

Terminal curl

```
1  # cURL Demo
2
3  In this activity, you will demonstrate how to use cURL in the terminal.
4
5  ## Instructions
6
7  * Open your terminal and type `curl https://api.github.com/users`.
8
9  * This should return a list of the last 30 users who made a profile
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

1: powershell

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS C:\Users\const\Documents\UNCC BootCamp\Class\Week 6\06-Server-Side-APIs> curl https://api.github.com/users

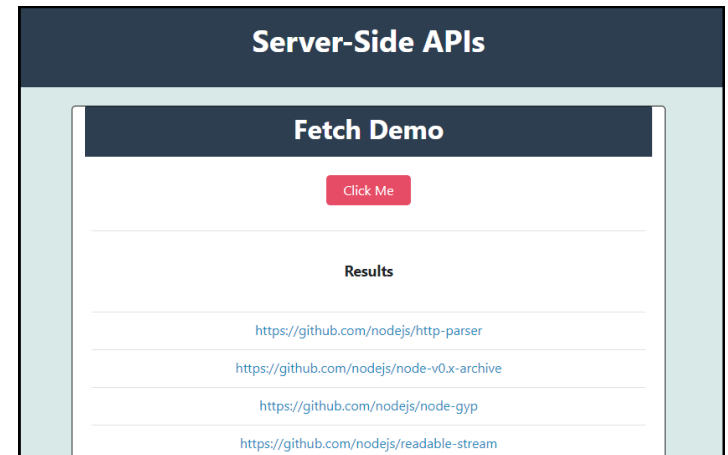
```
StatusCode      : 200
StatusDescription : OK
Content          : [{"login":"mojombo","id":1,"node_id":"MDQ6VXNlcjE=", "avatar_url":"https://avatars.githubusercontent.com/u/1?v=4", "gravatar_id":"","url":"https://api.github.com/users/mojombo", "html_url":"https://github.com/mojombo", "followers_url":"https://api.github.com/users/mojombo/followers", "following_url":"https://api.github.com/users/mojombo/following", "subscriptions_url":"https://api.github.com/users/mojombo/subscriptions", "organizations_url":"https://api.github.com/users/mojombo/orgs", "repos_url":"https://api.github.com/users/mojombo/repos", "events_url":"https://api.github.com/users/mojombo/events", "received_events_url":"https://api.github.com/users/mojombo/received_events", "type":"User", "site_admin":false}]
RawContent       : HTTP/1.1 200 OK
                  Vary: Accept, Accept-Encoding, Accept, X-Requested-With
                  X-GitHub-Media-Type: github.v3; format=json
                  Link: <https://api.github.com/users?since=46>; rel="next", <https://api.github.com/users?since=46>; rel="first"
Forms            : {}
Headers          : {[Vary, Accept, Accept-Encoding, Accept, X-Requested-With], [X-GitHub-Media-Type, github.v3; format=json], [Link, <https://api.github.com/users?since=46>; rel="next", <https://api.github.com/users?since=46>; rel="first"
Images           : {}
InputFields      : {}
Links            : {}
ParsedHtml       : mshtml.HTMLDocumentClass
RawContentLength : 26503
```

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Fetch Demo</title>
  <link rel="stylesheet" href="/assets/css/style.css" />
</head>
<body class="min-100-vh flex-column">
  <header class="p-5 text-center text-white bg-primary">
    <h1>Server-Side APIs</h1>
  </header>
  <main class="flex-row justify-center align-center col-auto p-4 bg-light">
    <section class="card card-rounded col-7 bg-white">
      <header class="card-header text-center">
        <h2 class="text-white">
          Fetch Demo
        </h2>
      </header>
      <div class="text-center pt-4">
        <button class="btn btn-danger" id="fetch-button">
          Click Me
        </button>
      </div>
      <table class="card-body table mx-auto my-4">
        <thead>
          <tr>
            <th>
              <p class="my-4">Results</p>
            </th>
          </tr>
        </thead>
        <tbody id="repo-table" class="mx-auto text-center"></tbody>
      </table>
    </section>
    <script src="/assets/js/script.js"></script>
  </main>
</body>
</html>

```

Demo Fetch



```

var tableBody = document.getElementById('repo-table');
var fetchButton = document.getElementById('fetch-button');
function getApi() {

```

// fetch request gets a list of all the repos for the node.js organization

```
var requestUrl = 'https://api.github.com/orgs/nodejs/repos';
```

```
fetch(requestUrl)
```

```
  .then(function (response) {
    return response.json();
  })
```

```
  .then(function (data) {
    console.log(data)

```

// Loop over the data to generate a table, each table row will have a link to the repo url
for (var i = 0; i < data.length; i++) {

// Creating elements, tablerow, tabledata, and anchor

```
var createTableRow = document.createElement('tr');
var tableData = document.createElement('td');
var link = document.createElement('a');
```

// Setting the text of link and the href of the link

```
link.textContent = data[i].html_url;
link.href = data[i].html_url;
```

// Appending the link to the tabledata and then appending the tabledata to the tablerow

// The tablerow then gets appended to the tablebody

```
tableData.appendChild(link);
createTableRow.appendChild(tableData);
tableBody.appendChild(createTableRow);

```

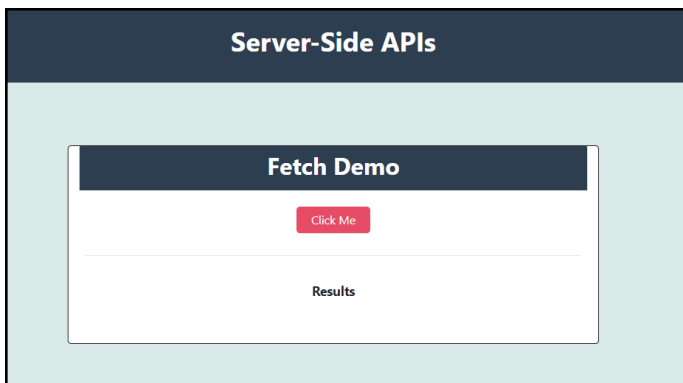
```
}
```

```
});
```

```
}
fetchButton.addEventListener('click', getApi);

