tt(fp)

api's

TODAY

- 9:30: Les over api’s
- 10.00: Hulp om je opweg te helpen (rondje support-groups)
- 12.00: Zelfstudie (concept werken, wiki bijwerken, coden)
- 13:30: Gastcollege Wiki/ReadMe
- 16:00 Standups


TECH-TRACK OLYMPICS

API's & ASYNC

- What's an API? (recap)

- Difference between sync and async
- Callbacks (XMLHttpRequest)
- Promises (fetch)
- Async Await

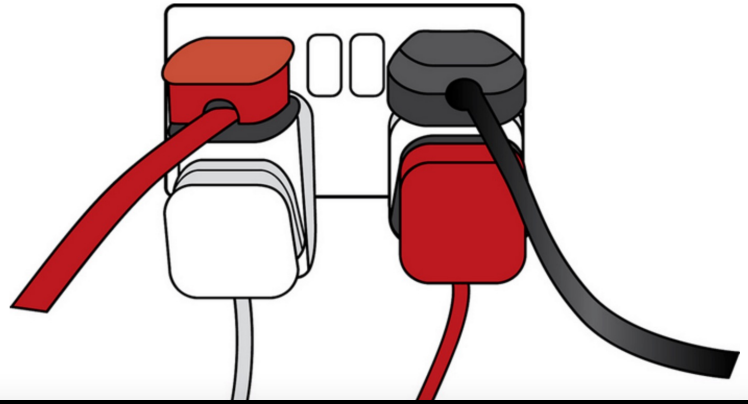
WHAT'S AN API?

I/VI

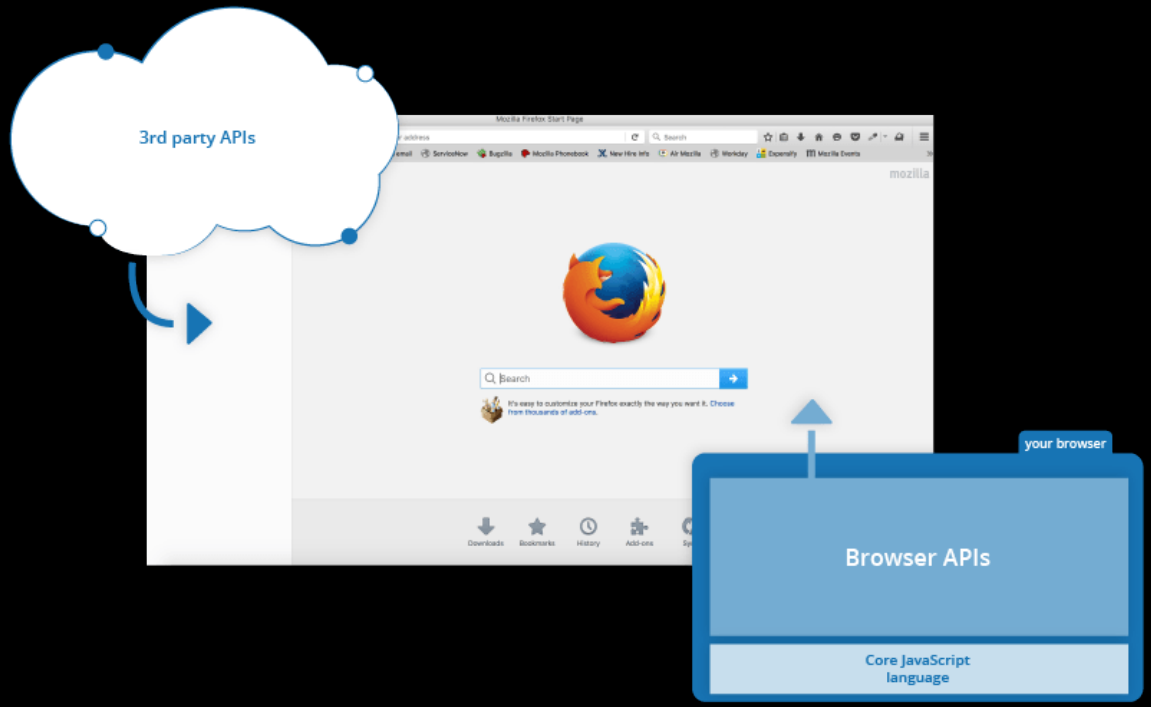
What are APIs?

Application Programming Interfaces (APIs) are constructs made available in programming languages to allow developers to create complex functionality more easily. They abstract more complex code away from you, providing some easier syntax to use in its place.

As a real-world example, think about the electricity supply in your house, apartment, or other dwellings. If you want to use an appliance in your house, you plug it into a plug socket and it works. You don't try to wire it directly into the power supply — to do so would be really inefficient and, if you are not an electrician, difficult and dangerous to attempt.



