Unit 2: Application of OOP (I): AJAX

Practice 4: Creating the AJAX-enabled Guessing the Number Game

Three files are involved in the application. The *index.php* presents the user interface. The Web page responds to the **click** event on the **Guess** button and the **keyup** event. The *guess.js* file contains all the JavaScript functions to handle client-side scripting including the AJAX request. The *guess.php* file handles server-side tasks including generating a random number, evaluating the player's guess, and sending response via an XML document.

Get Ready

1. Download the data file *guessingnum.zip* from Canvas and extract the **guessingnum** folder into the **htdocs/I211/Unit2** folder.
2. Review the code in PhpStorm and preview the application by running the *index.php* file.

Part 1: Modify the *index.php*

3. Switch to the *index.php* file in PhpStorm. Right above the closing </head> tag, add the following line of code to load an external file called *guess.js*:

```
<script src="guess.js"></script>
```

4. Toward the end of the file, locate the <button>Guess</button> for the **Guess** button. Inside the <button> tag, add the **onclick** event handler. The **play**() function should be called.

```
<button onclick="play()">Guess</button>
```

5. Locate the html comments that read “div block to display score”. Below the comments, add a div block and set its id to **score.**

<div id="score">100</div>

6. Below the html comments that read “div block to display message”, create a div block and set its id to **divMessage**.

```
<div id="divMessage"></div>
```

Part 2: Modify the *guess.js* file

6. Complete the anonymous function called by **window's onload** event handler.

```
//DOM objects
messageObj = document.getElementById("divMessage");
scoreObj = document.getElementById("score");
guessObj = document.getElementById("guess");
```

Part 3: Process AJAX requests

7. Inside the **play()** function, locate the JavaScript comments that read “complete the AJAX code below”. Perform the following tasks.
   a. Create an XHR object

   ```
   xmlHttp = new XMLHttpRequest();
   ```

   b. Define an asynchronous AJAX request.

   ```
   xmlHttp.open("GET", "guess.php?guess=" + guess, true);
   ```

   c. Handle server’s response.

   ```
   // handle server’s responses with an anonymous function
   xmlHttp.onload = function () {
   };
   ```

   d. Send the request.

   ```
   //make the request to the server
   xmlHttp.send(null);
   ```

Part 4: Create the *guess.php* file

8. Create a new PHP file and save it as *guess.php*.
9. Create a new cookie for a random number if it does not exist already; If a cookie already exists, just retrieve it.

   ```
   if (isset($_COOKIE['random']))
       $random = $_COOKIE['random'];
   else {
       $random = rand(1, 100);
       setcookie("random", $random);
   }
   ```

10. Retrieve player’s guess from a querystring variable named **guess**.

   ```
   //retrieve player’s guess
   $guess = (int)($_GET['guess']);
   ```

11. Evaluate the player’s guess and then output a JSON object containing value 1, 0, or -1.

   ```
   if ($guess > $random) {
       echo json_encode(array("result" => 1));
   } else if ($guess < $random) {
       echo json_encode(array("result" => -1));
   } else {
       echo json_encode(array("result" => 0));
   }
   ```

12. Save the file. Now the application is ready to run.

Part 5: handle server's responses

13. Locate the anonymous function that handles the server's responses in guess.js and type the following code.

```
xmlHttp.onload = function () {
    // retrieve the json document sent from the server
    var resultJSON = JSON.parse(xmlHttp.responseText);
    var result = resultJSON.result;

    // display the response received from the server
    if (result === 1) {
         messageObj.innerHTML = guess + " was too high. Try again";
    } else if (result === -1) {
         messageObj.innerHTML = guess + " was too low. Try again";
    } else if (result === 0) {
         messageObj.innerHTML = "Congratulation! You've guessed the secret number.";
         is_game_over = true;
    }
};
```

14. Save all files and run *index.php*.