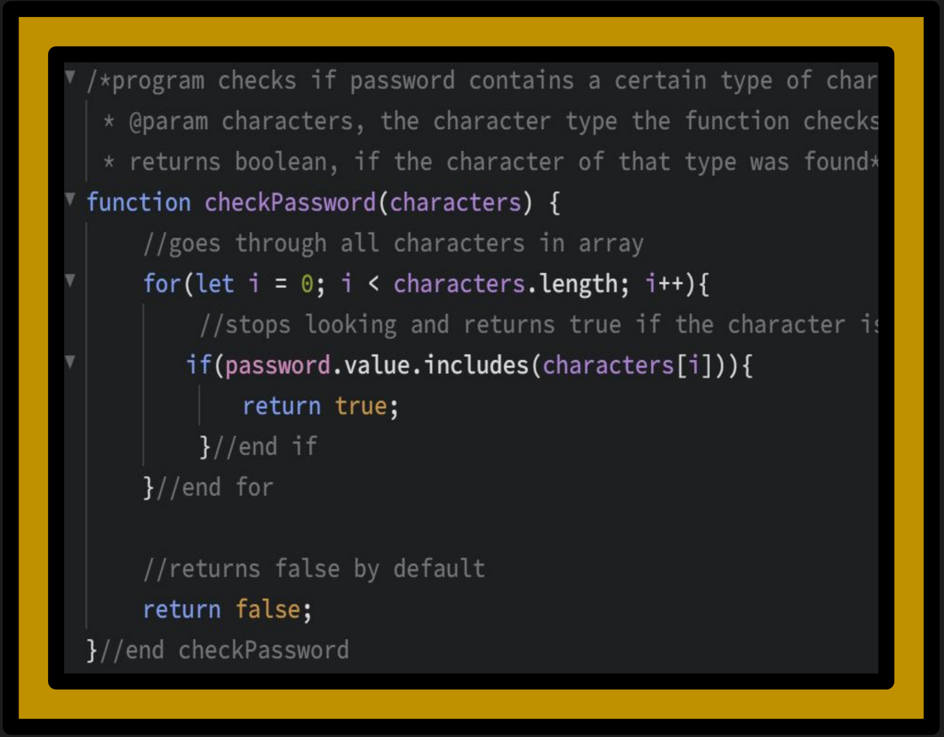
Project 2: Password Assistant

By: *Jesus De Leon*

Github Link: https://github.com/JDeLeon05/JDeLeon05-Project2.git
Website Link: https://jdeleon05.github.io/JDeLeon05-Project2/

checkPassword()

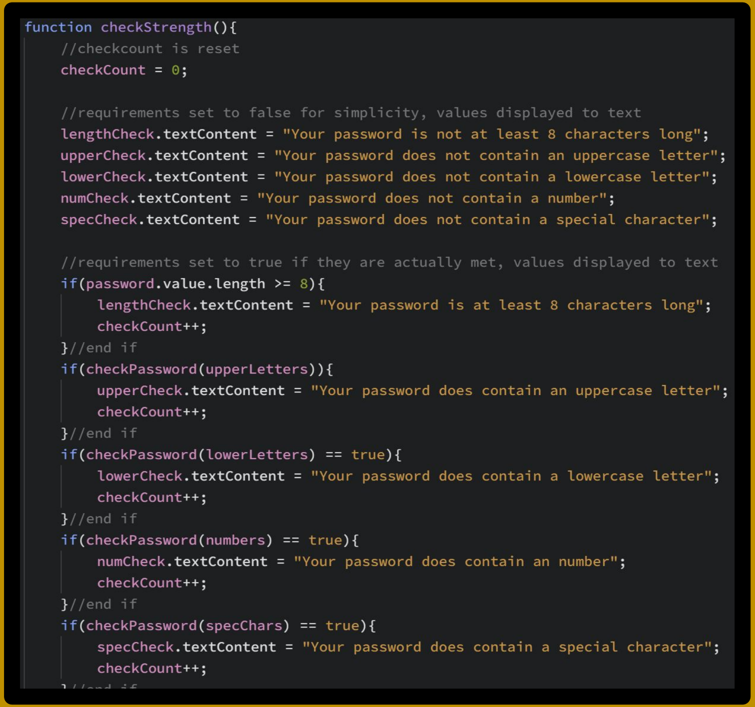


```
/*program checks if password contains a certain type of char
 * @param characters, the character type the function checks
 * returns boolean, if the character of that type was found*
function checkPassword(characters) {
    //goes through all characters in array
    for(let i = 0; i < characters.length; i++){
        //stops looking and returns true if the character is
       if(password.value.includes(characters[i])){
           return true;
        }//end if
    }//end for

    //returns false by default
    return false;
}//end checkPassword
```