```

Server-Side APIs



```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
===== */
/**

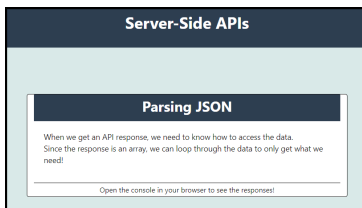
```



```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Parsing JSON</title>
  <link rel="stylesheet" type="text/css" href="/assets/css/style.css" />
  <script
    src="https://code.jquery.com/jquery-3.5.1.min.js"
    integrity="sha256-9/aliU8dGd2tb6OSsuzixeV4y/faTqgFtohetphbbj0="
    crossorigin="anonymous"
  ></script>
</head>
<body class="min-100-vh flex-column">
  <header class="p-5 text-center text-white bg-primary">
    <h1>Server-Side APIs</h1>
  </header>
  <main class="flex-row justify-center align-center col-auto p-4 bg-light">
    <section class="card card-rounded col-7 bg-white">
      <header class="card-header text-center">
        <h2 class="text-white">
          Parsing JSON
        </h2>
      </header>
      <div class="p-3">
        <p class="card-body">
          When we get an API response, we need to know how to access the data.
          <br />
          Since the response is an array, we can loop through the data to only
          get what we need!
        </p>
      </div>
      <footer class="card-footer text-center">
        Open the console in your browser to see the responses!
      </footer>
    </section>
  </main>
  <script src="assets/js/script.js"></script>
</body>
</html>

```



```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
=====*/
/**

```

Parse

```

Twitter Public Members: Raw data
-----
script.js:15
script.js:17
Array(30)
  0: {login: "MichaelEvans", id: 1211302, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://av..."}
  1: {login: "OrkutHunter", id: 8865911, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  2: {login: "adawcwt", id: 18803549, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  3: {login: "alykhantejani", id: 687194, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://av..."}
  4: {login: "andypiper", id: 552452, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  5: {login: "cacoco", id: 613846, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars.g..."}
  6: {login: "Cassaco", id: 6912589, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  7: {login: "couch", id: 189562, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars.g..."}
  8: {login: "enaforn", id: 184755, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars..."}
  9: {login: "evensobokidic", id: 1407340, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://a..."}
  10: {login: "feborghino", id: 166738, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avat..."}
  11: {login: "gr-laraper", id: 433817, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avat..."}
  12: {login: "jamesryanalexander", id: 5247389, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "htt..."}
  13: {login: "kennethcurflem", id: 156496, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://a..."}
  14: {login: "kevinoliver", id: 347229, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://www..."}
  15: {login: "kgoodier", id: 6953724, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avata..."}
  16: {login: "kwliz", id: 1885523, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars..."}
  17: {login: "klu", id: 13896, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars.github..."}
  18: {login: "mkouyda", id: 1431805, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avata..."}
  19: {login: "nvgrv", id: 2185177, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars..."}
  20: {login: "objsmagic", id: 3654079, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avata..."}
  21: {login: "pavanky", id: 364829, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars..."}
  22: {login: "richardveneman", id: 75705, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  23: {login: "roanta", id: 1617368, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars..."}
  24: {login: "rondevora", id: 21899, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars.g..."}
  25: {login: "ryancouto", id: 3813481, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  26: {login: "sagewinblue", id: 448162, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://av..."}
  27: {login: "skillixdev", id: 153525, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avate..."}
  28: {login: "sllyphon", id: 46668, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars.gith..."}
  29: {login: "stakra", id: 3238322, node_id: "MDQ6VWV1c3JyYXN0bWU=", avatar_url: "https://avatars..."}
  length: 30
  __proto__: Array(30)

```

Twitter Repositories: Names only	script.js:15

hadoop-1.2	script.js:17
thrift_client	script.js:17
twu1	script.js:17
elephant-bird	script.js:17
ostrich	script.js:17
scala_school	script.js:17
finagle	script.js:17
Joornb	script.js:17
twitter.github.io	script.js:17
util	script.js:17
scrooge	script.js:17
pycascading	script.js:17
hogan.js	script.js:17
scalding	script.js:17
effectivescala	script.js:17
finatra	script.js:17
cassovary	script.js:17
tuemproy	script.js:17
twitter-clr-rb	script.js:17
innovators-patent-agreement	script.js:17
recess	script.js:17
ambrose	script.js:17
activerecord-reputation-system	script.js:17
tuemcache	script.js:17
twitter-clr-js	script.js:17
algebird	script.js:17
twitter-clr-npm	script.js:17
tormenta	script.js:17
summingbird	script.js:17
chill	script.js:17

```

fetch("https://api.github.com/orgs/twitter/public_members")
.then(function (response) {
  return response.json();
})
.then(function (data) {
  console.log('Twitter Public Members: Raw data \n-----');
  console.log(data);
});

```

```

fetch("https://api.github.com/orgs/twitter/repos")
.then(function (response) {
  return response.json();
})
.then(function (data) {
  console.log('Twitter Repositories: Names only \n-----');
  for (var i = 0; i < data.length; i++) {
    console.log(data[i].name);
  }
});