https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Introduction

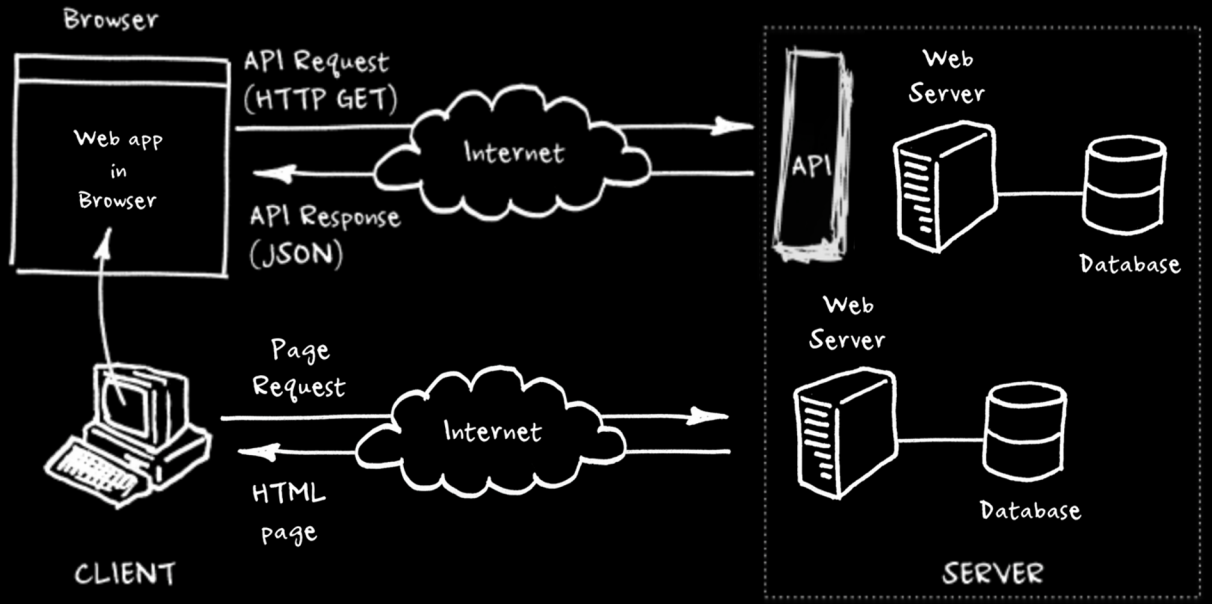


https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Introduction


What are
APIs ?

GRAND SCHEME

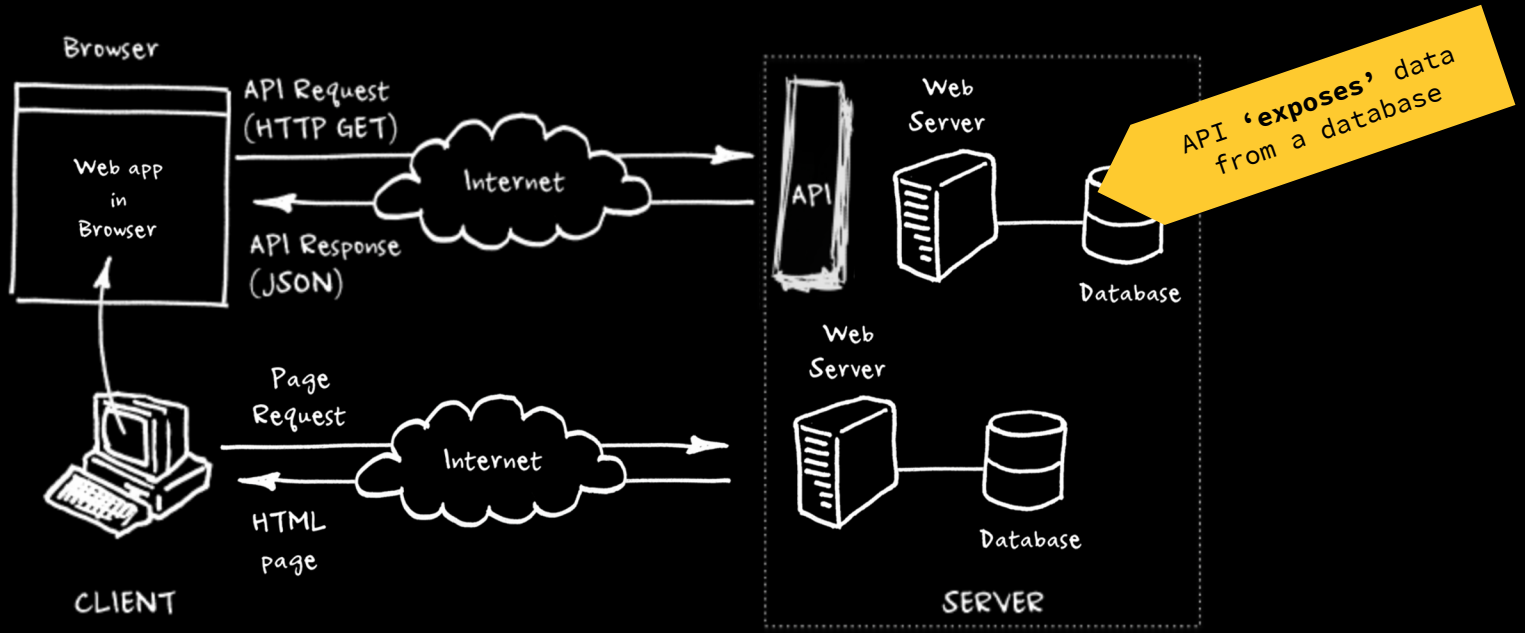
API



http://www.robert-drummond.com/2013/05/08/how-to-build-a-restful-web-api-on-a-raspberry-pi-in-javascript-2/