Explanation:

- Goes through elements/characters of an array
- checks if the password has element until it finds one or it goes through all elements.

checkStrength(): Part 1

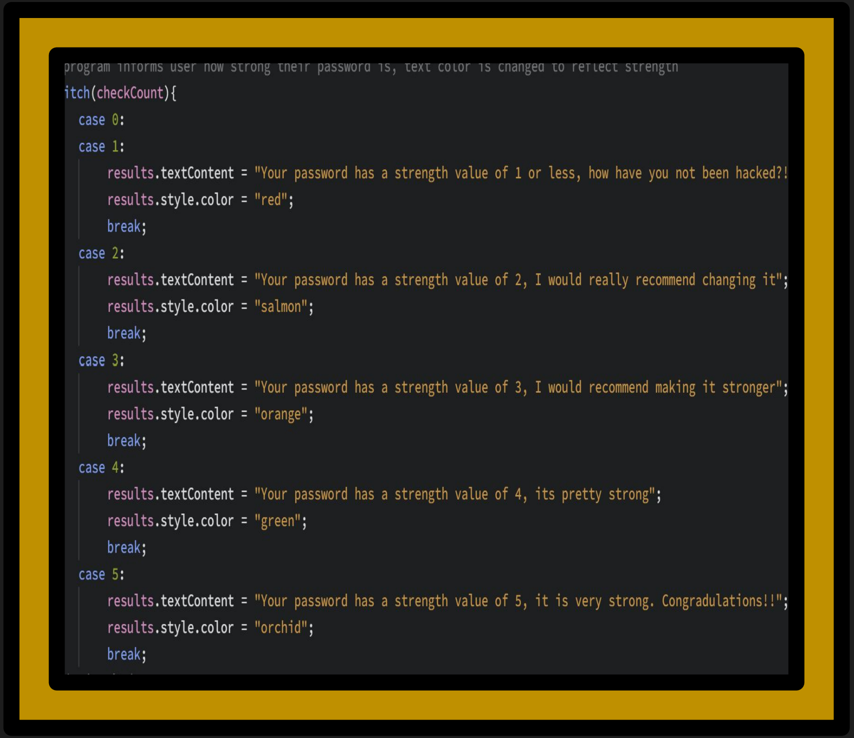


```
function checkStrength(){
    //checkcount is reset
    checkCount = 0;

    //requirements set to false for simplicity, values displayed to text
    lengthCheck.textContent = "Your password is not at least 8 characters long";
    upperCheck.textContent = "Your password does not contain an uppercase letter";
    lowerCheck.textContent = "Your password does not contain a lowercase letter";
    numCheck.textContent = "Your password does not contain a number";
    specCheck.textContent = "Your password does not contain a special character";

    //requirements set to true if they are actually met, values displayed to text
    if(password.value.length >= 8){
        lengthCheck.textContent = "Your password is at least 8 characters long";
        checkCount++;
    }//end if
    if(checkPassword(upperLetters)){
        upperCheck.textContent = "Your password does contain an uppercase letter";
        checkCount++;
    }//end if
    if(checkPassword(lowerLetters) == true){
        lowerCheck.textContent = "Your password does contain a lowercase letter";
        checkCount++;
    }//end if
    if(checkPassword(numbers) == true){
        numCheck.textContent = "Your password does contain an number";
        checkCount++;
    }//end if
    if(checkPassword(specChars) == true){
        specCheck.textContent = "Your password does contain a special character";
        checkCount++;
    }//end if
```

Explanation:

- checkCount represent strength of password from 0-5
- Uses variables that represent elements in html file
- Sets requirements to “not met” by default to avoid if-else statements
- Checks 5 requirements:
    - Changes them to “met” if the password meets them and increments checkCount

checkStrength(): Part 2

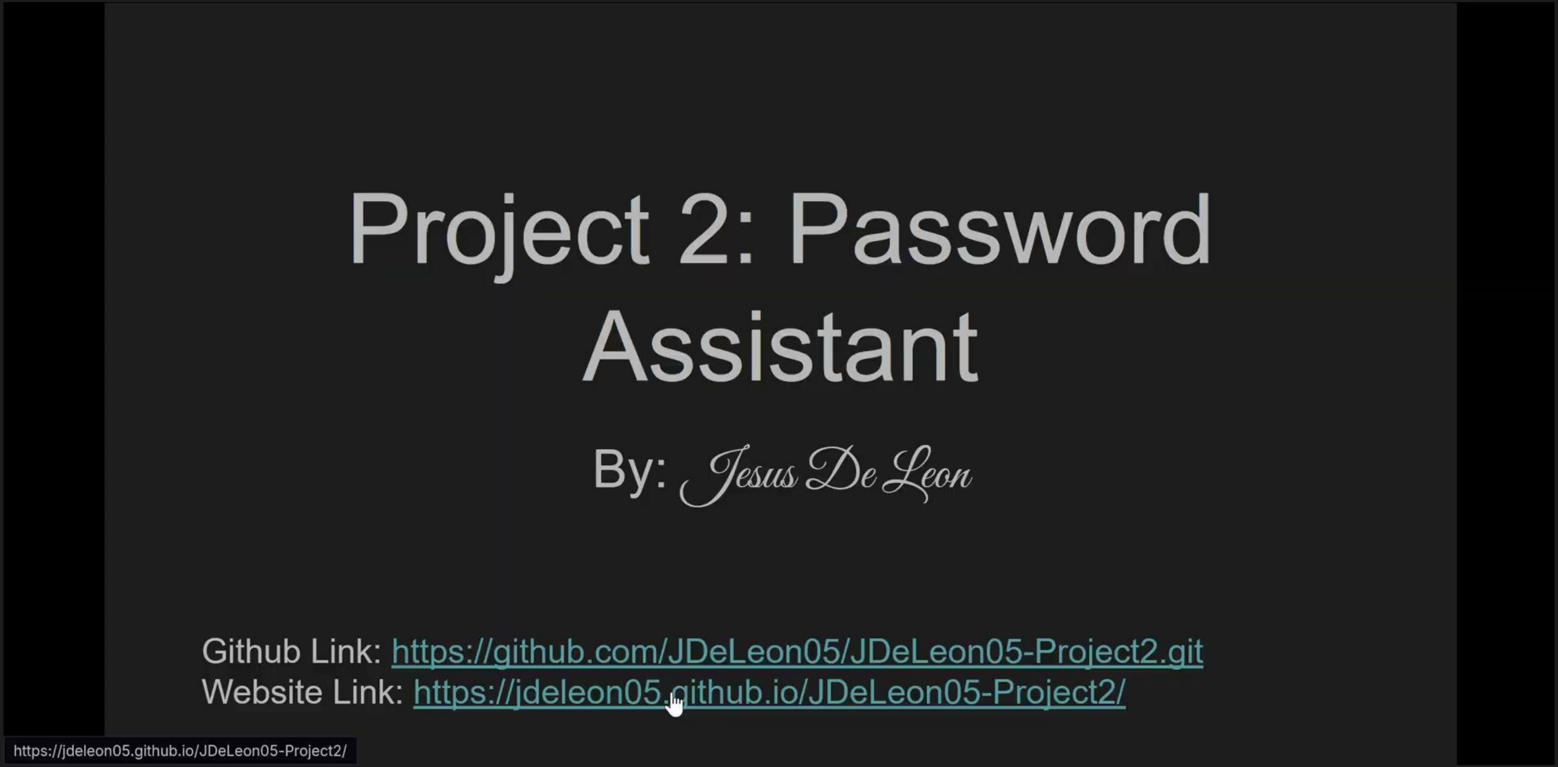


```
program informs user how strong their password is, text color is changed to reflect strength
itch(checkCount){
  case 0:
  case 1:
      results.textContent = "Your password has a strength value of 1 or less, how have you not been hacked?!
      results.style.color = "red";
      break;
  case 2:
      results.textContent = "Your password has a strength value of 2, I would really recommend changing it";
      results.style.color = "salmon";
      break;
  case 3:
      results.textContent = "Your password has a strength value of 3, I would recommend making it stronger";
      results.style.color = "orange";
      break;
  case 4:
      results.textContent = "Your password has a strength value of 4, its pretty strong";
      results.style.color = "green";
      break;
  case 5:
      results.textContent = "Your password has a strength value of 5, it is very strong. Congradulations!!";
      results.style.color = "orchid";
      break;
```