```

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Dynamically Generated Elements</title>
  <link rel="stylesheet" href="/assets/css/style.css" />
</head>
<body class="min-100-vh flex-column">
  <header class="p-5 text-center text-white bg-primary">
    <h1>Server-Side APIs</h1>
  </header>
  <main class="flex-row justify-center align-center col-auto p-4 bg-light">
    <section class="card card-rounded col-7 bg-white">
      <header class="card-header text-center">
        <h2 class="text-white">
          Dynamically Generated Elements
        </h2>
      </header>
      <div class="text-center p-4">
        <p class="card-body">
          When we hit the 'Click Me' button, we see a list of the last 5
          GitHub issues from the IBM 'clai' repo. This content is dynamically
          generated using the response we get from the API request!
        </p>
        <button class="btn btn-danger" id="fetch-button">
          Click Me
        </button>
      </div>
      <div class="container" id="issues"></div>
    </section>
  </main>
  <script src="/assets/js/script.js"></script>
</body>
</html>

```

Dynamic Elements

Server-Side APIs

Dynamically Generated Elements

When we hit the 'Click Me' button, we see a list of the last 5 GitHub issues from the IBM 'clai' repo. This content is dynamically generated using the response we get from the API request!

Click Me

Server-Side APIs

Dynamically Generated Elements

When we hit the 'Click Me' button, we see a list of the last 5 GitHub issues from the IBM 'clai' repo. This content is dynamically generated using the response we get from the API request!

Click Me

dependabot[bot]

Bump pydantic from 0.27 to 1.6.2

dependabot[bot]

Bump pillow from 7.1.1 to 8.1.1

dependabot[bot]

Bump pillow from 7.1.1 to 8.1.1 in /clai/server/plugins/dataplore

ojycojyc

Support CLAI execution in Windows container (and Windows OS local environment)

TathagataChakraborti

New NLC2CMD team

```

var issueContainer = document.getElementById('issues');
var fetchButton = document.getElementById('fetch-button');

```

```

function getApi() {
  var requestUrl = 'https://api.github.com/repos/IBM/clai/issues?per_page=5';

```

```

  fetch(requestUrl)
    .then(function (response) {
      return response.json();
    })
    .then(function (data) {
      console.log(data);
      for (var i = 0; i < data.length; i++) {
        var userName = document.createElement('h3');
        var issueTitle = document.createElement('p');
        userName.textContent = data[i].user.login;
        issueTitle.textContent = data[i].title;
        issueContainer.append(userName);
        issueContainer.append(issueTitle);
      }
    });
}

```

```

fetchButton.addEventListener('click', getApi);

```

```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
=====*/
/**

```

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Network Activity</title>
    <link rel="stylesheet" href="/assets/css/style.css" />
  </head>
  <body class="min-100-vh flex-column">
    <header class="p-5 text-center text-white bg-primary">
      <h1>Server-Side APIs</h1>
    </header>
    <main class="flex-row justify-center align-center col-auto p-4 bg-light">
      <section class="card card-rounded col-7 bg-white">
        <header class="card-header text-center">
          <h2 class="text-white">
            Network Activity Tab
          </h2>
        </header>
        <div class="p-3">
          <p class="card-body">
            Open the network activity tab!
          <br />
            Press command/ctrl R to record the reload.
          <br />
            The status codes indicate whether these resources came through
            successfully or not.
          <br />
            Status codes will quickly let us know what happened with a
            particular request.
          </p>
        </div>
        <footer class="card-footer text-center">
          Open the network activity tab and examine the status of each resource!
        </footer>
      </section>
    </main>
    <script src="assets/js/script.js"></script>
  </body>
</html>

```

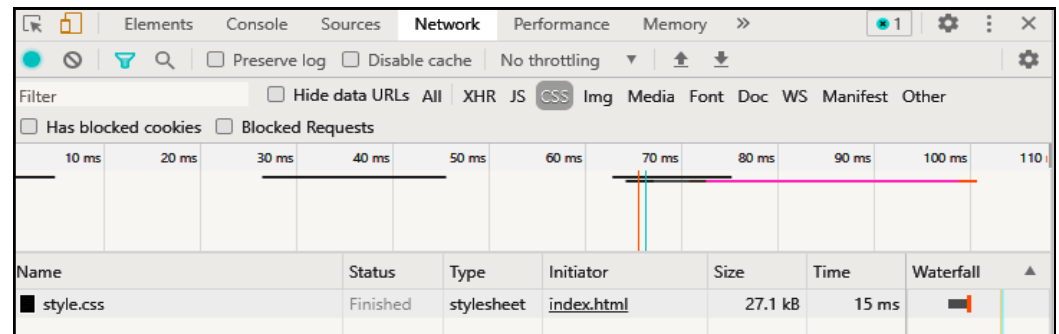
Network Activity

Server-Side APIs

Network Activity Tab

Open the network activity tab!
 Press command/ctrl R to record the reload.
 The status codes indicate whether these
 resources came through successfully or not.
 Status codes will quickly let us know what
 happened with a particular request.

Open the network activity tab and examine the status
 of each resource!



```

var requestUrl = 'https://api.github.com/orgs/nodejs/repos';
var badRequestUrl = 'https://api.github.com/orgs/nodejddd/repad';

```

```

fetch(requestUrl)
  .then(function (response) {
    return response.json();
  })
  .then(function (data) {
    console.log(data);
  });

```

```

fetch(badRequestUrl)
  .then(function (response) {
    return response.json();
  })
  .then(function (data) {
    console.log(data);
  });

