Practical 17

Advanced JavaScript and API Integration  
Task: GitHub Profile Viewer

* Objective: Create an application that fetches and displays GitHub profiles using the GitHub API.
* Activities:
* Create a form to input a GitHub username.
* Fetch the user’s profile data from the GitHub API.
* Display the user’s profile information, including avatar, repositories, and followers.
* <!DOCTYPE html>
* <html lang="en">
* <head>
* <meta charset="UTF-8">
* <meta http-equiv="X-UA-Compatible" content="IE=edge">
* <meta name="viewport" content="width=device-width, initial-scale=1.0">
* <!-- Font Awesome cdn -->
* <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.2/css/all.min.css"
* integrity="sha512-1sCRPdkRXhBV2PBLUdRb4tMg1w2YPf37qatUFeS7zlBy7jJI8Lf4VHwWfZZfpXtYSLy85pkm9GaYVYMfw5BC1A=="
* crossorigin="anonymous" referrerpolicy="no-referrer" />
* <!-- Google Font -->
* <link rel="preconnect" href="https://fonts.googleapis.com">
* <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
* <link
* href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,400;0,500;0,700;1,400;1,500;1,700&display=swap"
* rel="stylesheet">
* <!-- Css file -->
* <link rel="stylesheet" href="pr17.css">
* <title>GitHub Profile Viewer</title>
* </head>
* <body>
* <div class="container">
* <!-- Header -->
* <div class="header">
* <h3>GitHub Profile Viewer</h3>
* </div>
* <!-- Search Section -->
* <div class="search-section">
* <input type="text" placeholder="Search GitHub username.." class="search-input">
* <i class="fa-solid fa-magnifying-glass"></i>
* </div>
* <!-- Main Section -->
* <div class="github loading">
* <img src="https://avatars.githubusercontent.com/u/89848148?v=4" alt="" class="profile-picture">
* <div class="info">
* <div class="name">Ramin Mikayilov</div>
* <div class="login">RaminMikayilov</div>
* <div class="bio">FrontEnd Developer</div>
* <div class="d-grid-3">
* <div class="repo-section">
* <div class="repos">Repositories</div>
* <div class="repos-number">6</div>
* </div>
* <div class="follower-section">
* <div class="follower">Followers</div>
* <div class="follower-number">7</div>
* </div>
* <div class="following-section">
* <div class="following">Following</div>
* <div class="following-number">8</div>
* </div>
* </div>
* <div class="d-grid-2">
* <div class="location-section">
* <i class="fa-solid fa-location-dot"></i>
* <span class="location"></span>
* </div>
* <div class="company-section">
* <i class="fa-solid fa-building"></i>
* <span class="company"></span>
* </div>
* </div>
* </div>
* </div>
* <!-- Error Section -->
* <div class="error">
* <i class="fa-regular fa-circle-xmark"></i>
* Username not found/Incorrect UserName
* </div>
* </div>
* <script src="pr17.js"></script>
* </body>
* </html>

\* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

}

body {

    font-family: 'Roboto', sans-serif;

    background-color: #110611;

    height: 100vh;

    display: flex;

    align-items: center;

    justify-content: center;

    background-image: url('img/background.jpg');

    background-size: cover;

    background-position: center;

    background-repeat: no-repeat;

}

img {

    width: 100%;

}

.container {

    max-width: 90%;

    width: 600px;

    color: #eee;

}

.header {

    display: flex;

    align-items: center;

    justify-content: space-between;

}

.header h3 {

    font-style: italic;

    font-size: 30px;

}

.mode {

    text-transform: uppercase;

    cursor: pointer;

}

.search-section {

    margin: 20px 0;

    display: flex;

    align-items: center;

    justify-content: space-between;

    border-radius: 10px;

    padding-right: 20px;

    background: rgba(255, 255, 255, 0.2);

    box-shadow: 0 4px 30px rgba(0, 0, 200, 0.2);

    backdrop-filter: blur(10px);

    border: 1px solid rgba(255, 255, 255, 0.3);

}

.search-section i {

    font-size: 25px;

    cursor: pointer;

    transition: 0.3s ease-in-out;

