# Software Development for Web and Mobile

Pepe García

2020-04-20

# Software Development for Web and Mobile

# Plan for today

Today we will:

- Learn how to communicate with http services

- Do some exercises in class

# Fetch

**fetch** is the way of calling HTTP services from Javascript.

```
fetch("url"); //done!
```

# Fetch

We can customize our request using the second parameter:

```
fetch(
    "url",
    {
        method: "POST",
        headers: {},
        body: "this is the body"
    }
);
```

# Fetch

but, how do we use the data returned from the server?

let's open the console and see what does the following snippet return.

```
let a = fetch("http://google.com");
```

# Promises

Promises are the solution used in JS for when we don't want to **block the program** while a long running task is made.

By using promises, we create **asynchronous** code.

# Promises

**fetch** uses **Promises** to work asynchronously.

**Promises** can be **pending**, **fulfilled** or **rejected**.

- **pending**: the promise hasn't finished yet.
- **fulfilled**: the promise finished correctly.
- **rejected**: there was an error in the promise

# Promises

# Promises

We use the methods **then** and **catch** to handle the different outcomes of the promise (**fulfilled** and **rejected** respectively)

```
let a = fetch("https://google.com")

a.then((result) => console.log("the promise is fulfilled, and returned" + resul

a.catch((error) => console.log("the promise failed with " + error)
```

# Back to fetch

To get the JSON response from fetch we need to use promise's **then** method:

```
fetch("http://api.open-notify.org/iss-now.json")
  .then(data => data.json())
  .then(json => console.log(json))
```

# Exercises

# Exercise 2

Let's review **server.py** together

# Exercise 2

Open **exercises.js**