```

```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
===== */
/**

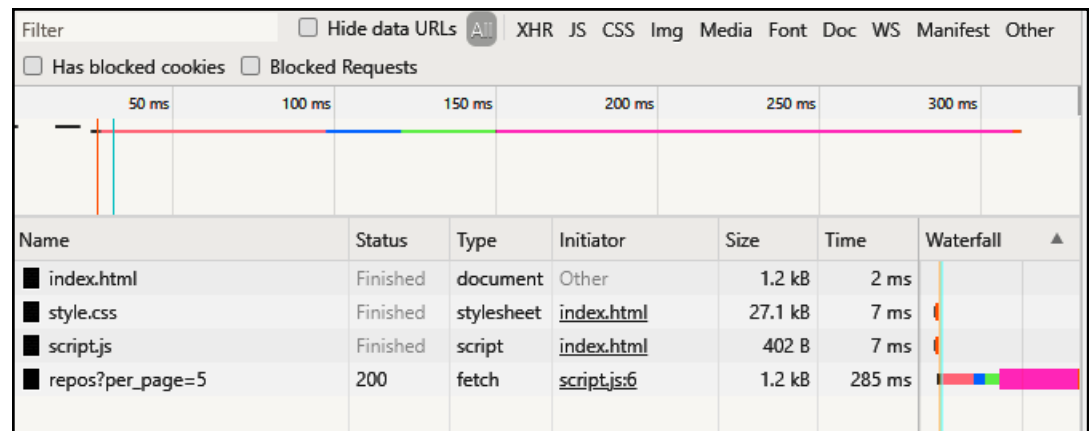
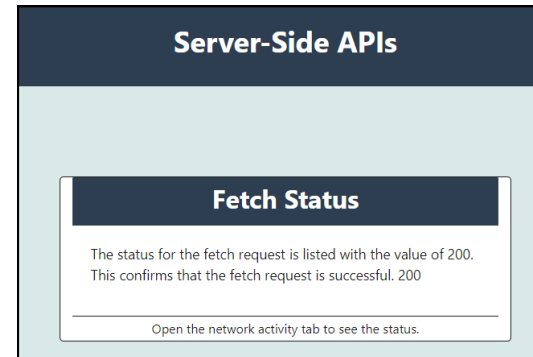
```

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Fetch Response Status</title>
    <link rel="stylesheet" href="/assets/css/style.css" />
  </head>
  <body class="min-100-vh flex-column">
    <header class="p-5 text-center text-white bg-primary">
      <h1>Server-Side APIs</h1>
    </header>
    <main class="flex-row justify-center align-center col-auto p-4 bg-light">
      <section class="card card-rounded col-7 bg-white">
        <header class="card-header text-center">
          <h2 class="text-white">
            Fetch Status
          </h2>
        </header>
        <div class="p-3">
          <p class="card-body">
            The status for the fetch request is listed with the value of 200.
            <br />
            This confirms that the fetch request is successful.
            <span id="response-text"></span>
          </p>
        </div>
        <footer class="card-footer text-center">
          Open the network activity tab to see the status.
        </footer>
      </section>
    </main>
    <script src="assets/js/script.js"></script>
  </body>
</html>

```

Fetch Status



```

var requestUrl = 'https://api.github.com/orgs/nodejs/repos?per_page=5';
var responseText = document.getElementById('response-text');

```

```

function getApi(requestUrl) {
  fetch(requestUrl)
    .then(function (response) {
      console.log(response);
      if (response.status === 200) {
        responseText.textContent = response.status;
      }
      return response.json();
    });
}

```

```

getApi(requestUrl);

```

```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
===== */
/**

```



```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Deconstruct Parameters</title>
<link rel="stylesheet" type="text/css" href="/assets/css/style.css" />
<script
src="https://code.jquery.com/jquery-3.5.1.min.js"
integrity="sha256-9/aliU8dGd2tb6OSsuzixeV4y/faTqgFtohetphbbj0="
crossorigin="anonymous"
></script>
</head>
<body class="min-100-vh flex-column">
<header class="p-5 text-center text-white bg-primary">
<h1>Server-Side APIs</h1>
</header>
<main class="flex-row justify-center align-center col-auto p-4 bg-light">
<section class="card card-rounded col-7 bg-white">
<header class="card-header text-center">
<h2 class="text-white">
Deconstruct Parameters
</h2>
</header>
<div class="p-3"></div>
<footer class="card-footer text-center">
Open the console in your browser to see the responses!
</footer>
</section>
</main>
<script src="assets/js/script.js"></script>
</body>
</html>

```

Deconstruct Parameters

