Luke Talcott – ldt9gn@umsystem.edu

GitHub link:
https://github.com/LukeT11/WebMobileProgramming/tree/main/Web_Lessons/Web_ICP4

Web ICP 4

Object-Oriented JavaScript, jQuery with API’s

Introduction

Object-Oriented JavaScript uses all objects with values with properties to be used as object methods. These methods are actions that can be used on objects.

jQuery is a JavaScript Library that makes it easier to use with simple API’s with things like HTML documentation traversal and manipulation, event handling, Ajax, etc.

API’s is an Application Programming Interface is a type of software interface that services as a connection between other devices and software that allows a bridge for easy access and transfer of information.

Tasks

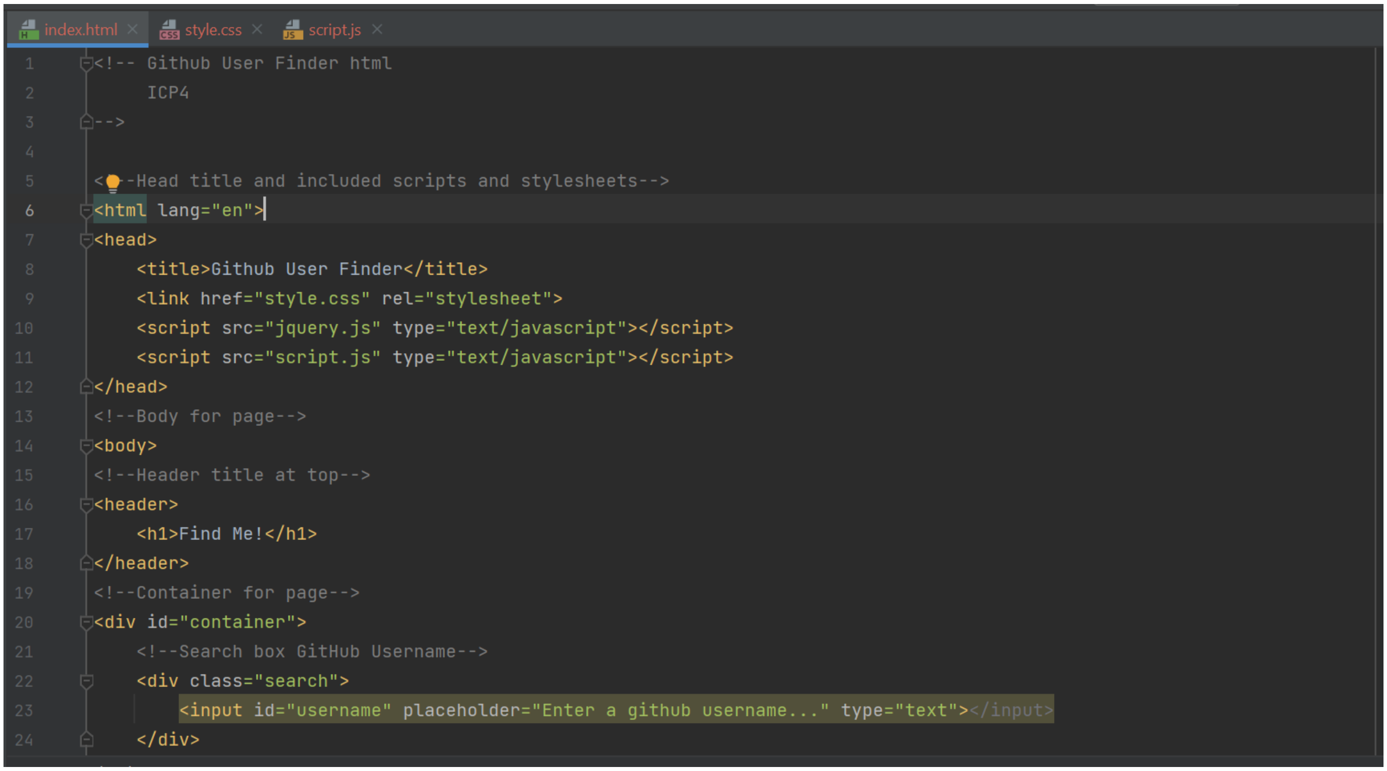
ICP4 tasks, description and example uses.

ICP3 Tasks (jQuery, JavaScript (OO), API):

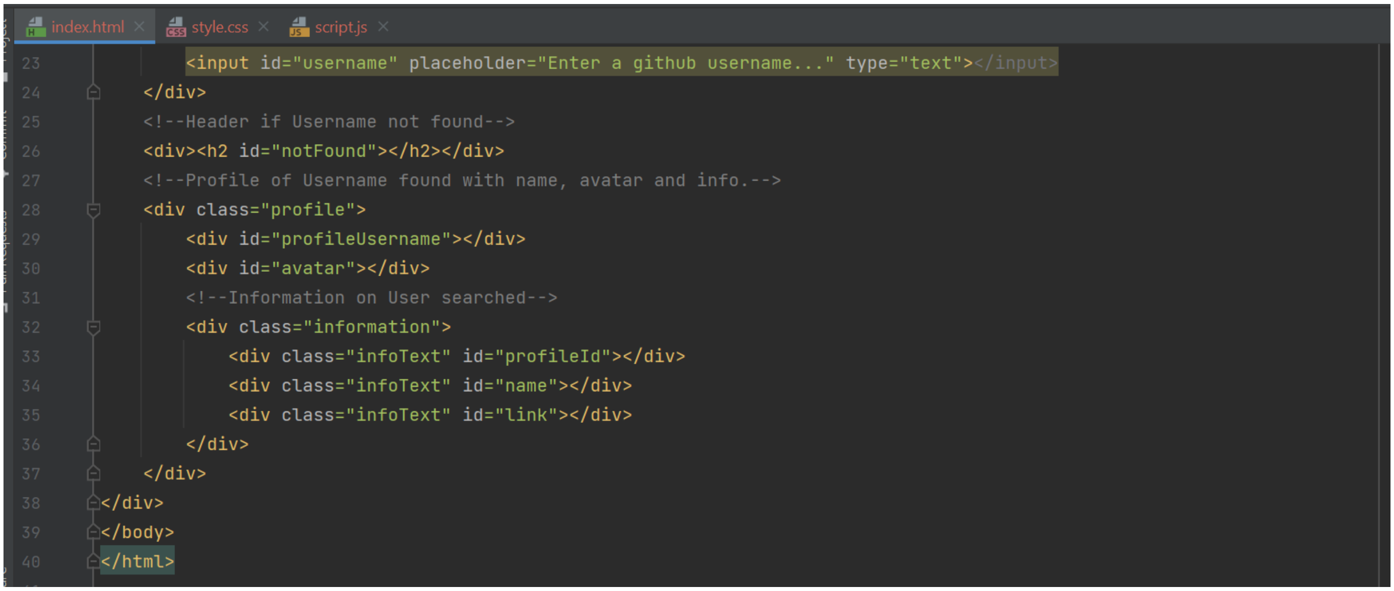
- GitHub User Finder
  - Create an App that calls the GitHub API
  - Searched by Username
  - Display’s the following user’s accounts details:
    - Name of the user
    - Id of the user
    - Profile Picture
    - Link to the user’s account