GRAND SCHEME

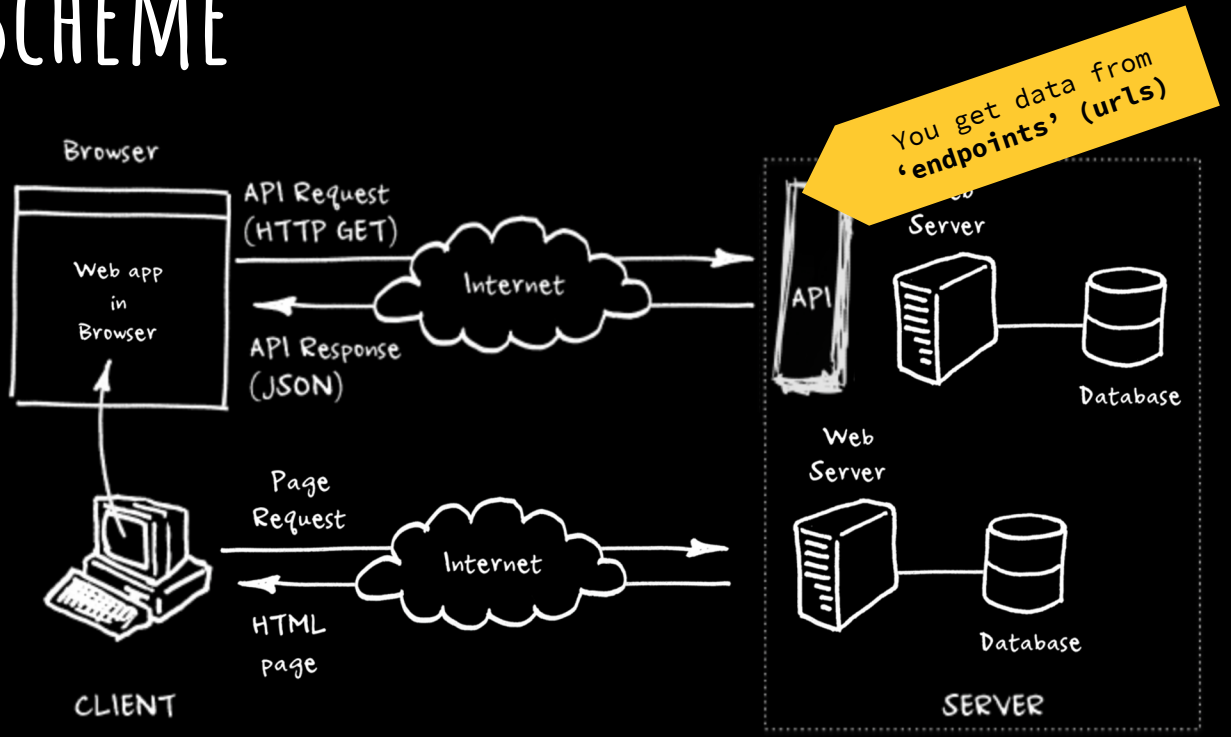
API



http://www.robert-drummond.com/2013/05/08/how-to-build-a-restful-web-api-on-a-raspberry-pi-in-javascript-2/

GRAND SCHEME

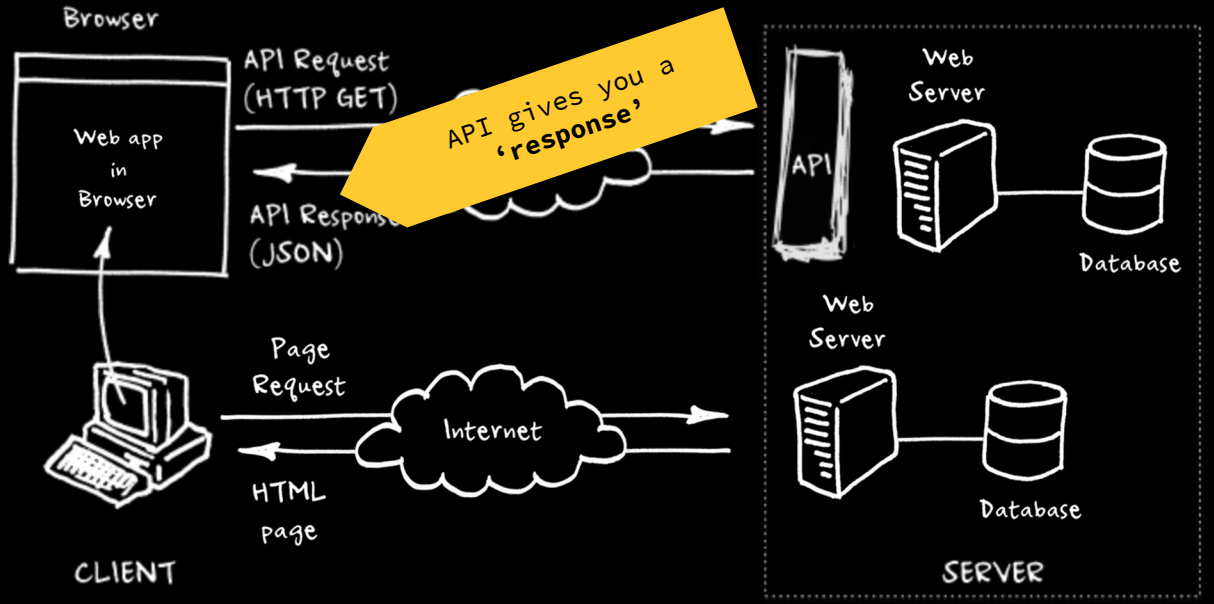
API



http://www.robert-drummond.com/2013/05/08/how-to-build-a-restful-web-api-on-a-raspberry-pi-in-javascript-2/

GRAND SCHEME

API



http://www.robert-drummond.com/2013/05/08/how-to-build-a-restful-web-api-on-a-raspberry-pi-in-javascript-2/

API

DEFINITION

An API is an application programming interface. **It is a set of rules that allow programs to talk to each other.** The developer creates the API on the server and allows the client to talk to it.

Smashing - Understanding REST API’s

API

REST

REST determines how the API looks like. It stands for “Representational State Transfer”. It is a set of rules that developers follow when they create their API.

Smashing - Understanding REST API’s

Open Data Parkeren: PARKEERGEBIED

Parkeren

Data bekijken | Visualiseren en ontdekken | Exporteer | API | ...

Deze tabel legt een koppeling tussen de gebieden zoals de
de gebieden zoals deze voor Open Data Parkeren volgens
gepubliceerd worden.

Krijg toegang tot deze Dataset via SODA API

De Socrata Open Data API (SODA) voorziet een programmatische toegang tot deze dataset en de mogelijkheid om gegevens te filteren, op te vragen en te combineren.

API-documenten | Ontwikkelingsport

Endpoint and data format

API-eindpunt

https://opendata.rdw.nl/resource/mz4f-59 | JSON | Kopiëren

Blokkeren

Betreffende deze dataset

Bijgewerkt
25 oktober 2020

Laatst bijgewerkt op
25 oktober 2020

Metadata laatst bijgewerkt op
25 oktober 2020

Gemaakt op
28 oktober 2014

Weergaves
4.800

Downloads
6.776

Data voorzien door
(geen)