```

Array(100)
0: {url: "https://api.github.com/gists/182e68574d01d7ac78a7f4a720c45c60", forks_url: "h..."}
1: {url: "https://api.github.com/gists/f504e45e85d89e3404c5a8766911da16", forks_url: "h..."}
2: {url: "https://api.github.com/gists/150924588b5d5029779888a8f8e808f", forks_url: "h..."}
3: {url: "https://api.github.com/gists/dbc9f9c8902de87918855dc79c17e4b3", forks_url: "h..."}
4: {url: "https://api.github.com/gists/b5f9fef37ee47c04fa06142934b24be5", forks_url: "h..."}
5: {url: "https://api.github.com/gists/c75052912c8d98cf1a677c69425e32e1", forks_url: "h..."}
6: {url: "https://api.github.com/gists/b48fc7860b0983470ac678c56479fa7f", forks_url: "h..."}
7: {url: "https://api.github.com/gists/044b7817021e5bce6e380e5ec1511b65", forks_url: "h..."}
8: {url: "https://api.github.com/gists/e0b2f05a594cdd7d8e8a709c7e2d9529", forks_url: "h..."}
9: {url: "https://api.github.com/gists/400b1367d5549ed91c958dd71471c008", forks_url: "h..."}
10: {url: "https://api.github.com/gists/1920f52a1b23d019d3680bf018dae2ff", forks_url: "h..."}
11: {url: "https://api.github.com/gists/5d9df4ff4bf7a4b036245fcb3b9232", forks_url: "h..."}
12: {url: "https://api.github.com/gists/30591a3acdb237e63d1c5e31b3f6d956", forks_url: "h..."}
13: {url: "https://api.github.com/gists/22cb4d281b88b90969ee6f0e3c7e0f81", forks_url: "h..."}
14: {url: "https://api.github.com/gists/9b2c591732405887f58414a940449afa", forks_url: "h..."}
15: {url: "https://api.github.com/gists/18bed6a56fa42df4afcd4099c8e68dc9", forks_url: "h..."}
16: {url: "https://api.github.com/gists/3ae2e2015319d41abdb2891e5b55b95b", forks_url: "h..."}
17: {url: "https://api.github.com/gists/226a2da5151e1b5d8c6b1f8f11b3e02f", forks_url: "h..."}
18: {url: "https://api.github.com/gists/b6eeabae985afded841b69bca89e4cde", forks_url: "h..."}
19: {url: "https://api.github.com/gists/513f6110b9863210252578f5d633969a", forks_url: "h..."}
20: {url: "https://api.github.com/gists/a08560dd9a8b367a51e8f324387c4816", forks_url: "h..."}
21: {url: "https://api.github.com/gists/db84716444b8e24aca21621b39f580", forks_url: "h..."}
22: {url: "https://api.github.com/gists/1aa3f8d8864210b88d6bda7662bd09bf", forks_url: "h..."}
23: {url: "https://api.github.com/gists/429c6cdfaef32d2a0fde4357bd56791a", forks_url: "h..."}
24: {url: "https://api.github.com/gists/02483b114398693bd18bc2f013ac76e0", forks_url: "h..."}
25: {url: "https://api.github.com/gists/2e86b35bd3cc3ddfd58810b88392d7ed", forks_url: "h..."}
26: {url: "https://api.github.com/gists/d3f8a0f00d1daae6affb58b8cbe3c69e", forks_url: "h..."}
27: {url: "https://api.github.com/gists/94adc7b21c9b897137c572d67e0b0b44", forks_url: "h..."}
28: {url: "https://api.github.com/gists/c83eb8fa52e0b81e3465dab23b365044", forks_url: "h..."}
29: {url: "https://api.github.com/gists/574b5a050a031a1d67129b9c0731f77a", forks_url: "h..."}
30: {url: "https://api.github.com/gists/849616552d2788266e5cc774139f9071", forks_url: "h..."}
31: {url: "https://api.github.com/gists/2035a46ce49a9f1e675124643588db13", forks_url: "h..."}
32: {url: "https://api.github.com/gists/ffaf1d0ecbb0255f4340159c21a44637", forks_url: "h..."}
33: {url: "https://api.github.com/gists/d1c460d02c561c473b4e1f81bf0f1278", forks_url: "h..."}
34: {url: "https://api.github.com/gists/3349f776266adbff6b31fe5f9b25e0fa", forks_url: "h..."}
35: {url: "https://api.github.com/gists/6117d220ef85068f78e2595ec598bf36", forks_url: "h..."}

```

Server-Side APIs

Deconstruct Parameters

Open the console in your browser to see the responses!

```

fetch('https://api.github.com/gists/public?since=2020-06-01&per_page=100')
.then(function (response) {
  return response.json();
})
.then(function (data) {
  console.log(data);
});

```

```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
=====*/
/**

```



```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Fetch Options</title>
    <link rel="stylesheet" type="text/css" href="/assets/css/style.css" />
    <script
      src="https://code.jquery.com/jquery-3.5.1.min.js"
      integrity="sha256-9/aliU8dGd2tb6OSsuzixeV4y/faTqgFtohetphbbj0="
      crossorigin="anonymous"
    ></script>
  </head>
  <body class="min-100-vh flex-column">
    <header class="p-5 text-center text-white bg-primary">
      <h1>Server-Side APIs</h1>
    </header>
    <main class="flex-row justify-center align-center col-auto p-4 bg-light">
      <section class="card card-rounded col-7 bg-white">
        <header class="card-header text-center">
          <h2 class="text-white">
            Fetch Options
          </h2>
        </header>
        <div class="p-3"></div>
        <footer class="card-footer text-center">
          Open the console in your browser to see the responses!
        </footer>
      </section>
    </main>
    <script src="assets/js/script.js"></script>
  </body>
</html>

```

Fetch Options

```

/*! normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css */
/* Document
===== */
/**

```

Server-Side APIs

Fetch Options

Open the console in your browser to see the responses!

```

fetch('https://api.github.com/repos/nodejs/node/issues?per_page=5', {
  method: 'GET', //GET is the default.
  credentials: 'same-origin', // include, *same-origin, omit
  redirect: 'follow', // manual, *follow, error
})
.then(function (response) {
  return response.json();
})
.then(function (data) {
  console.log(data);
});

```

```

▼ Array(5)
  ▼ 0:
    active_lock_reason: null
    assignee: null
    assignees: []
    author_association: "NONE"
    body: "***Is your feature request related to a problem? Please describe.***\r\nI found out that using the `withFileTypes: true` option on `fs.readdirSync` could return information of whether a file is a director...
    closed_at: null
    comments: 0
    comments_url: "https://api.github.com/repos/nodejs/node/issues/38809/comments"
    created_at: "2021-05-25T13:54:11Z"
    events_url: "https://api.github.com/repos/nodejs/node/issues/38809/events"
    html_url: "https://github.com/nodejs/node/issues/38809"
    id: 900847137
    labels: (2) [{"-}, {-}]
    labels_url: "https://api.github.com/repos/nodejs/node/issues/38809/labels{/name}"
    locked: false
    milestone: null
    node_id: "MDU6SXNzdWU5MDA4NDcxMzc="
    number: 38809
    performed_via_github_app: null
    repository_url: "https://api.github.com/repos/nodejs/node"
    state: "open"
    title: "[Feature Request]: add `isHiddenFile` property on `fs` dirent"
    updated_at: "2021-05-25T14:01:03Z"
    url: "https://api.github.com/repos/nodejs/node/issues/38809"
    user: {login: "kimlinjustin", id: 61906684, node_id: "MDQ6VXN1c3R5OTA2Njg0", avatar_url: "https://avatars.githubusercontent.com/u/61906684?v=4", gravatar_id: "", ...}
    __proto__: Object
  1: {url: "https://api.github.com/repos/nodejs/node/issues/38807", repository_url: "https://api.github.com/repos/nodejs/node", labels_url: "https://api.github.com/repos/nodejs/node/issues/38807/labels{/name}", c...
  2: {url: "https://api.github.com/repos/nodejs/node/issues/38805", repository_url: "https://api.github.com/repos/nodejs/node", labels_url: "https://api.github.com/repos/nodejs/node/issues/38805/labels{/name}", c...
  3: {url: "https://api.github.com/repos/nodejs/node/issues/38803", repository_url: "https://api.github.com/repos/nodejs/node", labels_url: "https://api.github.com/repos/nodejs/node/issues/38803/labels{/name}", c...
  4: {url: "https://api.github.com/repos/nodejs/node/issues/38799", repository_url: "https://api.github.com/repos/nodejs/node", labels_url: "https://api.github.com/repos/nodejs/node/issues/38799/labels{/name}", c...
    length: 5
    __proto__: Array(0)

```

Document.location

The `Document.location` read-only property returns a [Location](#) object, which contains information about the URL of the document and provides methods for changing that URL and loading another URL.

Though `Document.location` is a `read-only` `Location` object, you can also assign a [DOMString](#) to it. This means that you can work with `document.location` as if it were a string in most cases:

```
document.location = 'http://www.example.com' // is a synonym of document.location.href = 'http://www.example.com', if you assign another string to it, browser will load the website you assigned.
```

To retrieve just the URL as a string, the read-only `document.URL` property can also be used.

If the current document is not in a browsing context, the returned value is `null`.

Syntax

```
locationObj = document.location
document.location = 'http://www.mozilla.org' // Equivalent to document.location.href = 'http://www.mozilla.org'
```

Examples

```
console.log(document.location);
// Prints a Location object to the console
```

Specifications

Specification	Status	Comment
HTML Living Standard The definition of "Document.location" in that specification. ↗	Living Standard	No change from HTML5 . ↗
HTML5 The definition of "Document.location" in that specification. ↗	Recommendation	Initial definition.

Browser compatibility

[Report problems with this compatibility data on GitHub](#) [↗](#)

location	Desktop						Mobile					
	Chrome	Edge	Firefox	Internet Explorer	Opera	Safari	WebView Android	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet
	1	12	1	4	3	1	1	18	4	10.1	1	1.0

```

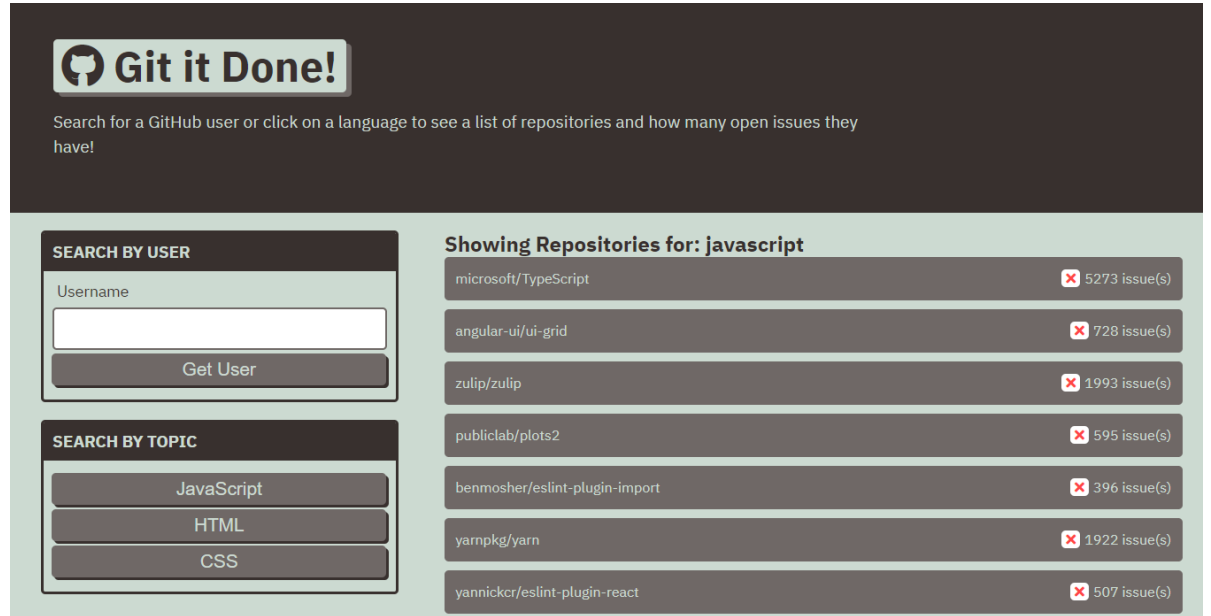
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link href="https://fonts.googleapis.com/css?family=IBM+Plex+Sans:400,400i,700&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.11.2/css/all.min.css">
  <link rel="stylesheet" href="/assets/css/style.css" />
  <title>Git it Done</title>
</head>
<body class="flex-column min-100-vh">
  <header class="hero">
    <h1 class="app-title">
      <i class="fab fa-github"></i> Git it Done!
    </h1>
    <p>Search for a GitHub user or click on a language to see a list of repositories and how many open issues they have!
    </p>
  </header>

  <main class="flex-row justify-space-between">
    <div class="col-12 col-md-4">
      <div class="card">
        <h3 class="card-header text-uppercase">Search By User</h3>
        <form id="user-form" class="card-body">
          <label class="form-label" for="username">Username</label>
          <input name="username" id="username" type="text" autofocus="true" class="form-input" />
          <button type="submit" class="btn">Get User</button>
        </form>
      </div>
      <div class="card">
        <h3 class="card-header text-uppercase">Search By Topic</h3>
        <div id="language-buttons" class="card-body">
          <button data-language="javascript" class="btn">JavaScript</button>
          <button data-language="html" class="btn">HTML</button>
          <button data-language="css" class="btn">CSS</button>
        </div>
      </div>
    </div>
    <div class="col-12 col-md-8">
      <h2 class="subtitle">Showing Repositories for: <span id="repo-search-term"></span></h2>
      <div id="repos-container" class="list-group"></div>
    </div>
  </main>
<script src="/assets/js/homepage.js"></script>
</body>
</html>

```

Review
CSS
Attached

Review pt.1