GitHub User Finder

- HTML

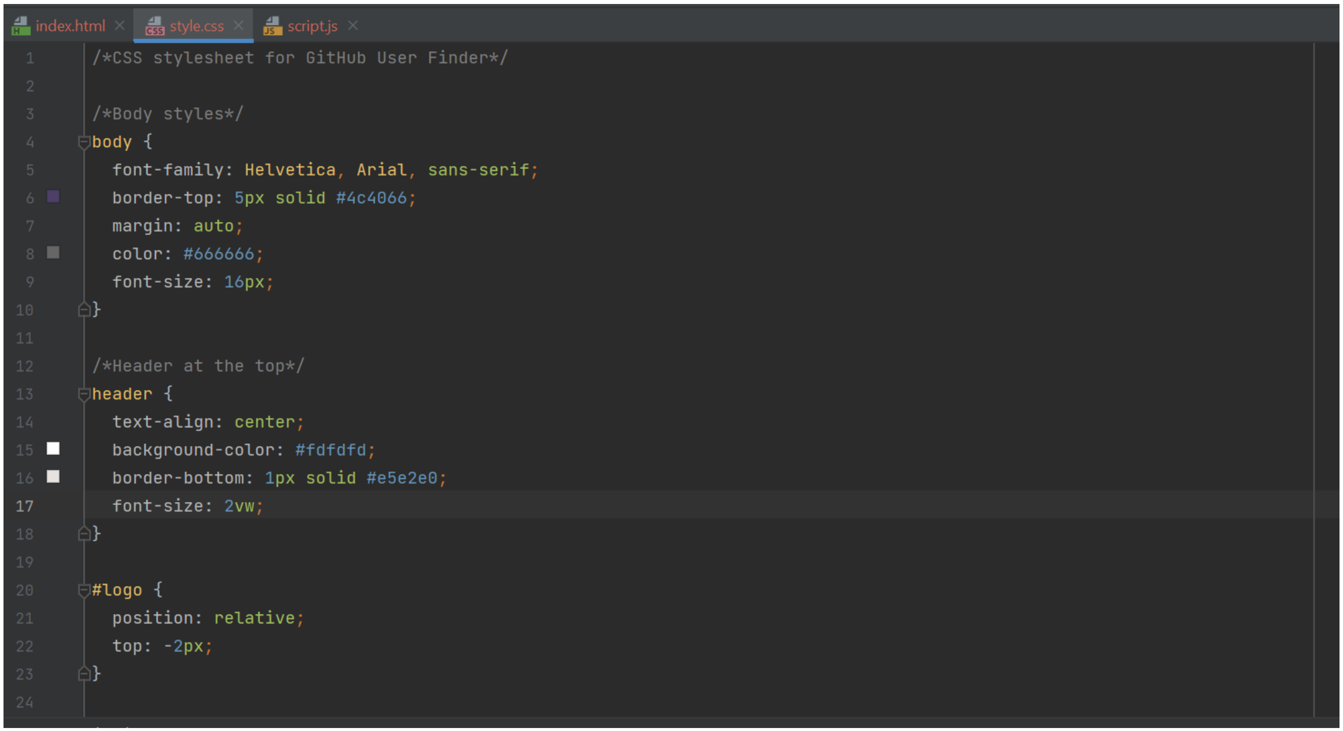


```html
<!-- Github User Finder html
     ICP4
-->

<!--Head title and included scripts and stylesheets-->
<html lang="en">
<head>
    <title>Github User Finder</title>
    <link href="style.css" rel="stylesheet">
    <script src="jquery.js" type="text/javascript"></script>
    <script src="script.js" type="text/javascript"></script>
</head>
<!--Body for page-->
<body>
<!--Header title at top-->
<header>
    <h1>Find Me!</h1>
</header>
<!--Container for page-->
<div id="container">
    <!--Search box GitHub Username-->
    <div class="search">
        <input id="username" placeholder="Enter a github username..." type="text"></input>
    </div>
```

