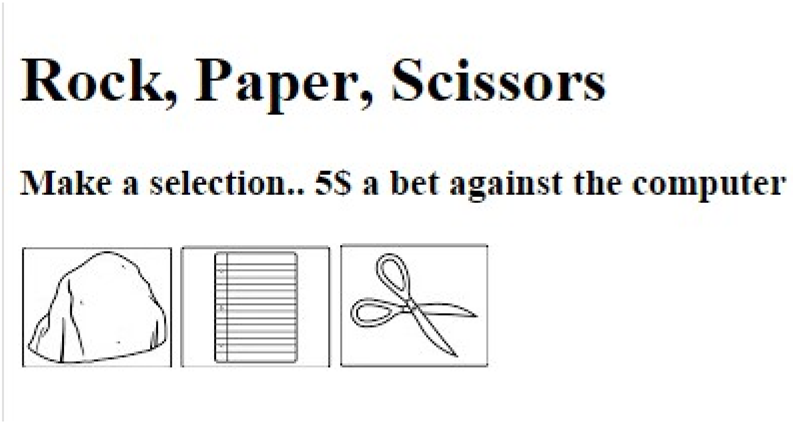
Lab 5: Rock Paper Scissors

You're gonna code a rock, paper, scissors game!

Here's what it looks like:



Rock, Paper, Scissors

Make a selection.. 5$ a bet against the computer

The first screen.

Click on, say, **rock**

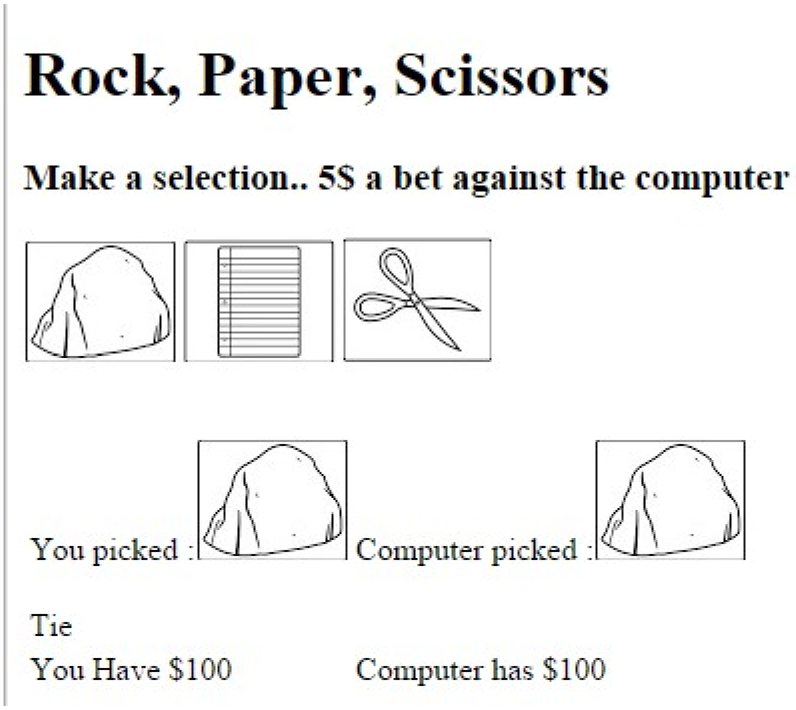
Looks like a **tie**.

You and the computer start with $100.

Winners get 5$; losers lose 5$

The program displays the *round status* (here, a **tie**) and adjusts $$$.

Notice the initial screen did not show monies.



Rock, Paper, Scissors

Make a selection.. 5$ a bet against the computer

You picked : Computer picked :

Tie

You Have $100 Computer has $100

Keep playing until you get bored.

As you might imagine, you have a *starter file named **rps.html*** to get you started.

Before we look at your starter file, a few comments on the overall processing are in order.

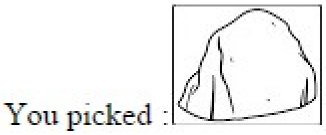
The Program at 30,000 Feet

You have the images used (the rock, etc.) in your *lab5* folder. We use the *names of the image files* to identify whether you (and the computer) selected rock, paper or scissors.

There's also two variables representing the monies.

The HTML for the program is already coded. The images are *buttons* that, when clicked, call a JavaScript function named **play** that gets everything moving. The **play** function:

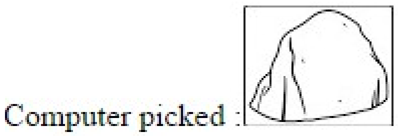
Uses a function already provided in *miscfuncs.js* to *display your selection.*



You picked :

Calls a function **that you will write** to have the computer make a selection.

Calls the same function in *miscfuncs.js* that displayed your selection to *display the computer selection*.