```

var userFormEl = document.querySelector("#user-form");
var languageButtonsEl = document.querySelector("#language-buttons");
var nameInputEl = document.querySelector("#username");
var repoContainerEl = document.querySelector("#repos-container");
var repoSearchTerm = document.querySelector("#repo-search-term");
var formSubmitHandler = function (event) {
  event.preventDefault();
  var username = nameInputEl.value.trim();
  if (username) {
    getUserRepos(username);
    repoContainerEl.textContent = "";
    nameInputEl.value = "";
  } else {
    alert('Please enter a GitHub username');
  }
};
var buttonClickHandler = function (event) {
  var language = event.target.getAttribute('data-language');
  if (language) {
    getFeaturedRepos(language);
    repoContainerEl.textContent = "";
  }
};

```

```

var getUserRepos = function (user) {
  var apiUrl = 'https://api.github.com/users/' + user + '/repos';
  fetch(apiUrl)
    .then(function (response) {
      if (response.ok) {
        response.json().then(function (data) {
          displayRepos(data, user);
        });
      } else {
        alert('Error: ' + response.statusText);
      }
    })
    .catch(function (error) {
      alert('Unable to connect to GitHub');
    });
};
var getFeaturedRepos = function (language) {
  var apiUrl = 'https://api.github.com/search/repositories?
q=' + language + '+is:featured&sort=help-wanted-issues';
  fetch(apiUrl).then(function (response) {
    if (response.ok) {
      response.json().then(function (data) {
        displayRepos(data.items, language);
      });
    } else {
      alert('Error: ' + response.statusText);
    }
  });
};

```

```

var displayRepos = function (repos, searchTerm) {
  if (repos.length === 0) {
    repoContainerEl.textContent = 'No repositories found.';
    return;
  }
  repoSearchTerm.textContent = searchTerm;
  for (var i = 0; i < repos.length; i++) {
    var repoName = repos[i].owner.login + '/' + repos[i].name;
    var repoEl = document.createElement('div');
    repoEl.classList = 'list-item flex-row justify-space-between align-center';
    var titleEl = document.createElement('span');
    titleEl.textContent = repoName;
    repoEl.appendChild(titleEl);
    var statusEl = document.createElement('span');
    statusEl.classList = 'flex-row align-center';
    if (repos[i].open_issues_count > 0) {
      statusEl.innerHTML =
        "<i class='fas fa-times status-icon icon-danger'></i>" + repos
[i].open_issues_count + ' issue(s)';
    } else {
      statusEl.innerHTML = "<i class='fas fa-check-square status-icon icon-
success'></i>";
    }
    repoEl.appendChild(statusEl);
    repoContainerEl.appendChild(repoEl);
  }
};
userFormEl.addEventListener('submit', formSubmitHandler);
languageButtonsEl.addEventListener('click', buttonClickHandler);

```

Review pt. 2

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link href="https://fonts.googleapis.com/css?family=IBM+Plex+Sans:400,400i,700&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="/assets/css/style.css" />
  <title>Git it Done</title>
</head>
<body class="flex-column min-100-vh">
  <header class="hero">
    <h1 class="app-title">Showing Issues for: <span id="repo-name"></span></h1>

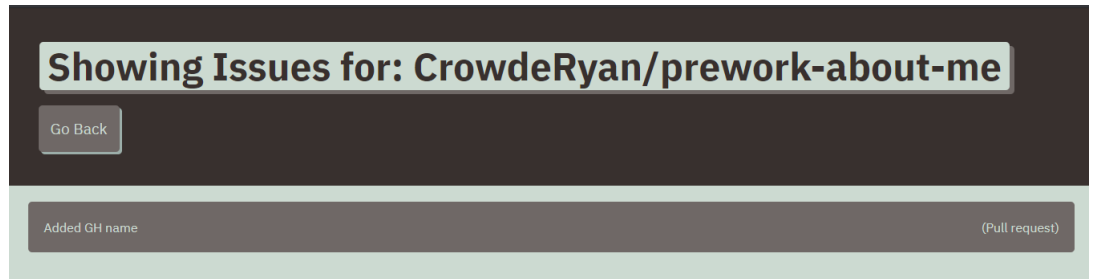
    <div>
      <a href="/index.html" class="btn-back">
        Go Back
      </a>
    </div>
  </header>

  <main>

    <div id="issues-container" class="list-group"></div>

    <div id="limit-warning"></div>
  </main>
<script src="/assets/js/single.js"></script>
</body>
</html>
```