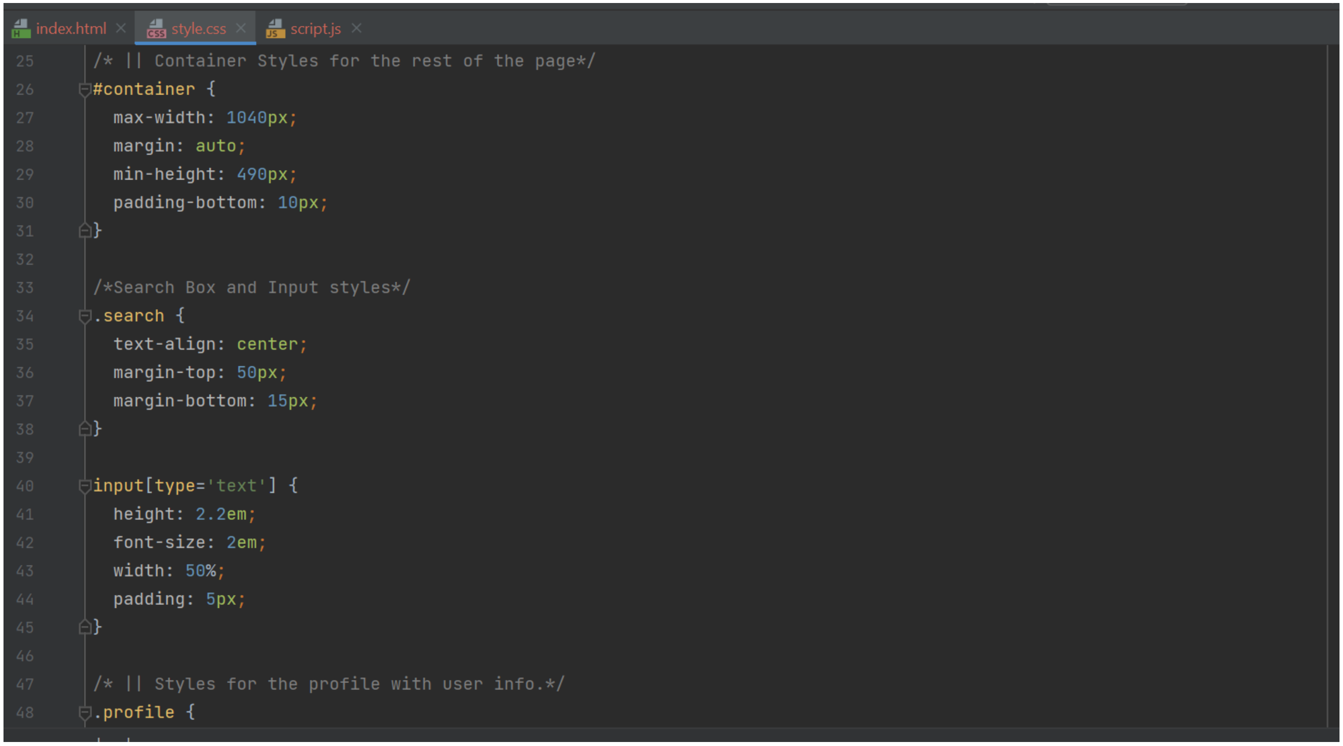


```html
        <input id="username" placeholder="Enter a github username..." type="text"></input>
    </div>
    <!--Header if Username not found-->
    <div><h2 id="notFound"></h2></div>
    <!--Profile of Username found with name, avatar and info.-->
    <div class="profile">
        <div id="profileUsername"></div>
        <div id="avatar"></div>
        <!--Information on User searched-->
        <div class="information">
            <div class="infoText" id="profileId"></div>
            <div class="infoText" id="name"></div>
            <div class="infoText" id="link"></div>
        </div>
    </div>
</div>
</body>
</html>
```