Explanation:

- Message transmitted to <p> tag, telling the user how strong their password is
- Score based on requirements met
- Message font color is based on score

Video:



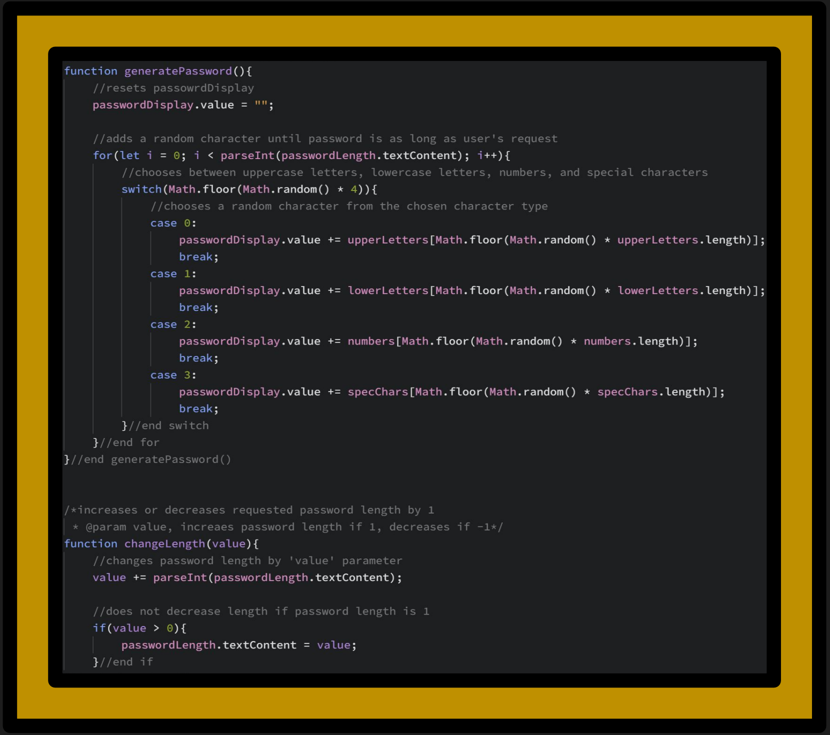
Project 2: Password Assistant

By: Jesus De Leon

Github Link: https://github.com/JDeLeon05/JDeLeon05-Project2.git
Website Link: https://jdeleon05.github.io/JDeLeon05-Project2/

https://jdeleon05.github.io/JDeLeon05-Project2/

generatePassword() & changeLength()



```
function generatePassword(){
    //resets passowrdDisplay
    passwordDisplay.value = "";

    //adds a random character until password is as long as user's request
    for(let i = 0; i < parseInt(passwordLength.textContent); i++){
        //chooses between uppercase letters, lowercase letters, numbers, and special characters
        switch(Math.floor(Math.random() * 4)){
            //chooses a random character from the chosen character type
            case 0:
                passwordDisplay.value += upperLetters[Math.floor(Math.random() * upperLetters.length)];
                break;
            case 1:
                passwordDisplay.value += lowerLetters[Math.floor(Math.random() * lowerLetters.length)];
                break;
            case 2:
                passwordDisplay.value += numbers[Math.floor(Math.random() * numbers.length)];
                break;
            case 3:
                passwordDisplay.value += specChars[Math.floor(Math.random() * specChars.length)];
                break;
        }//end switch
    }//end for
}//end generatePassword()


/*increases or decreases requested password length by 1
 * @param value, increaes password length if 1, decreases if -1*/
function changeLength(value){
    //changes password length by 'value' parameter
    value += parseInt(passwordLength.textContent);

    //does not decrease length if password length is 1
    if(value > 0){
        passwordLength.textContent = value;
    }//end if
```

Explanation:

- passwordDisplay represents <textarea> for generated password
- Random character picked from 4 different character types
- Repeats until password is requested length

- Value set to 1 if “^” button is pressed
- Value set to -1 if “v” button is pressed
- Current password length added to value
- Password length is updated only if value > 0

Video