Eigenaar dataset
Open data team RDW

Licentie

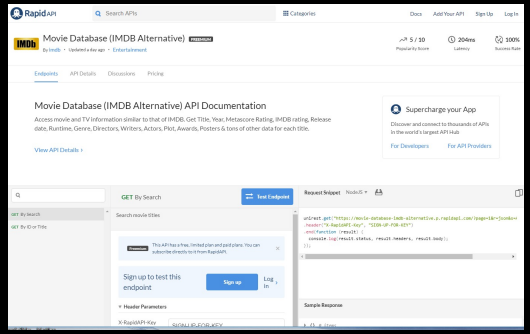
Licentie	

Onderwerpen

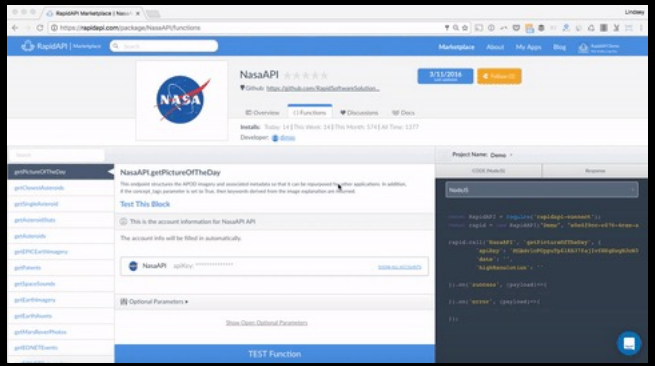
Categorie	Parkeren
Tags	parkeergebied, parkeren

musixmatch

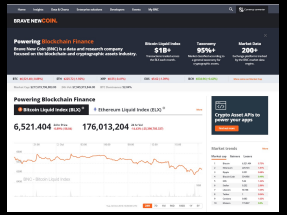

TheCocktailDB
Home
API
Forum
Search
Welcome to TheCocktailDB
An open, crowd-sourced database of drinks and cocktails from around the world.
We also offer a free JSON API for anyone wanting to use it.
If you like the site, please consider supporting us on Patreon by clicking the link below...
SUPPORT US ON PATREON
Click to Support: Currently 21 supporters
LATEST NEWS: We have heard the sad news that the Absolut cocktail API has closed down. Luckily we think we are a great alternative!
NEWS: New darker purple design for the refreshed website, more responsive images and a new homepage.
NEWS: Check our the newly redesigned search and alphabetic filter pages.
Search for a cocktail...
Total Drinks: 578
Total Ingredients: 471
Drink Images: 578
Popular Drinks

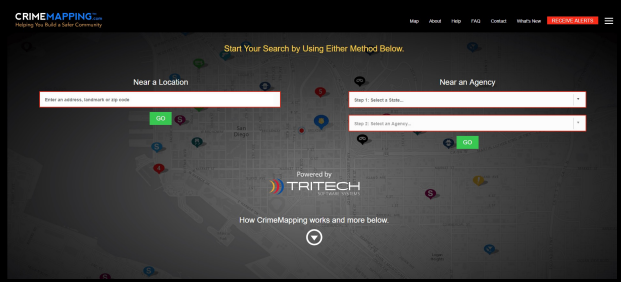

RapidAPI
Search APIs
Categories
Docs
Add Your API
Sign Up
Log In
Movie Database (IMDB Alternative)
Endpoints
API Details
Discussions
Pricing
Movie Database (IMDB Alternative) API Documentation
Access movie and TV information similar to that of IMDB. Get Title, Year, Metascore Rating, IMDB rating, Release date, Runtime, Genre, Directors, Writers, Actors, Plot, Awards, Posters & tons of other data for each title.
View API Details
Supercharge your App
Discover and connect to thousands of APIs in the world's largest API Hub
For Developers
For API Providers
GET By Search
Test Endpoint
Request Snippet
Search movie titles
Sign up to test this endpoint
Sign up
Log in
Header Parameters
Sample Response


skyscanner


RapidAPI Marketplace
Marketplace
About
My Apps
Blog
NasaAPI
NASA
3/11/2016
Overview
Functions
Discussions
Docs
NasaAPI.getPictureOfTheDay
Test This Block
The account information for NasaAPI API
The account info will be filled in automatically.
Optional Parameters
TEST Function


WORDSAPI
An API for the English Language.
Learn More
Try It


BRAVE NEW COIN
Powering Blockchain Finance
Powering Blockchain Finance
Bitcoin Liquid Index (BLX)
Ethereum Liquid Index (ELX)
6,521.404
176,013.204
Crypto Asset APIs to power your apps


CRIMEMAPPING
Map
About
Help
FAQ
Contact
What's New
RECEIVE ALERTS
Start Your Search by Using Either Method Below.
Near a Location
Enter an address, landmark or zip code
GO
Near an Agency
Step 1: Select a State...
Step 2: Select an Agency...
GO
Powered by
TRITECH
How CrimeMapping works and more below.

FUNCTIONAL PROGRAMMING

assignment: get async data from an external API and structure and model that data

To do:

- **Create a concept with an external API**
- **Clean and transform data (chains)**
- **Get async / sync data from endpoints (tomorrow)**