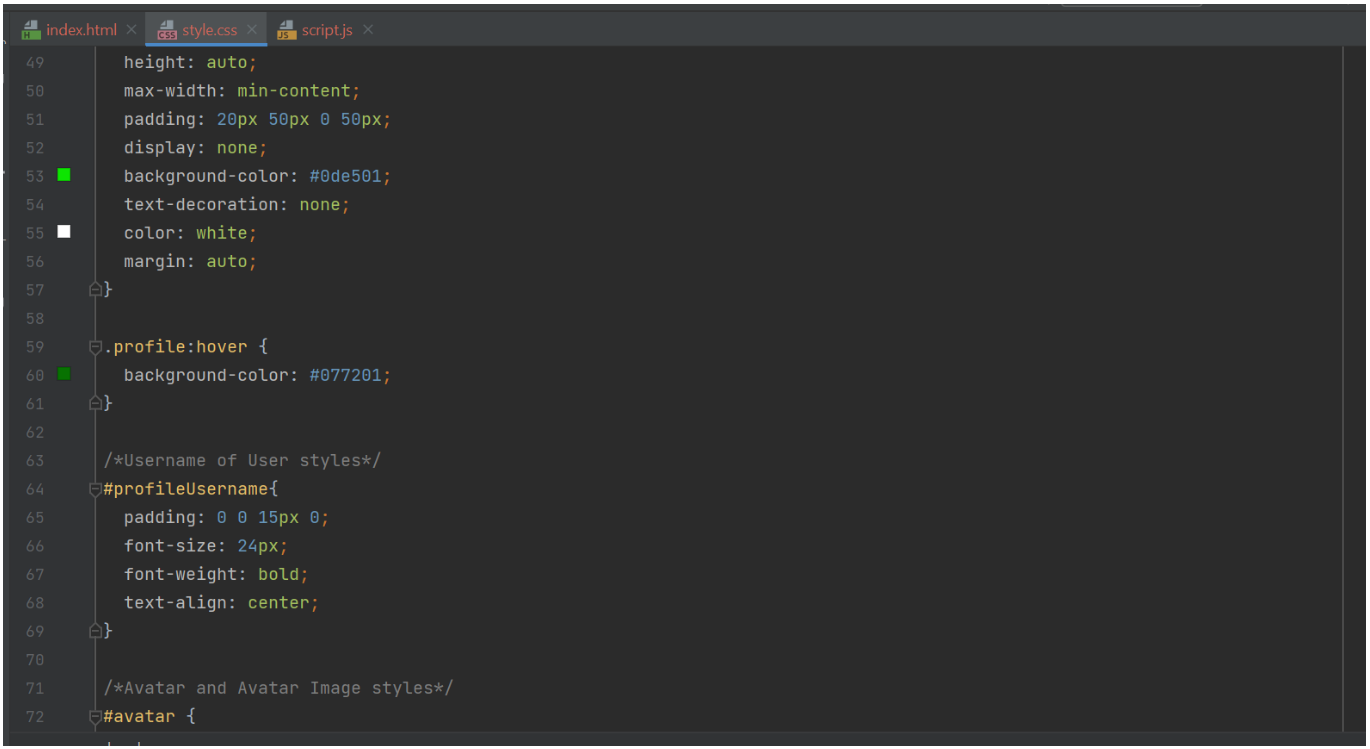
- CSS



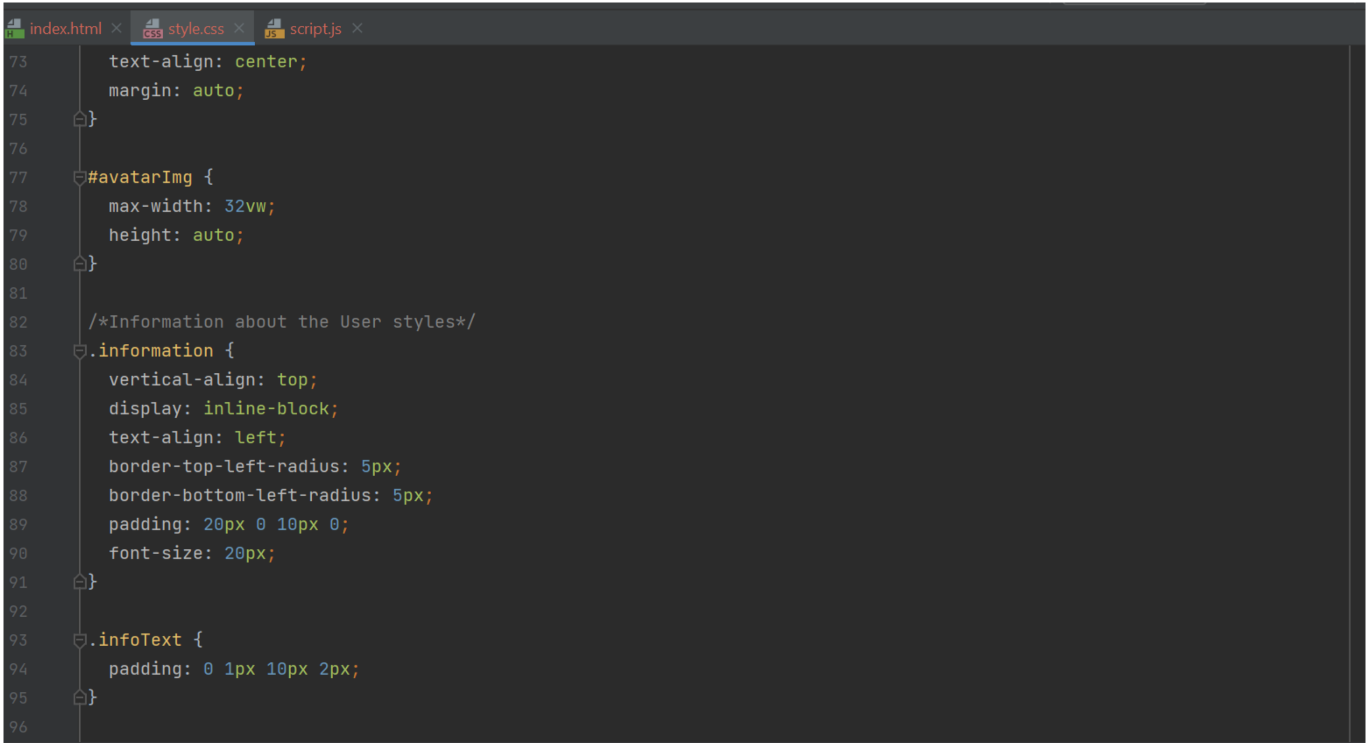
```css
/*CSS stylesheet for GitHub User Finder*/

/*Body styles*/
body {
  font-family: Helvetica, Arial, sans-serif;
  border-top: 5px solid #4c4066;
  margin: auto;
  color: #666666;
  font-size: 16px;
}

/*Header at the top*/
header {
  text-align: center;
  background-color: #fdfdfd;
  border-bottom: 1px solid #e5e2e0;
  font-size: 2vw;
}

#logo {
  position: relative;
  top: -2px;
}

```



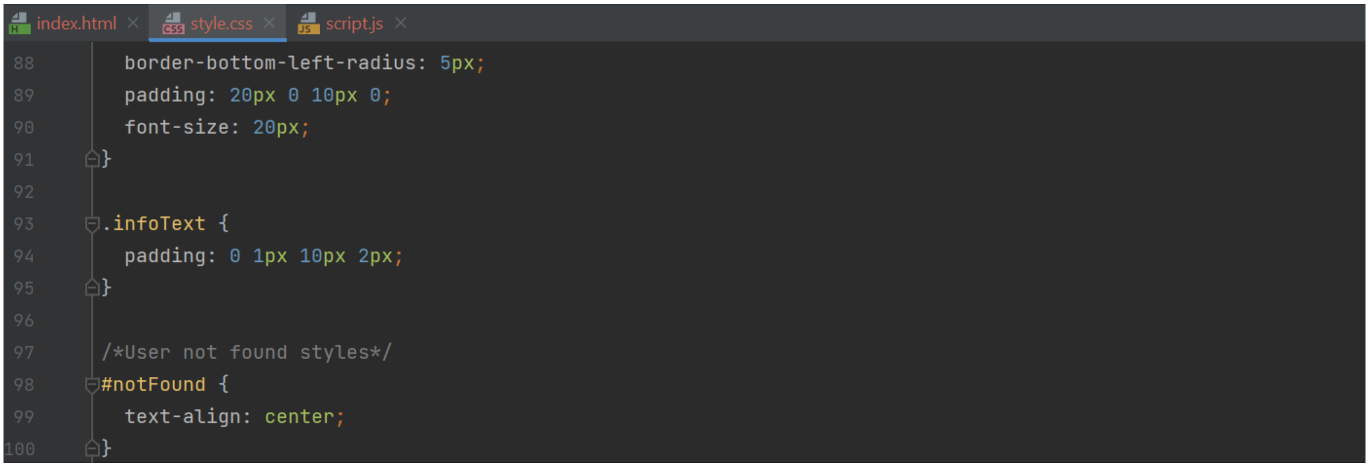
```css
/* || Container Styles for the rest of the page*/
#container {
  max-width: 1040px;
  margin: auto;
  min-height: 490px;
  padding-bottom: 10px;
}

/*Search Box and Input styles*/
.search {
  text-align: center;
  margin-top: 50px;
  margin-bottom: 15px;
}

input[type='text'] {
  height: 2.2em;
  font-size: 2em;
  width: 50%;
  padding: 5px;
}

/* || Styles for the profile with user info.*/
.profile {
```



```css
  height: auto;
  max-width: min-content;
  padding: 20px 50px 0 50px;
  display: none;
  background-color: #0de501;
  text-decoration: none;
  color: white;
  margin: auto;
}

.profile:hover {
  background-color: #077201;
}

/*Username of User styles*/
#profileUsername{
  padding: 0 0 15px 0;
  font-size: 24px;
  font-weight: bold;
  text-align: center;
}

/*Avatar and Avatar Image styles*/
#avatar {
```