}

.search-section i:hover {

    opacity: 0.7;

}

.search-input {

    width: 100%;

    height: 100%;

    outline: none;

    border: none;

    background-color: transparent;

    color: #eee;

    font-size: 20px;

    padding: 15px 20px;

}

.github {

    padding: 25px;

    display: grid;

    grid-template-columns: 1fr 3fr;

    border-radius: 10px;

    background: rgba(255, 255, 255, 0.1);

    box-shadow: 0 4px 30px rgba(0, 0, 200, 0.2);

    backdrop-filter: blur(5px);

    border: 1px solid rgba(255, 255, 255, 0.3);

}

.profile-picture {

    width: 100px;

    height: 100px;

    border-radius: 50%;

}

.info {

    padding-left: 10px;

}

.name {

    font-size: 25px;

    font-weight: 700;

    padding-top: 15px;

    padding-bottom: 5px;

}

.login {

    color: #ccc;

}

.bio {

    padding: 15px 0;

}

.d-grid-3 {

    display: grid;

    grid-template-columns: repeat(3, 1fr);

    text-align: center;

    padding: 15px;

    background-color: rgba(32, 23, 39, 0.8);

    border-radius: 10px;

}

.repos-number,

.follower-number,

.following-number {

    font-weight: 700;

    font-size: 25px;

    padding-top: 5px;

}

.d-grid-2 {

    display: grid;

    grid-template-columns: repeat(2, 1fr);

    padding-top: 25px;

}

.d-grid-2 i {

    margin-right: 3px;

}

.github.loading {

    visibility: hidden;

    max-height: 20px;

    position: relative;

}

.github.loading:after {

    visibility: visible;

    content: "Loading...";

    color: white;

    font-size: 25px;

    position: absolute;

    top: 20px;

    left: 20px;

}

.error {

    display: none;

    padding: 25px;

    border-radius: 10px;

    background: rgba(255, 255, 255, 0.1);

    box-shadow: 0 4px 30px rgba(0, 0, 200, 0.2);

    backdrop-filter: blur(5px);

    border: 1px solid rgba(255, 255, 255, 0.3);

    text-align: center;

    font-size: 25px;

}

@media (max-width: 650px) {

    .github {

        padding: 15px;

        grid-template-columns: 1fr;

    }

    .github img {

        margin: 0 auto;

    }

}

const api = 'https://api.github.com/users/';

const input = document.querySelector(".search-input");

const search = document.querySelector(".search-section i");

function getData(data) {

    fetch(`${api}${data}`)

        .then(response => response.json())

        .then(display)

        .catch(err => console.log(err));

}

function display(response) {

    if (response.message == 'Not Found') {

        // Show error

        document.querySelector(".github").style.display = "none";

        document.querySelector(".error").style.display = "block";

    } else {

        document.querySelector(".github").style.display = "grid";

        document.querySelector(".error").style.display = "none";

        input.value = '';

        document.querySelector(".profile-picture").src = response.avatar\_url;

        document.querySelector(".name").innerHTML = response.name;

        document.querySelector(".login").innerHTML = response.login;

        document.querySelector(".bio").innerHTML = response.bio;

        document.querySelector(".repos-number").innerHTML = response.public\_repos;

        document.querySelector(".follower-number").innerHTML = response.followers;

        document.querySelector(".following-number").innerHTML = response.following;

        // document.querySelector(".location").innerHTML = response.location;

        // document.querySelector(".company").innerHTML = response.company;

        const loc = document.querySelector(".location");

        const comp = document.querySelector(".company");

        response.location == null ? loc.innerHTML = 'Not Available' : loc.innerHTML = response.location;

        response.company == null ? comp.innerHTML = 'Not Available' : comp.innerHTML = response.company;

    }

    document.querySelector(".github").classList.remove("loading");

}

search.addEventListener("click", () => {

    getData(input.value);

