.

We need to write a conditional that checks each of the player's possible choices. Let's start by checking if the player chose rock.

```
14 print("You chose", player_choice, "I chose", comp_choice)
15
16 if (player_choice == "rock"): #Player chooses rock
```

Inside this if, we will nest another conditional to check for the computer's choices. Remember, in order for the code inside the nested if to be executed, both the outer and inner conditions need to be true. So, if this print statement is executed, then that means that the battle is "rock" vs "rock" which would be a tie.

```
16 if (player_choice == "rock"): #Player chooses rock
17     if (comp_choice == "rock"):
18         print("It's a draw") # Rock vs Rock
```

Let's add on to the nested conditional to code the rest of the possible choices for the computer. Remember for all of these conditionals, we know that the player has chosen "rock" so all you really need to think about is if the computer's choice can beat "rock".

```
16 if (player_choice == "rock"): #Player chooses rock
17     if (comp_choice == "rock"):
18         print("It's a draw") # Rock vs Rock
19     elif (comp_choice == "paper"):
20         print("You have lost!") # Rock vs Paper
21     else:
22         print("You have won!") # Rock vs Scissors
```

Right now, our game works perfect if the player only chooses rock, but what kind of RPS game only lets you pick rock? We'll add on to the outer conditional with elif to code the rest of the game.

If you look closely, you'll see that the code is structurally the same as the code we wrote before. The only difference is that computer has chosen paper, so you must check the player's choice against paper. The differences are highlighted below.

```
16 if (player_choice == "rock"): #Player chooses rock
17     if (comp_choice == "rock"):
18         print("It's a draw") # Rock vs Rock
19     elif (comp_choice == "paper"):
20         print("You have lost!") # Rock vs Paper
21     else:
22         print("You have won!") # Rock vs Scissors
23 elif (player_choice == "paper"): #Player chooses paper
24     if (comp_choice == "rock"):
25         print("You have won!") # Paper vs Rock
26     elif (comp_choice == "paper"):
27         print("It's a draw") # Paper vs Paper
28     else:
29         print("You have lost!") # Paper vs Scissors
```

PAY CLOSE ATTENTION TO THE INDENTATION! The elif is aligned with the outer if and has its own nested conditional inside.

Part 3 – Finishing the Game

Try finishing up the game yourself. All you need to do is add another elif to check if the player chose "scissors" then nest a conditional that checks all of the computer's choices. It will be very similar to the code that we just wrote in the last section. You could even copy and paste lines 23-29 and just change the highlighted portions shown above. Give it an honest effort and if you need help ask your teacher.

BONUS: Do you remember what we mentioned about user input in the last lesson? It's important to account for the fact that a person might not follow your instructions. Think about what happens in our program when the player doesn't type *rock*, *paper*, or *scissors*. Try to fix this the same way we did in the last lesson.