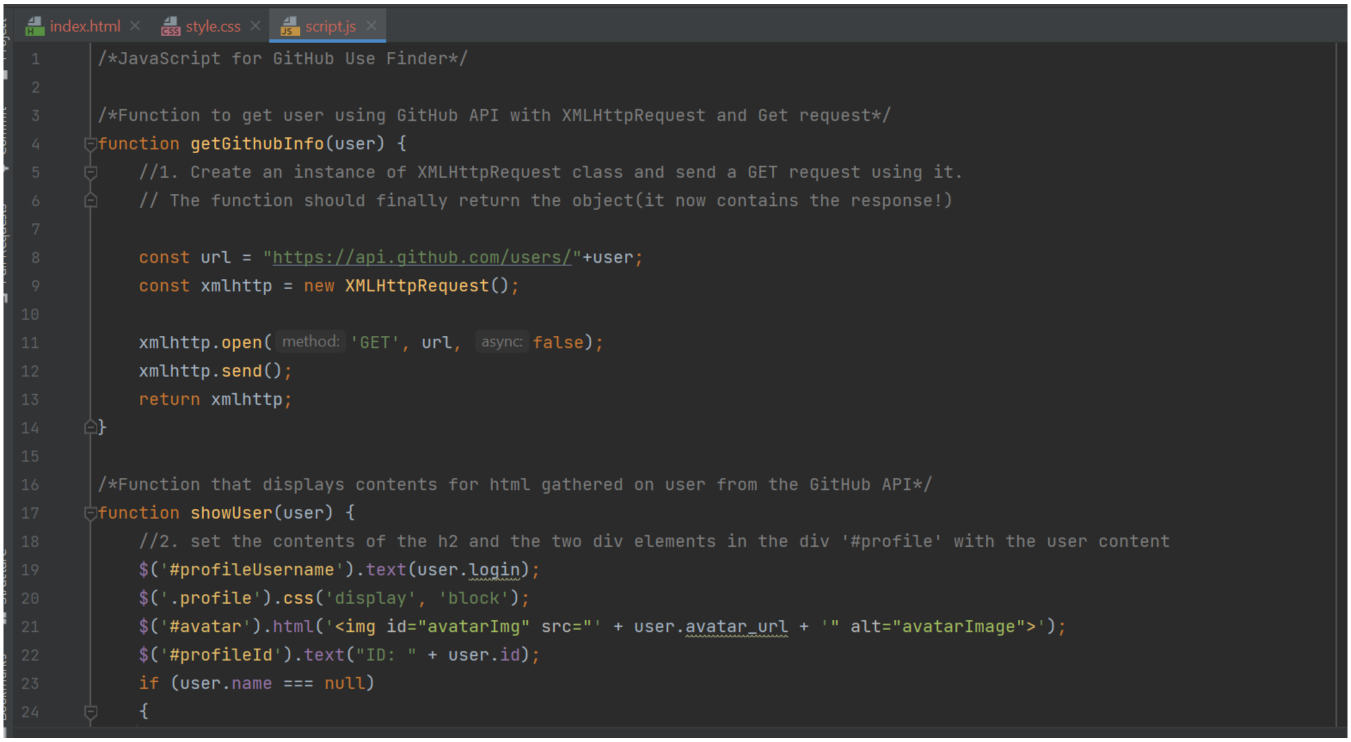


```css
  text-align: center;
  margin: auto;
}

#avatarImg {
  max-width: 32vw;
  height: auto;
}

/*Information about the User styles*/
.information {
  vertical-align: top;
  display: inline-block;
  text-align: left;
  border-top-left-radius: 5px;
  border-bottom-left-radius: 5px;
  padding: 20px 0 10px 0;
  font-size: 20px;
}

.infoText {
  padding: 0 1px 10px 2px;
}

```

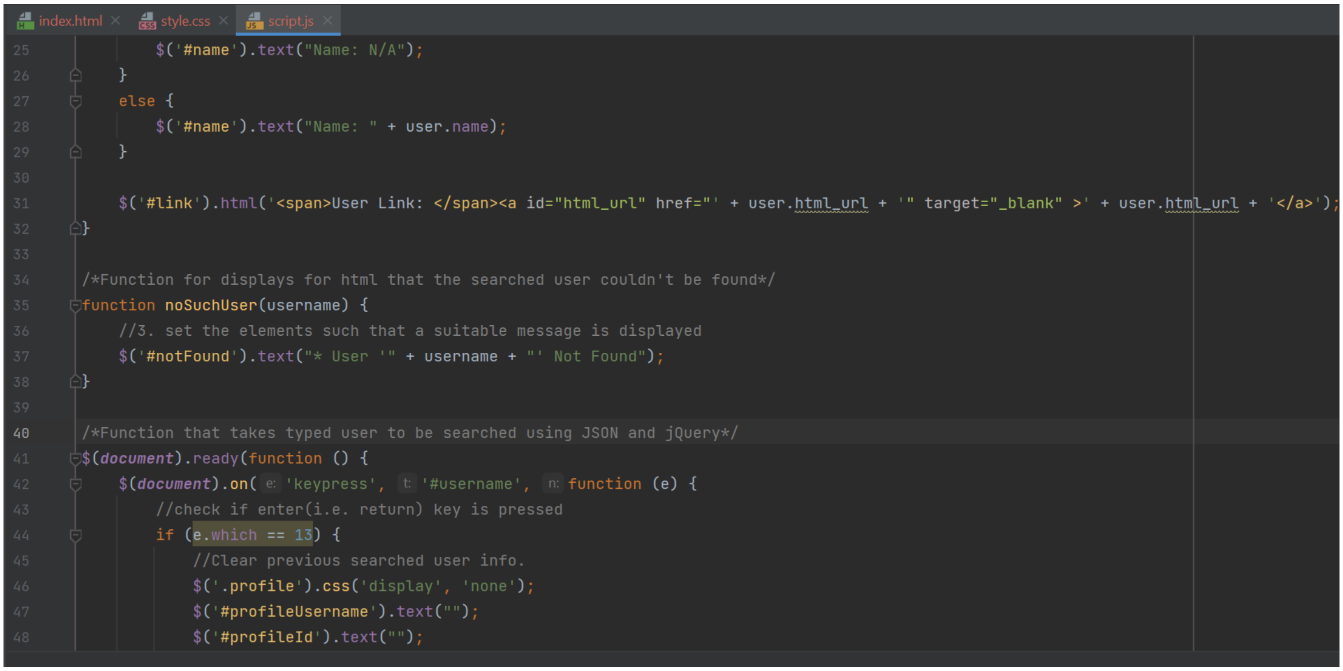


```css
    border-bottom-left-radius: 5px;
    padding: 20px 0 10px 0;
    font-size: 20px;
}

.infoText {
    padding: 0 1px 10px 2px;
}

/*User not found styles*/
#notFound {
    text-align: center;
}
```

