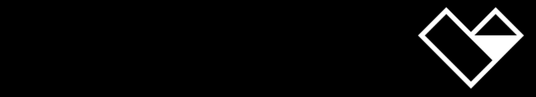
Asynchronous Programming

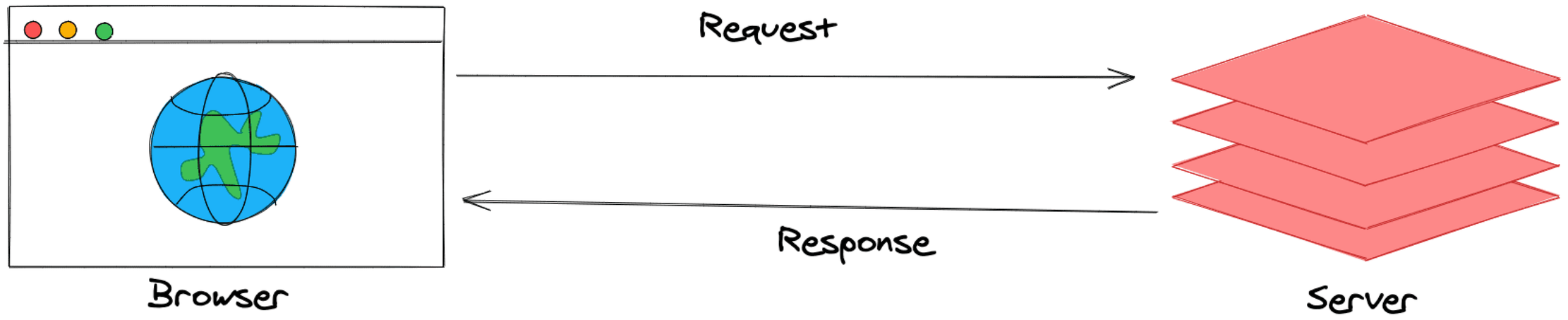
Week 2: Fetch and REST

Fetch



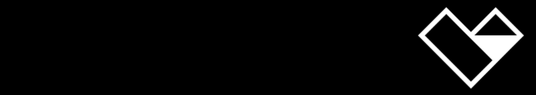
A built-in JavaScript function inside the browser, used to fetch a resource across the network.

https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API

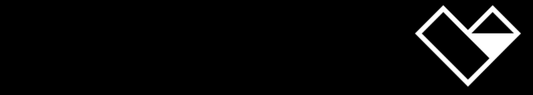

Request
Response
Browser
Server

```
fetch("https://jsonplaceholder.typicode.com/albums") // Pass URL as argument
  .then(response => response.json()) // Transform the *raw* response to JSON
  .then(data => {
    // Do something with your JSON data
    console.log(data);
  });

/*
[
    {userId: 1, id: 1, title: "quidem molestiae enim"},
    {userId: 1, id: 2, title: "sunt qui excepturi placeat culpa"},
]

*/
```

Example time



REST APIs



An architecture for data or resource transfers through the network, accessed via a URL. Currently the most standard design of how the data is transfered on the web

https://restfulapi.net/

https://jsonplaceholder.typicode.com/albums

https://jsonplaceholder.typicode.com/albums

host / base url

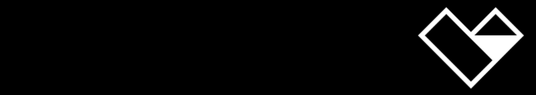
https://jsonplaceholder.typicode.com/albums

host / base url

resource

https://jsonplaceholder.typicode.com/albums/1

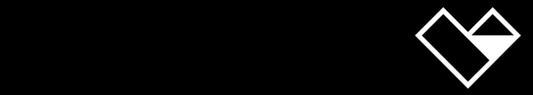
- GET /albums



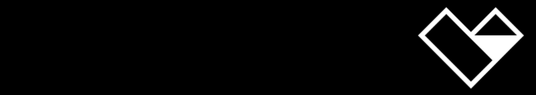
- GET /albums
- POST /albums

- GET /albums
- POST /albums
- PUT /albums/1

- GET /albums
- POST /albums
- PUT /albums/1
- DELETE /albums/1

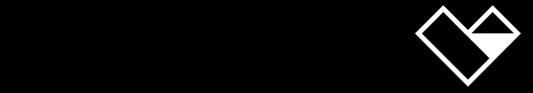


Example time



Excercise time!

Until 13:50

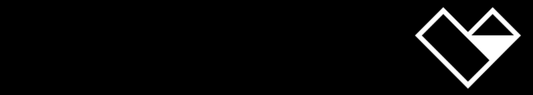


What was challenging for your group?
What is the most helpful thing you've learned?

Break Time!

Back at 14:30

Integrate!



Excercise time!

Until 15:45

What was challenging for your group?
What is the most helpful thing you've learned?

Portfolio

- Individual project
- Build a portfolio to showcase all of your work.
- Use at least one class.
- Use at least one call to the GitHub API

Wednesday Review

- Remember to post your weekly Check-Ins.
- Evening review call will be posted on Slack.