Login Assignment

Assignment 1

Make an array of random numbers that do not repeat, then make another array with those numbers sorted. Show both of them on the page.

This assignment makes use of one for loop and one function that includes a for loop.

```JavaScript
for(let i = 0; i < 10; i++) {

  let randInt = Math.floor(Math.random() * 101);

  if (randArray.includes(randInt)) {
    i--;
  }
  else {
    randArray.push(randInt);
  }
}
```

This for loop generates a random number from the Math.random() method. Math.random() generates a random number from 0 - 1 therefore it is multiplied by 101 to output a number between 0 - 101 and edited by the Math.floor() method to output round numbers. The reason it's 101 and not 100 is because the highest number it can generate is about 100.99999999 which after going through the Math.floor() method will give out a maximum number 100.

```JavaScript
function normalSort(array, maxNumber) {
  let newArray = []
  let c = 0;
  for(let i = 0; i < maxNumber + 1; i++) {
    if (array.includes(i)) {
      newArray[c] = i;
      c++;
    }
  }
  return newArray;
}
```

*array.*sort() method cannot be used in this assignment because it will sort numbers based on only the first number (ex. {1, 14, 2, 25, 3}) therefore this function was made. The function takes two arguments: an array that will be sorted and the maximum number that can be generated (here it will be 100). The function makes use of 2 variables, newArray which is an array that the numbers will be sorted into and c which is a counter of index to that array. The for loop will go through every number from 0 to the number given as the max number and check if it's in the array that is being sorted. If it is in the array the number will be appended to the new array as the current index after which the index will increase. Because the function goes from 0 to the max number it will append lower numbers as earlier indexes in the new array. After everything is done the function returns that new array.

This is the line of code that takes the two arrays and puts them into a <p> element with "RandomArrays" id.

```javascript
document.getElementById("RandomArrays").innerHTML = `${randArray} <br>
${normalSort(randArray, 100)}`;
```

Assignment 2

Make a registration window using JavaScript. The window should include a Username, an Email that has "@" in the middle, a Password with at least 6 signs, Repeat Password which should have the same input as Password and a Terms of Service checkbox that must be checked before proceeding to a new page that thanks the user for registration.

This assignment makes use of 3 functions that check if a given input is correct and 1 function that checks if every other function outputs the value of "true" to then redirect users to a different page.

The checkUsername function takes an id of an element as an argument and returns true if that box isn't empty. If it is empty the border color of that input element will turn red and an alert will pop out informing the user that they need to input their username, then returns false.

```javascript
function checkUsername(id){
  if(id.value == "") {
    alert("Username is required");
    id.style.borderColor = "red";
    return false;
  }
  else {
    id.style.borderColor = "black";
    return true;
  }}
```

The checkEmail function has three different conditions for returning true: the email must include an “@” sign, the “@” sign isn’t the first symbol and the “@” isn't the last symbol. If at least one of these conditions is false it will send an alert of what’s wrong, turn the input box red and return false.

```javascript
function checkEmail(id){
  if(id.value.includes("@") && id.value[0] != "@" && id.value[id.value.length - 1] != "@") {
    id.style.borderColor = "black";
    return true;
  }
  else {
    id.style.borderColor = "red";
    alert("E-mail must include an \"@\" sign and must start and end with a letter");
    return false;
  }
}
```

The checkPassword function takes two arguments: id of the first password and id of the repeated password. It returns false if either the first password is less than 6 symbols or if the repeated password doesn’t match the first one. If either of those outcomes happen the function will turn both inputs red, notify the user what’s wrong via alert and return false.

```javascript
function checkPassword(id1, id2) {
  if(id1.value.length < 6 || id1.value != id2.value) {
    id1.style.borderColor = "red";
    id2.style.borderColor = "red";
    alert("Password denied. Make sure its at least 6 characters long and that both passwords are the same.");
    return false;
    }
    else {
      id1.style.borderColor = "black";
      id2.style.borderColor = "black";
      return true;
    }
}
```

This function uses every other function to check if they output true and if the Terms of Service button also outputs true. If everything is correct then it redirects the user to a page that thanks them for registering by making use of window.location.href = “link”.

```JavaScript
function Validate() {

  var username = document.getElementById("username");
  var email = document.getElementById("email");
  var password = document.getElementById("password");
  var repeatPassword = document.getElementById("password2");
  var termsChecked = document.getElementById("terms").checked

  var userIsCorrect = checkUsername(username);
  var emailIsCorrect = checkEmail(email);
  var passIsCorrect = checkPassword(password, repeatPassword);

  if(termsChecked != true) {
    alert("Accept Terms of Service & Conditions before continuing")
  }
  else if( userIsCorrect == true && emailIsCorrect == true && passIsCorrect == true
&& termsChecked == true) {
    window.location.href = "./html/registration.html"
  }
}
```

Assignment 3

Repeat assignment 2 but only use html. Repeated password is not mandatory.

```Unset
<form class="htmllogin" action="./html/registration.html">
    <h3>Login HTML</h3>
    <input type="text" placeholder="Username" required>
    <input type="email" placeholder="E-mail" required>
    <input type="text" placeholder="Password" minlength="6" required>
    <div class="terms_conditions">
      <input type="checkbox" id="termshtml" required>
      <label for="termshtml">I have read and accepted Terms and
Conditions.</label>
    </div>
    <input type="submit" value="Submit">
  </form>
```

This assignment makes use of the <form> tag in html. Every input inside has the “required” attribute so that the user has to put something inside. The email input is of type=”email” so the page knows that it should have an “@” in the middle, the password input has the “minlength” attribute to make the password have at least 6 characters. The only thing that checkbox needs to have is the “required” attribute. After the type=”submit” input is clicked the action described in the “action” form attribute will be executed which in this case is redirecting the user to a different page.