SYNC VS ASYNC

II/VI

SYNC VS ASYNC

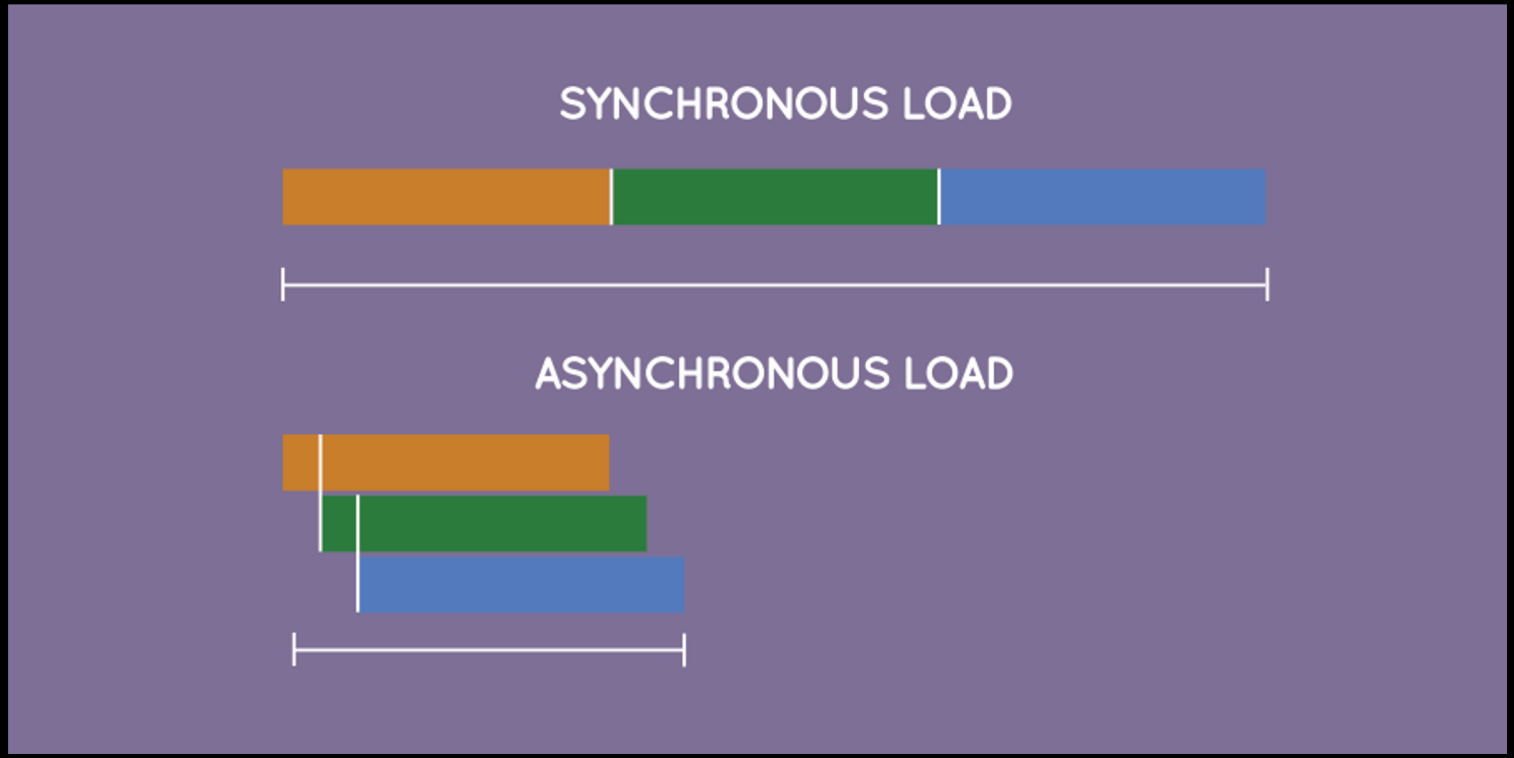
WHY?

Getting data from a resource (API) takes time. It needs to **fetch** the resource, parse it etc. But also, what if the data isn't available (no internet connection e.g.) how should *errors* be handled?

SYNC VS ASYNC

Only one thing can happen at a time, on a single main thread, and everything else is *blocked* until an operation completes.

MDN - Introducing asynchronous JavaScript


SYNCHRONOUS LOAD
ASYNCHRONOUS LOAD

SYNC VS ASYNC

Many Web API features now use **asynchronous code to run,** especially those that access or *fetch some kind of resource from an external* device, such as fetching a file from the network, accessing a database and returning data from it.

MDN - Introducing asynchronous JavaScript

CALLBACKS

III/VI

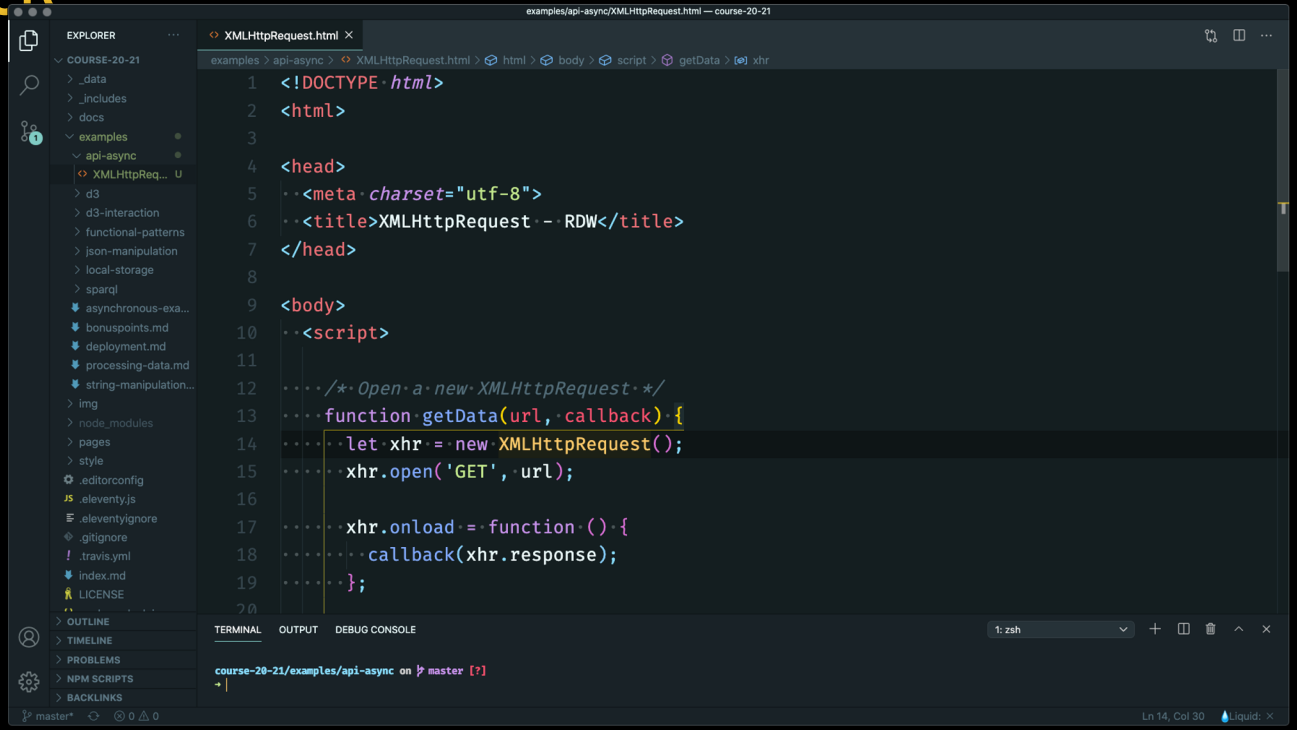
CALLBACK

DEFINITION

Async callbacks are functions that are specified as arguments when calling a function which *will start executing code in the background.* **When the background code finishes running, it calls the callback function.**

CALLBACK

XMLHTTPREQUEST



XMLHttpRequest Example

PROMISES

IV/VI

PROMISES

DEFINITION

The Promise object represents the eventual completion (or failure) of **an asynchronous operation and its resulting value.**