Review
CSS
Attached



```
var repoNameEl = document.querySelector('#repo-name');
var issueContainerEl = document.querySelector('#issues-container');
var limitWarningEl = document.querySelector('#limit-warning');
var getRepoName = function () {
  var queryString = document.location.search;
  var repoName = queryString.split('=')[1];
  if (repoName) {
    repoNameEl.textContent = repoName;
    getRepoIssues(repoName);
  } else {
    document.location.replace('/index.html');
  }
};
var getRepoIssues = function (repo) {
  var apiUrl = 'https://api.github.com/repos/' + repo + '/issues?direction=asc';
  fetch(apiUrl).then(function (response) {
    if (response.ok) {
      response.json().then(function (data) {
        displayIssues(data);
        if (response.headers.get('Link')) {
          displayWarning(repo);
        }
      });
    } else {
      document.location.replace('/index.html');
    }
  });
};
```

```
var displayIssues = function (issues) {
  if (issues.length === 0) {
    issueContainerEl.textContent = 'This repo has no open issues!';
    return;
  }
  for (var i = 0; i < issues.length; i++) {
    var issueEl = document.createElement('a');
    issueEl.classList =
      'list-item flex-row justify-space-between align-center';
    issueEl.setAttribute('href', issues[i].html_url);
    issueEl.setAttribute('target', '_blank');
    var titleEl = document.createElement('span');
    titleEl.textContent = issues[i].title;
    issueEl.appendChild(titleEl);
    var typeEl = document.createElement('span');
    if (issues[i].pull_request) {
      typeEl.textContent = '(Pull request)';
    } else {
      typeEl.textContent = '(Issue)';
    }
    issueEl.appendChild(typeEl);
    issueContainerEl.appendChild(issueEl);
  }
};
```

```
var displayWarning = function (repo) {
  limitWarningEl.textContent = 'To see more than 30 issues, visit ';
  var linkEl = document.createElement('a');
  linkEl.textContent = 'GitHub.com';
  linkEl.setAttribute('href', 'https://github.com/' + repo + '/issues');
  linkEl.setAttribute('target', '_blank');
  limitWarningEl.appendChild(linkEl);
};

getRepoName();
```

Git Collaborate

GitHub Repository Setup for Collaborative Projects

In this activity, we will walk through how to set up GitHub repositories for a collaborative project. When multiple developers work in a single GitHub repository, certain settings and rules need to be put in place to ensure the integrity of the codebase. These rules also ensure that no one can make changes to the main code in the repository without another developer reviewing first.

> ****Important:**** Because this is a group-based application, only one person in the group needs to make the repository. That person will invite the other developers to join the repo as a collaborator once they have everything set up.

Instructions

Create the Repository and Set Up .gitignore File

- * Like with all projects, we need to start by making a repository. Navigate to [GitHub](https://github.com) and create a new repository for your project. Ensure that the option to add a 'README.md' file is checked.

- * Once the repository has been created, clone it to your machine using 'git clone <repository-name>' and open the entire application in VS Code.

- * At the root of the application, use VS Code or the command line to create a file named '.gitignore'. This file will store a list of files or folders that may end up in the application but that we don't want to keep on GitHub.

- * Add the following line to the newly created '.gitignore' file:

```
...
.DS_Store
...
```

- * The '.DS_Store' file is a file unique to macOS that the operating system automatically creates to internally store a directory's attributes, so Windows users have likely not encountered this before. This file has no bearing on the actual project code, so it's best to always add it to the '.gitignore' file to ensure that it never makes it to the actual GitHub repository.

- * Now save your code, commit it, and push it to the 'main' branch using the following commands:

```
```bash
git add -A
git commit -m 'add gitignore file'
git push origin main
```
```

- * Great, now the application is ready for other developers! Before we add those developers, however, let's add some protective measures to the repo on GitHub.

Protect the main Branch

When you work alone on a project, it's a safe assumption that any code that is pushed to the 'main' branch will come from one course, so it's easy to keep track of who added what code and whether it works or not. When working with others, however, certain rules need to be put in place to keep every developer in sync and in agreement on what code is ready for production. Luckily, GitHub has some features we can add to a repo to stay in sync more easily. Let's go and add them now:

- * Navigate to the repository on GitHub and select the Settings tab for the repo.

The GitHub repository's settings page shows options to set for the repo](./Images/01-repo-settings.png)

- * Once there, look in the left column's menu and select the Branches option. Then find the Add Rule button in the main content on the right, and select it.

The branch settings page in the repository shows a button to add a rule to a branch.

- * After you select Add Rule, a page titled "Branch protection rule" will appear. On this page, we'll set up the repository so that no code can be pushed to the 'main' branch by any member of the developer team. All code that is to be merged into the 'main' branch MUST be merged through a GitHub pull request. Let's add the following to this page:

- * For the "Branch name pattern" input field, enter the branch 'main'.

- * Next, check the box next to the option for "Require pull request reviews before merging". It will open another set of options asking if we want multiple reviewers before allowing a pull request to be merged, but we can leave it at just one for now.

- * Finally, select the option to "Include administrators". This way, even the repository owner cannot push to 'main' and has to follow all of the same rules as the rest of the developers.

- * Confirm that the correct rules are in place.

[The branch protection rule page shows all of the options in place to protect the main branch

- * Make sure to select the Create button at the bottom of the page. The page will refresh and inform us that the rules have been created.

- * Great! Now the repository's 'main' branch cannot have code added to it without a pull request, and that pull request must be reviewed and approved by at least one other developer on the project. Let's go add those other developers now!

> ****Important:**** We can still work locally in the 'main' branch, though it's not recommended. The limitation we just put in place is that 'git push origin main' will not work anymore. If you have accidentally done work in the local 'main' branch and need to push it, you should checkout a new branch for that updated code and push it to that branch on GitHub instead.

Add Collaborators to the Repository

GitHub allows developers to work together and contribute to a single repository in a few different ways. We will set up the repository so that certain other GitHub users can write data and read data to it. Let's go add these collaborators with the following steps:

- * In the GitHub repository's Settings page, select Manage Access from the menu on the left side. The GitHub repository's settings page shows who has access to read or write to the repository and the option to invite collaborators.

- * Once the modal dialog opens, start typing in the GitHub username of one of your collaborators, and the list should automatically populate that user's entry. The invite a collaborator modal dialog shows the user that needs to be invited to the repo.

- * Select the entry and click the "Add '<github-username>' to '<project-repo-name>'" button.

The manage access page shows there is a pending invitation for a collaborator.

- * Now the user will need to accept the invitation to contribute to the repository. The owner of the repository can also select the small clipboard icon next to the Pending Invite note, copy the invite link, and send it to the collaborator directly.

- * This will only handle one collaborator, so repeat this process for every collaborator on the project.

- * Once all of the collaborators have accepted the invite, it's time to get coding!