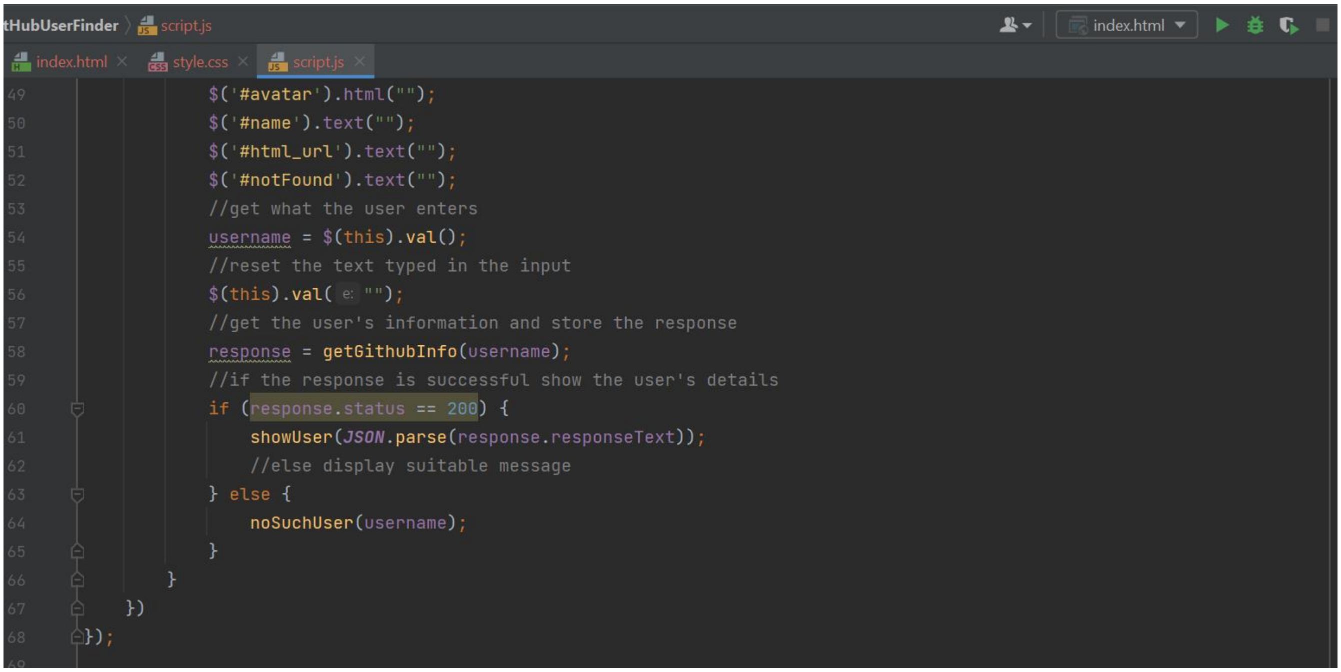
- JS



```javascript
/*JavaScript for GitHub Use Finder*/

/*Function to get user using GitHub API with XMLHttpRequest and Get request*/
function getGithubInfo(user) {
    //1. Create an instance of XMLHttpRequest class and send a GET request using it.
    // The function should finally return the object(it now contains the response!)

    const url = "https://api.github.com/users/"+user;
    const xmlhttp = new XMLHttpRequest();

    xmlhttp.open('GET', url, false);
    xmlhttp.send();
    return xmlhttp;
}

/*Function that displays contents for html gathered on user from the GitHub API*/
function showUser(user) {
    //2. set the contents of the h2 and the two div elements in the div '#profile' with the user content
    $('#profileUsername').text(user.login);
    $('.profile').css('display', 'block');
    $('#avatar').html('<img id="avatarImg" src="' + user.avatar_url + '" alt="avatarImage">');
    $('#profileId').text("ID: " + user.id);
    if (user.name === null)
    {
```