PROMISES

DEFINITION

A promise can only succeed or fail once. It cannot succeed or fail twice, neither can it switch from success to failure or vice versa. A pending promise can either be **fulfilled with a value or rejected with a reason (error).**

WEB.DEV - Promises

PROMISES

CHAINING

A common need is to execute two or more asynchronous operations back to back, where each subsequent operation starts when the previous operation succeeds, with the result from the previous step. We accomplish this by creating a **promise chain.**

WEB.DEV - Promises

PROMISES

VERSUS

Callback ‘hell’ (christmas tree)

```
doSomething(function(result) {
  doSomethingElse(result, function(newResult) {
    doThirdThing(newResult, function(finalResult) {
      console.log('Got the final result: ' + finalResult);
    }, failureCallback);
  }, failureCallback);
}, failureCallback);
```

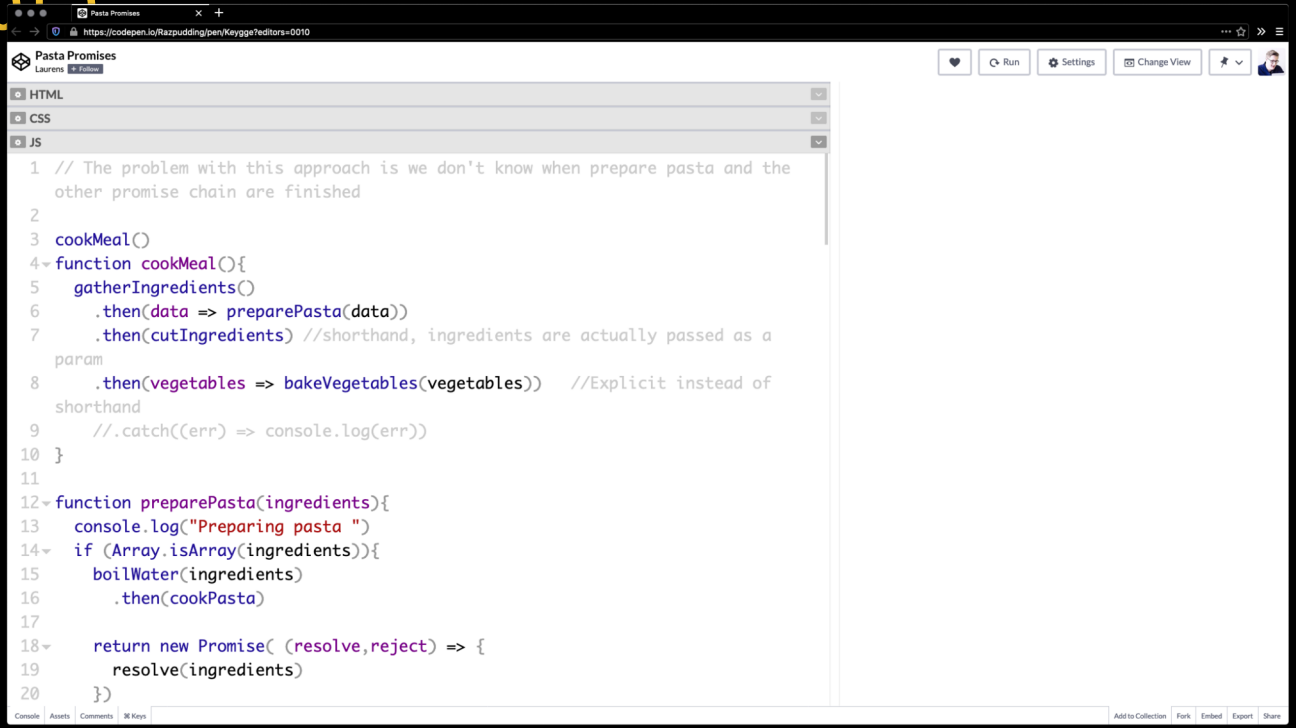
Promise chain

```
doSomething()
.then(function(result) {
  return doSomethingElse(result);
})
.then(function(newResult) {
  return doThirdThing(newResult);
})
.then(function(finalResult) {
  console.log('Got the final result: ' + finalResult);
})
.catch(failureCallback);
```

