# The University of the West Indies
## Cave Hill Campus
## **Faculty of Science and Technology** Department of Computer Science, Mathematics and Physics SWEN1005:
Mobile Web Programming
Laboratory Exercise Six
Using HTML5, CSS and JavaScript

---

**INSTRUCTIONS**

1. This week we will use **HTML5, CSS3 and JavaScript** to perform a number of tasks for assessment.

2. There are **three questions**, you should attempt each on a separate webpage.

3. **Follow the instructions as carefully as possible**.

4. Please **include all the necessary files** for each webpage in **separate folders** and make sure all your links work from that folder.

1. Using JavaScript, write code that tests to see whether geolocation is available for the browser being used:
    a. If geolocation is available, retrieve the user's current location and display it to the web page.
    b. If geolocation is not available, display a statement that lets the user know his or her browser does not support geolocation.

2. Use the element IDs provided below and the getElementById function to retrieve the geolocation values required for the table to display the listed values then convert the table to mobile responsive page using **GridView** and **FlexBox** CSS modules to create a mobile responsive layout.

   Create a simple interface for your page, which tells the user about the Geolocation element and the various properties being displayed and their meanings. Use styles to make the page look aesthetically pleasing.

```
<table>
  <tr>
    <td>accuracy:</td>
    <td><span id="accuracyValue"></span></td>
  </tr>
  <tr>
    <td>altitude:</td>
    <td><span id="altitudeValue"></span></td>
  </tr>
  <tr>
    <td>altitudeAccuracy:</td>
    <td><span id="altitudeAccuracyValue"></span></td>
  </tr>
  <tr>
    <td>heading:</td>
    <td><span id="headingValue"></span></td>
  </tr>
  <tr>
    <td>latitude:</td>
    <td><span id="latitudeValue"></span></td>
  </tr>
  <tr>
    <td>longitude:</td>
    <td><span id="longitudeValue"></span></td>
  </tr>
  <tr>
    <td>speed:</td>
    <td><span id="speedValue"></span></td>
  </tr>
</table>
```

3. Two players (player X, and player O) play on a 3x3 grid. A player can put his/her letter (either X or O) in an empty cell in the grid. If a player forms a row, a column or a diagonal with his/her letter, that player wins and the game ends. If the grid is full and there is no row, column or diagonal of the same letter, the game ends at draw. A player should try to win in the lowest possible number of moves.

Using the three files located in the “TicTacToe” folder. Render the game and play it a few times to get a feel for the game. You are required to use the **sessionStorage** attribute to create a session scorecard to be displayed on the game board. For each round, the scorecard should display the:

- date and time
- names of the two players
- number of games played
- number of games won by each player
- number of games drawn
- the number of moves made by the winner

At the end of each round the round-winner’s name should be displayed and the players should be asked if they want to play again or if they wish to quit.

You are also required to use the **localStorage** attribute to maintain a record of:

- player’s names
    - number of games won
    - number of rounds won
    - number of games lost
    - number of rounds lost
- total number of games played
- total number of rounds played
- a congratulatory message to the person with the most wins

This data will only be displayed on the gameboard **if requested by the user**. You may modify the appearance of the game to fit your content or to make it more aesthetically pleasing.

**Please note this game was created as a part of The Odin Project. Be mindful that we are only using it as a learning tool and credit must be given and permission must be sought if you want to use the code for other purpose. Visit https://www.theodinproject.com/courses/javascript/lessons/tic-tac-toe-javascript to see the original project requirements.**