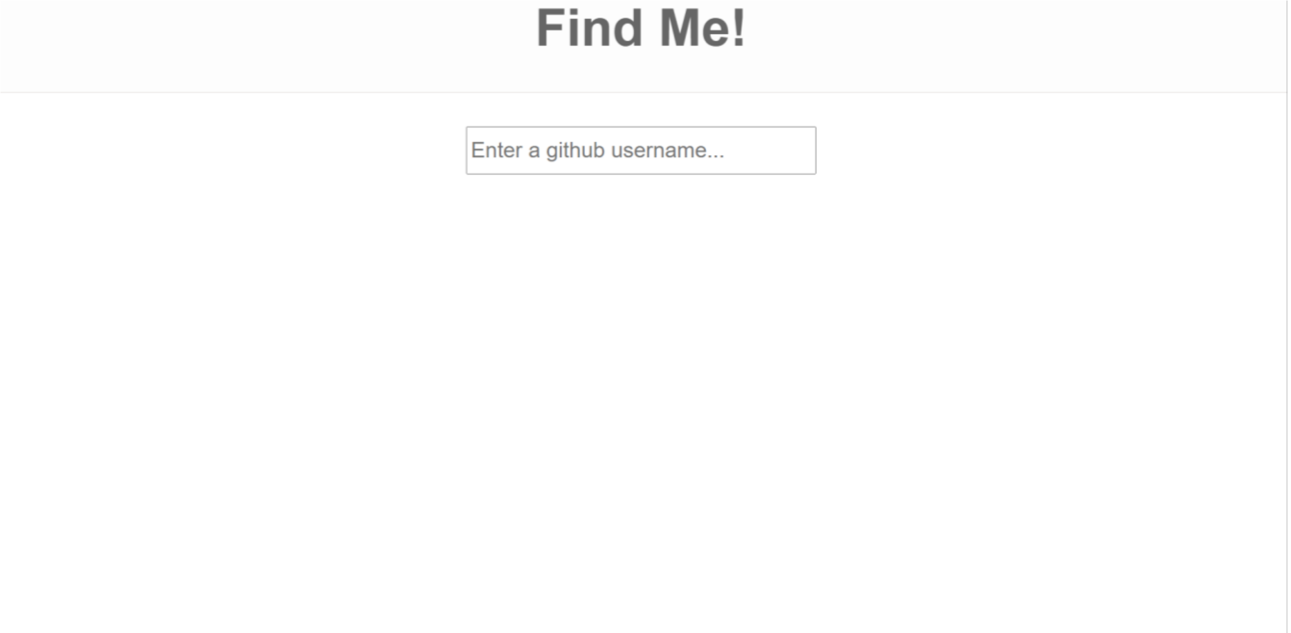


```
        $('#name').text("Name: N/A");
    }
    else {
        $('#name').text("Name: " + user.name);
    }

    $('#link').html('<span>User Link: </span><a id="html_url" href="' + user.html_url + '" target="_blank" >' + user.html_url + '</a>');
}

/*Function for displays for html that the searched user couldn't be found*/
function noSuchUser(username) {
    //3. set the elements such that a suitable message is displayed
    $('#notFound').text("* User '" + username + "' Not Found");
}

/*Function that takes typed user to be searched using JSON and jQuery*/
$(document).ready(function () {
    $(document).on('keypress', '#username', function (e) {
        //check if enter(i.e. return) key is pressed
        if (e.which == 13) {
            //Clear previous searched user info.
            $('.profile').css('display', 'none');
            $('#profileUsername').text("");
            $('#profileId').text("");
```



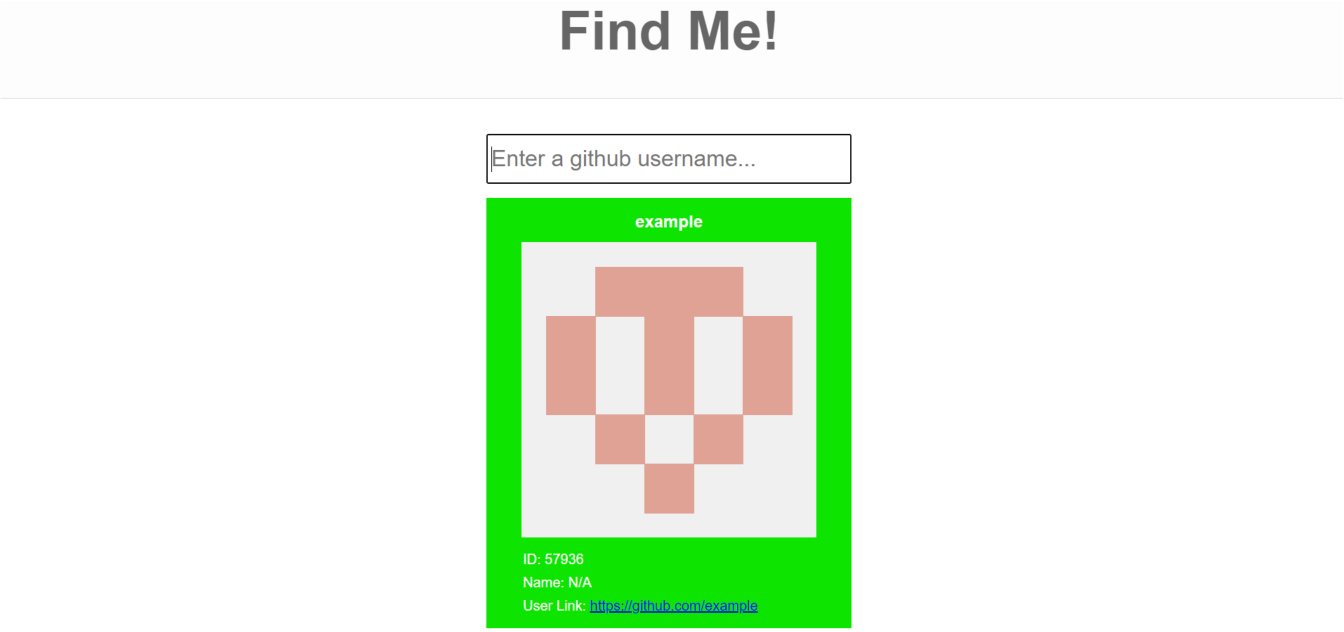
```
            $('#avatar').html("");
            $('#name').text("");
            $('#html_url').text("");
            $('#notFound').text("");
            //get what the user enters
            username = $(this).val();
            //reset the text typed in the input
            $(this).val("");
            //get the user's information and store the response
            response = getGithubInfo(username);
            //if the response is successful show the user's details
            if (response.status == 200) {
                showUser(JSON.parse(response.responseText));
                //else display suitable message
            } else {
                noSuchUser(username);
            }
        }
    })
});
```

- Output



Find Me!

Enter a github username...



Find Me!