WEB.DEV - Promises

PROMISES

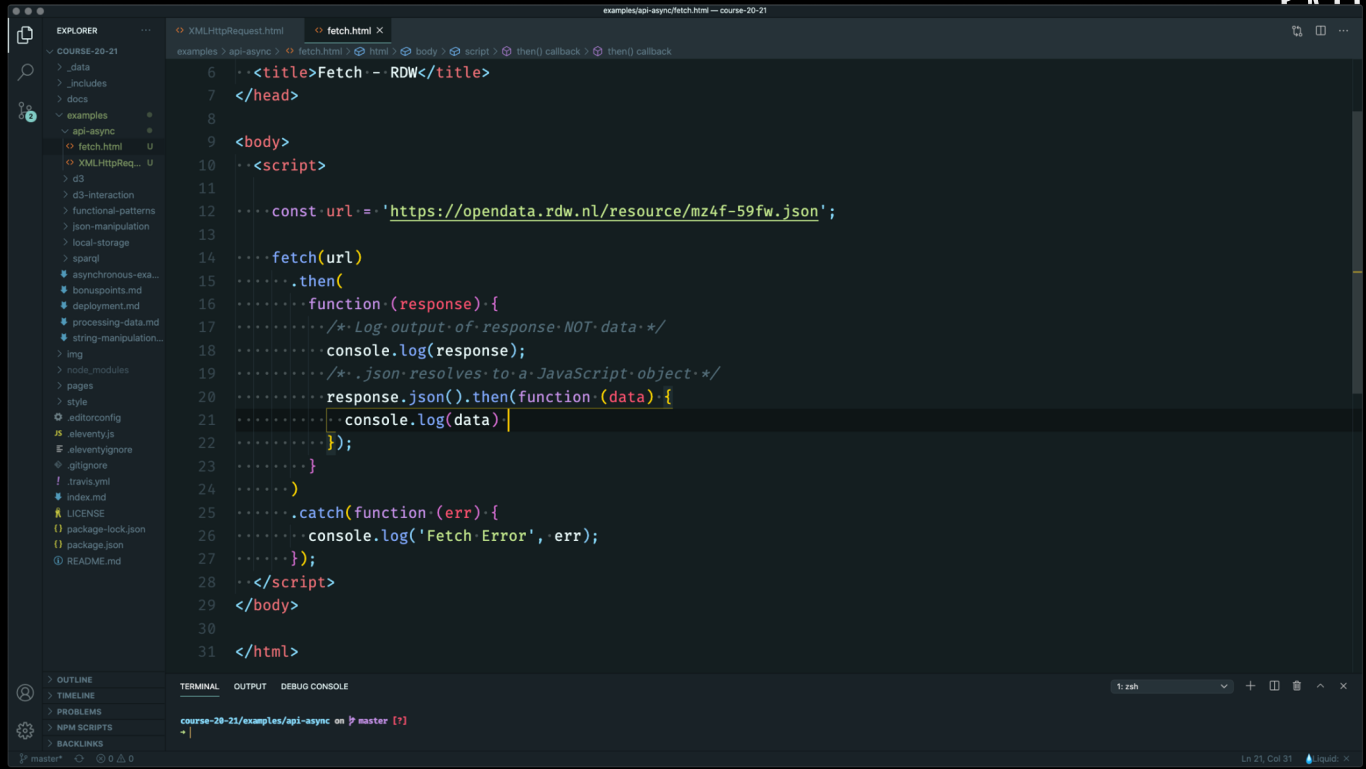
PROMISES



[Pasta Promise Example]

FETCH

PROMISES



Fetch Example

ASYNC AWAIT

VI/VI

ASYNC/AWAIT

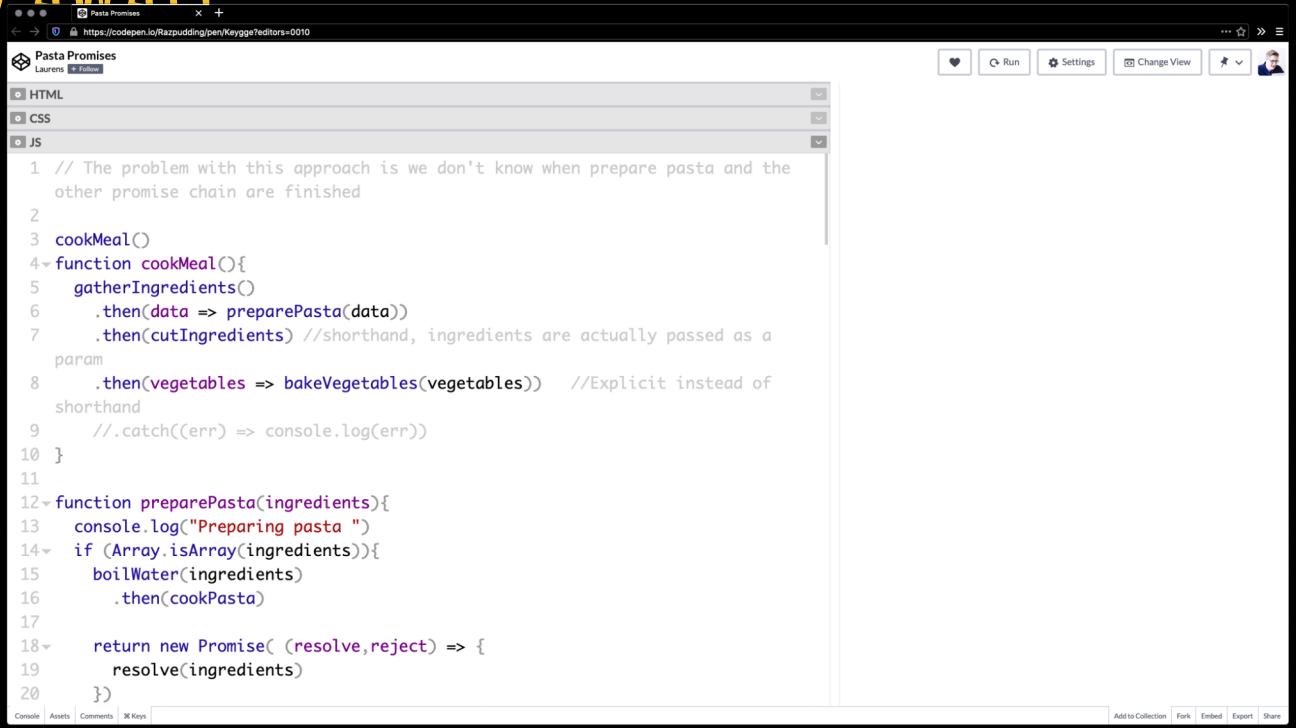
CHAINING

An async function is a function that knows how to e**xpect the possibility of the await keyword** being used to invoke asynchronous code. Act as *syntactic sugar* on top of promises, making asynchronous code easier to write and to read afterwards.