});

input.addEventListener("keypress", (event) => {

    // If the user presses the "Enter" key on the keyboard

    if (event.key === "Enter") {

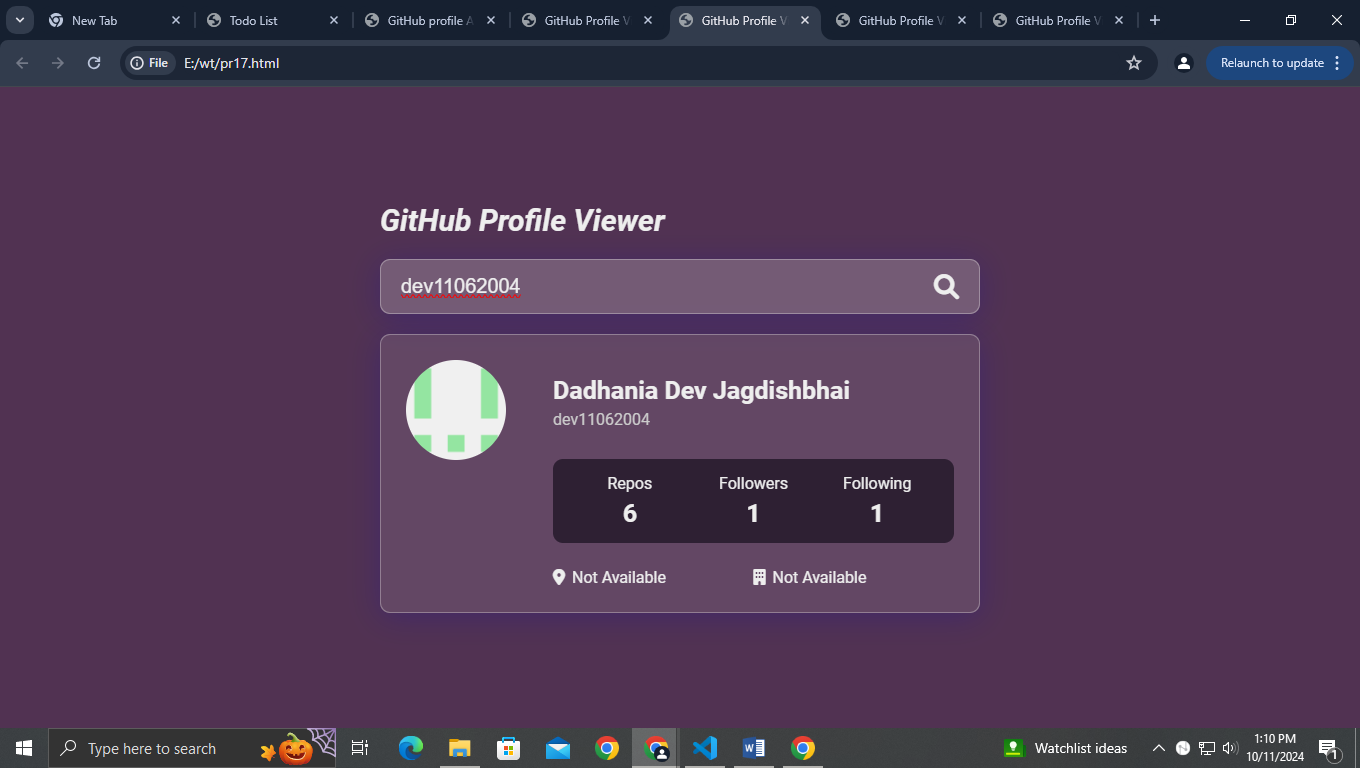
        getData(input.value);

    }

});

// My account

getData('dev11062004');



Learning outcomes:-

1. **Improved API Integration Skills**: By working with the GitHub API, you will enhance your ability to send requests, handle responses, and manage errors in JavaScript. This experience will deepen your understanding of RESTful APIs and how to interact with them effectively.
2. **Enhanced Front-End Development**: Creating a user interface for the application will help you practice and refine your skills in HTML, CSS, and JavaScript. You'll learn how to design forms, manage state, and dynamically update the DOM based on API responses.
3. **Data Handling and Presentation**: You will gain experience in parsing and manipulating JSON data. This includes extracting relevant information such as avatar URLs, repository details, and follower counts, and presenting it in a user-friendly format.
4. **User Experience Improvement**: Implementing features like input validation and error handling will teach you the importance of user experience (UX) design.