Enter a github username...

example

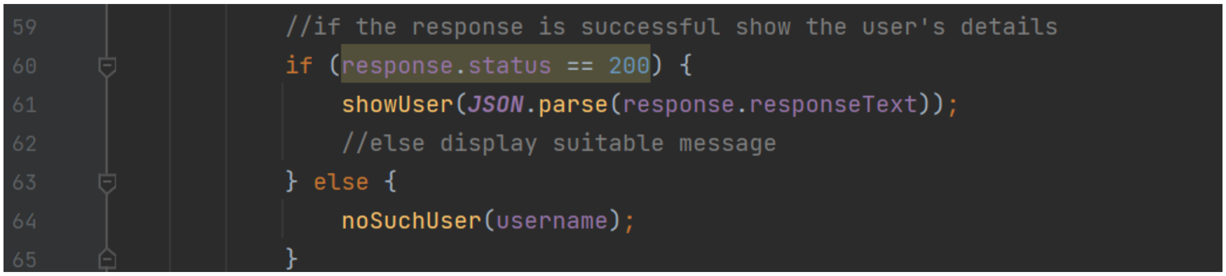
ID: 57936

Name: N/A

User Link: https://github.com/example

JavaScript JSON

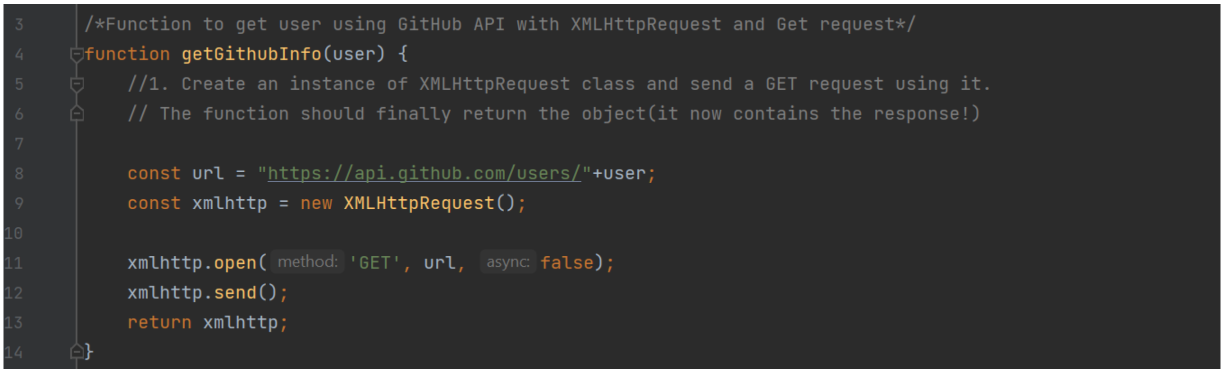
Format to store and move data from a server to a web page.



```
//if the response is successful show the user's details
if (response.status == 200) {
    showUser(JSON.parse(response.responseText));
    //else display suitable message
} else {
    noSuchUser(username);
}
```

XMLHttp Request

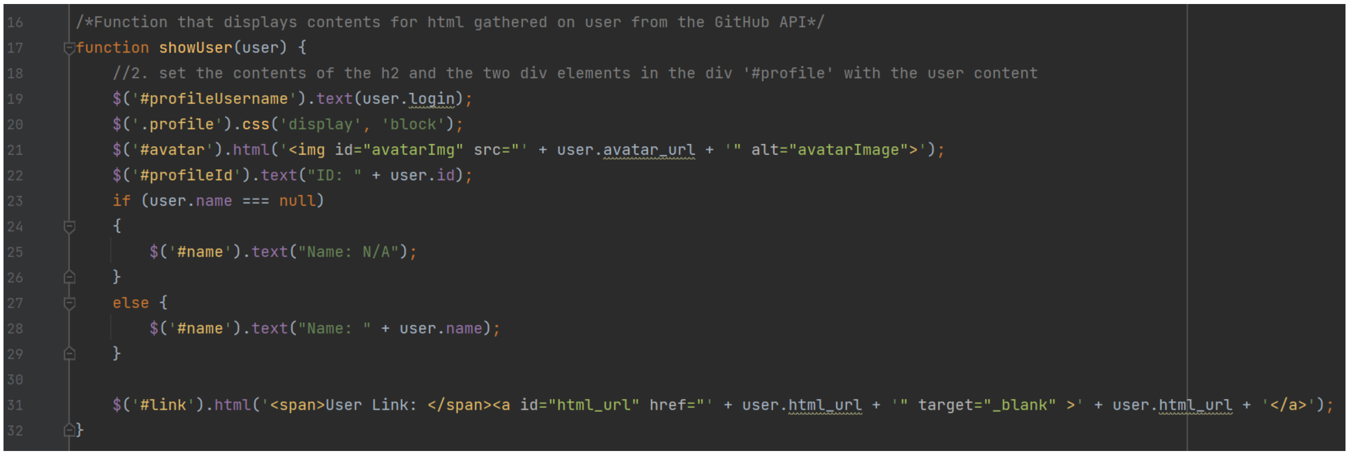
Performs an API or HTTP request exchange data with a web server.



```
/*Function to get user using GitHub API with XMLHttpRequest and Get request*/
function getGithubInfo(user) {
    //1. Create an instance of XMLHttpRequest class and send a GET request using it.
    // The function should finally return the object(it now contains the response!)

    const url = "https://api.github.com/users/"+user;
    const xmlhttp = new XMLHttpRequest();

    xmlhttp.open('GET', url, false);
    xmlhttp.send();
    return xmlhttp;
}
```

Example jQuery Selectors

Selectors to match elements like in a html document to show data collected from a web server.



```
/*Function that displays contents for html gathered on user from the GitHub API*/
function showUser(user) {
    //2. set the contents of the h2 and the two div elements in the div '#profile' with the user content
    $('#profileUsername').text(user.login);
    $('.profile').css('display', 'block');
    $('#avatar').html('<img id="avatarImg" src="' + user.avatar_url + '" alt="avatarImage">');
    $('#profileId').text("ID: " + user.id);
    if (user.name === null)
    {
        $('#name').text("Name: N/A");
    }
    else {
        $('#name').text("Name: " + user.name);
    }

    $('#link').html('<span>User Link: </span><a id="html_url" href="' + user.html_url + '" target="_blank" >' + user.html_url + '</a>');
}
```

Contributors

I worked independently, so I am the sole contributor.

Conclusion

I used Object-Oriented JavaScript, jQuery, and the GitHub API to create a web application that display's a valid GitHub user's account details that's searched by their username such as their name, avatar, profile picture and link to their GitHub account. I used the Object-Oriented part of JavaScript to use functions to perform actions on other objects. I called the GitHub API to find and collect information on a GitHub user. I used jQuery and it's selectors to pass elements of user that is called by the GitHub API to show information from that user. Lastly, I used XMLHttp Request to call and get information from the GitHub API.