MDN - Making asynchronous programming

ASYNC/AWAIT

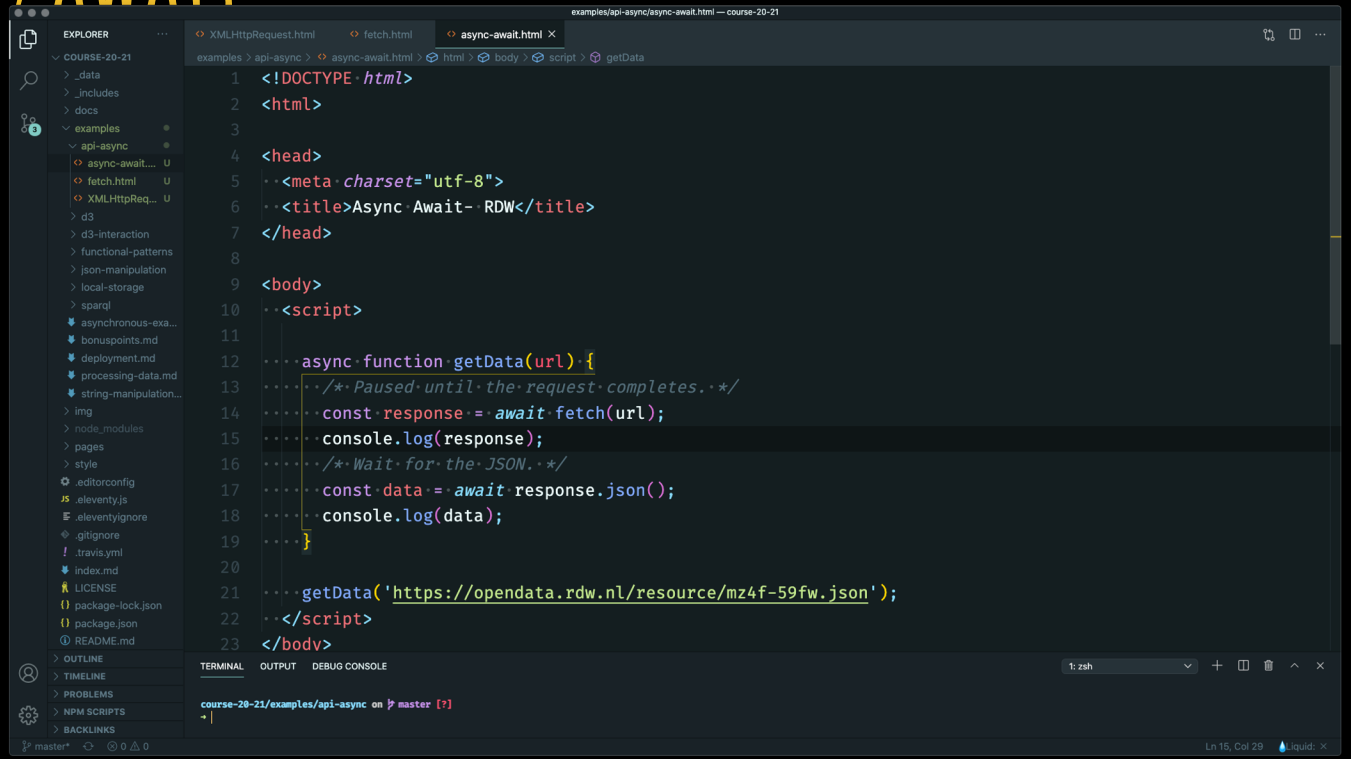
PROMISES



[Pasta Async/Await Example]

ASYNC/AWAIT

PROMISES

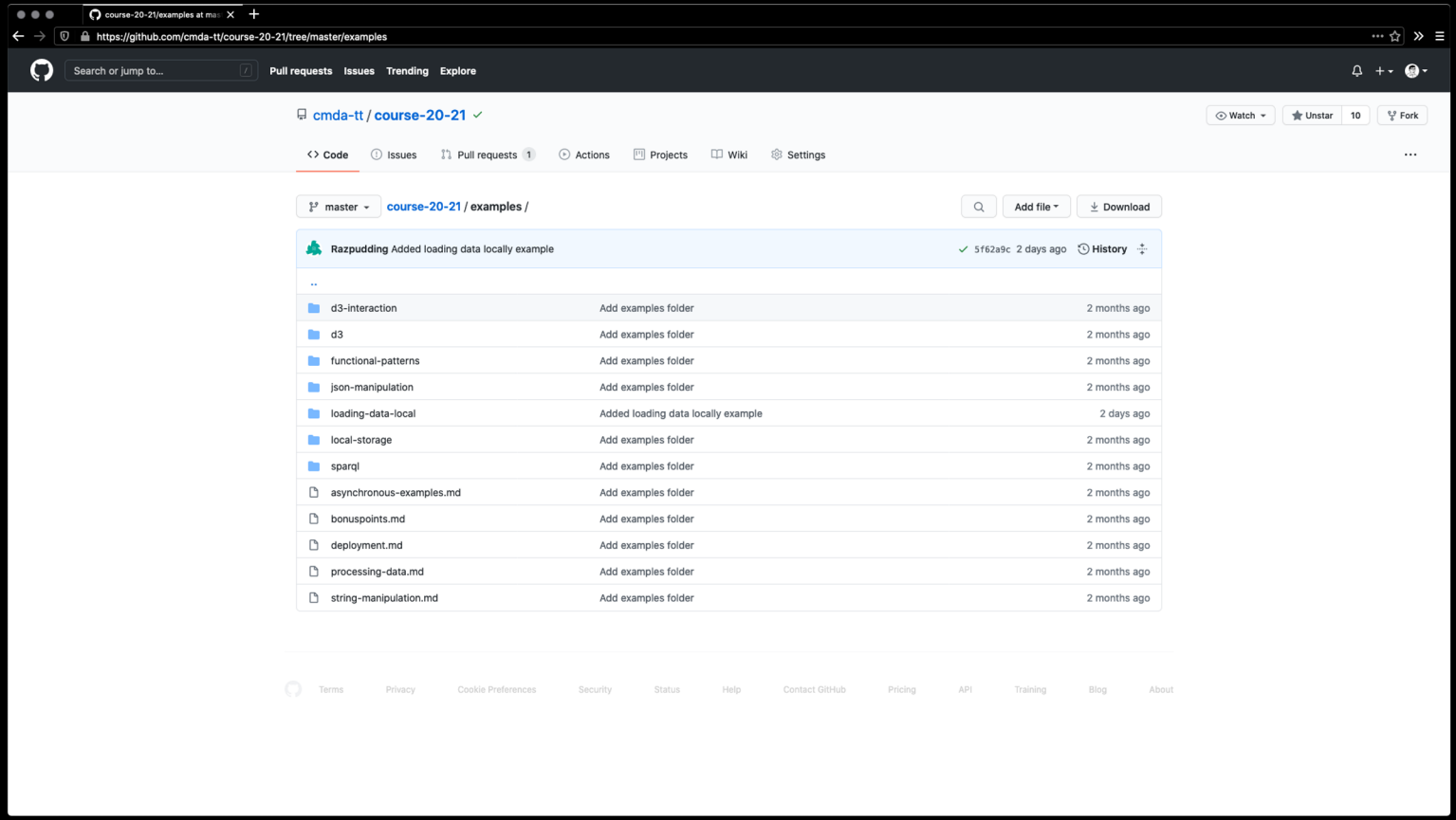


Async Await Example

Error handling

With either method you choose, make sure you handle errors.

- .catch
- if statements (status codes)
- throw



github.com/cmda-tt/course-20-21/examples

EXIT;