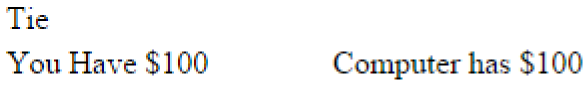


Computer picked :

Calls a function **that you will write** to determine **who won/lost, return a status string and adjust monies**.

The **status string** is either *Tie, You Won* or *You Lost*.

Finally, the **play** function calls a function already provided in *miscfuncs.js* that *displays the status string and monies*.



```
Tie
You Have $100              Computer has $100
```

The starter HTML file has beaucoup comments but we can take a look at it and the external JavaScript file containing functions that dynamically update the page.

miscfuncs.js

Here's the file:

```
// Display the computer pick using image returned by random generation
function showPicks( whoPickedId, imageFile ) {
        document.getElementById( whoPickedId ).innerHTML =
                "<p>" + whoPickedId + " picked :<img src='" + imageFile + "'/></p>"

}
// Show status of round and monies
function showStatusAndMonies( status, yourLoot, computerLoot ) {
        // Show status
        document.getElementById( 'status' ).innerHTML = status ;
        // Show monies
        document.getElementById( 'yourMoney' ).innerHTML = "You Have $" + yourLoot ;
        document.getElementById( 'computerMoney' ).innerHTML =
                                                  "Computer has $" + computerLoot ;
}
```

The first one, *showPicks*, changes a DIV section in the HTML page by including the identity (You or Computer) of the selector and the image file corresponding to the selection. Note the parameters *whoPickedId* and *imageFile*. The first parm is the value of the ID attribute in the HTML that will be updated; the second is the image file used for display.

This is the *table row* that *showPicks* updates:

```
<tr><td><div id="You"></div></td><td><div id='Computer'></div></td></tr>
```

Note the ID attribute has 'convenient' values.

The *showPicks* function is called *twice.*

The other function *showStatusAndMonies* is similar to *showPicks* inasmuch as both use the Document Object Model to locate HTML tags and update their values. Again, note the ID attribute used in the call to *getDocumentById* is the same as the value coded in the HTML page for the *status and monies*.

On to the provided starter file:

The starter file *rps.html*

At the top of the file, you'll find the variables corresponding to the *images and monies:*

```
var rockImage     = "rock.jpg" ;
var paperImage    = "paper.jpg" ;
var scissorsImage = "scissors.jpg" ;

var yourLoot     = 100 ;
var computerLoot = 100 ;
```

Use the image files to identify what was picked (by you and the computer)

There are function stubs for several needed routines. The stub for the function that makes the computer selection is:

```
function getComputerPick( ) {
      var pickNum = Math.floor( Math.random( ) * 10000 ) % 3  ;

}
```

This code generates a number between 0 and 2, corresponding to the rock, etc. The function should **return the image** corresponding to the random number generated.

This function determines if you lost. You could code a function that determines if the computer loses but you need only one; best to stick with what is stubbed out already.

```
function youLost( yourPick, computerPick ) {

}
```

All you need to make this determination are the images corresponding to the selections. This function should **return true if you lost**.

The next function creates the *status string* and *adjusts the monies*.

```
function adjustMoniesGetStatus( myPick, computerPick ) {

}
```

Again, all we need are the images corresponding to the picks. This function needs to call *youLost* to determine who won, lost or tied to create the status string and adjust the $$$.

Here's the *play* function called when you *click an image*.

```
function play( yourPickImage ) {
      // Display your pick - this is a gimme using the function in miscfuncs.js
      showPicks( 'You', yourPickImage ) ;
      // Retrieve the image corresponding to the computer pick.
      // Use the function you coded for this purpose
      var computerPickImage = ???????????????????
      // Display the computer's pick again using the provided function in miscfuncs.js
      showPicks( 'Computer', computerPickImage ) ;
      // Use the function you coded above to retrieve who won or lost the round and to
      // adjust your and computer's monies
      var status = ??????????????
      // Change data in table to show who won and current $$$ using function
      // in miscfuncs.js
      showStatusAndMonies( status, yourLoot, computerLoot ) ;
}
```

Notice in *showPicks* we **pass the ID of the tag** we will change using the provided function in *miscfuncs.js*.

Finally, let's look at the HTML:

```
<input type="image"  onclick='play( rockImage );' src="rock.jpg" >
<input type="image"  onclick='play( paperImage );' src="paper.jpg" >
<input type="image"  onclick='play( scissorsImage );' src="scissors.jpg" >
<p>
<table>
      <tr><td><div id="You"></div></td><td><div id='Computer'></div></td></tr>
      <tr><td colspan="2"><div id="status"></div></td></tr>
      <tr><td><div id="yourMoney"></div></td><td><div id="computerMoney"></div></td></tr>
</table>
```

Not gonna bother with the title, body, paragraph tags...

Just want to point out the *event handler* executed when you click a button and the *Ids of the tags* used by JavaScript to locate and change the page content.

We'll cover *event handlers* later in the course; using the DOM to change page content, well, not so much.