

SYNCHRONOUS CODE

BLOCKING

```
const p = document.querySelector('.p');  
p.textContent = 'My name is Jonas!';  
alert('Text set!');  
p.style.color = 'red';
```

THREAD OF
EXECUTION



Part of execution
context that actually
executes the code in
computer's CPU

SYNCHRONOUS

- Most code is **synchronous**;
- Synchronous code is **executed line by line**;
- Each line of code **waits** for previous line to finish;
- Long-running operations **block** code execution.

problems with synchronous code

ASYNCHRONOUS CODE

CALLBACK WILL
RUN AFTER TIMER

```
const p = document.querySelector('.p');  
setTimeout(function () {  
  p.textContent = 'My name is Jonas!';  
}, 5000);  
p.style.color = 'red';
```

Example: Timer with callback

```
[1, 2, 3].map(v => v + 2);
```

Callback does NOT automatically
make code asynchronous!

Executed after
all other code

THREAD OF
EXECUTION



"BACKGROUND"

(When on this in the
return on Event Loop)

ASYNCHRONOUS

- Asynchronous code is executed **after** a task that runs in the "background" finishes;
- Asynchronous code is **non-blocking**;
- Execution doesn't wait for an asynchronous task to finish its work;
- Callback functions alone do **NOT** make code asynchronous!

Coordinating behavior of a
program over a period of time

example for asynchronous code

ASYNCHRONOUS CODE

Asynchronous

```
const img = document.querySelector('.dog');
img.src = 'dog.jpg';
img.addEventListener('load', function () {
  img.classList.add('fadeIn');
});
p.style.width = '300px';
```

CALLBACK WILL RUN AFTER IMAGE LOADS

Example: Asynchronous image loading with event and callback

Other examples: Geolocation API or AJAX calls

addEventListener does NOT automatically make code asynchronous!

ASYNCHRONOUS

Coordinating behavior of a program over a period of time

THREAD OF EXECUTION

"BACKGROUND"

- Asynchronous code is executed **after** a task that runs in the "background" finishes;
- Asynchronous code is **non-blocking**;
- Execution doesn't wait for an asynchronous task to finish its work;
- Callback functions alone do **NOT** make code asynchronous!

img loading acting as async code

WHAT ARE AJAX CALLS?

AJAX

Asynchronous JavaScript And XML: Allows us to communicate with remote web servers in an **asynchronous way**. With AJAX calls, we can **request data** from web servers dynamically.



WHAT IS AN API?

API

- Application Programming Interface: Piece of software that can be used by another piece of software, in order to allow applications to talk to each other.
- There are many types of APIs in web development:
 - DOM API
 - Geolocation API
 - Own Class API
 - "Online" API
- "Online" API: Application running on a server, that receives requests for data, and sends data back as response.
- We can build our own web APIs (requires back-end development, e.g. with node.js) or use 3rd-party APIs.

Just "API"

There is an API for everything

AJAX

~~XML~~
XML data format



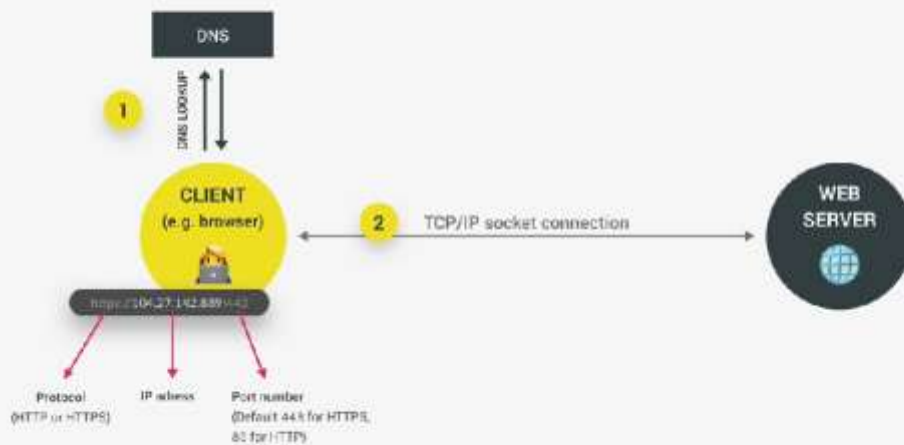
JSON data format

```
{  
  "name": "John",  
  "age": 30,  
  "gender": "male",  
  "email": "john.doe@example.com",  
  "phone": "123-456-7890"  
}
```

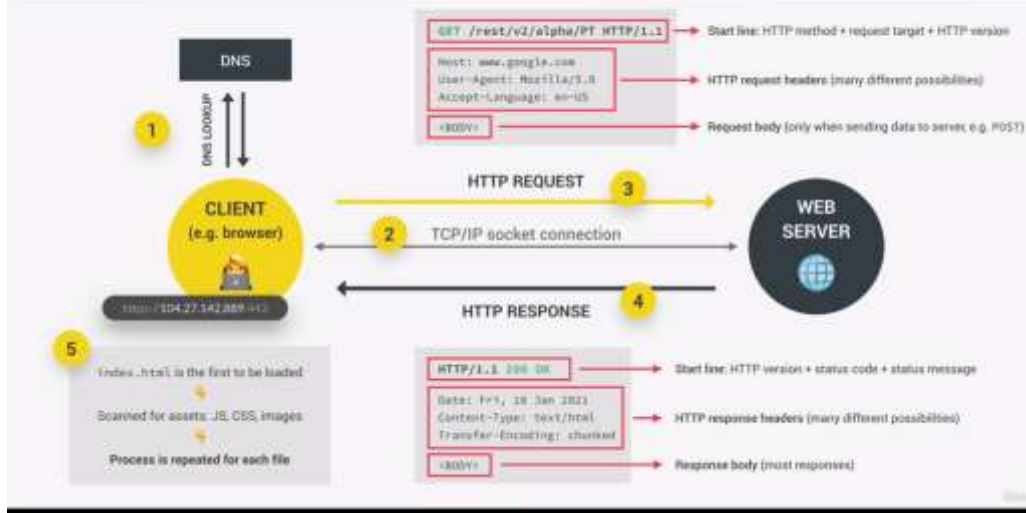
Most popular API data format

- Weather data
- Data about countries
- Flights data
- Currency conversion data
- APIs for sending email or SMS
- Google Maps
- Millions of possibilities...

WHAT HAPPENS WHEN WE ACCESS A WEB SERVER



WHAT HAPPENS WHEN WE ACCESS A WEB SERVER



WHAT ARE PROMISES?

PROMISE

- Promise:** An object that is used as a placeholder for the future result of an asynchronous operation.
Less formal
- Promise:** A container for an asynchronously delivered value.
Less formal
- Promise:** A container for a future value.
Example: Response from AJAX call
- We no longer need to rely on events and callbacks passed into asynchronous functions to handle asynchronous results;
- Instead of nesting callbacks, we can **chain promises** for a sequence of asynchronous operations: **escaping callback hell** 🙌

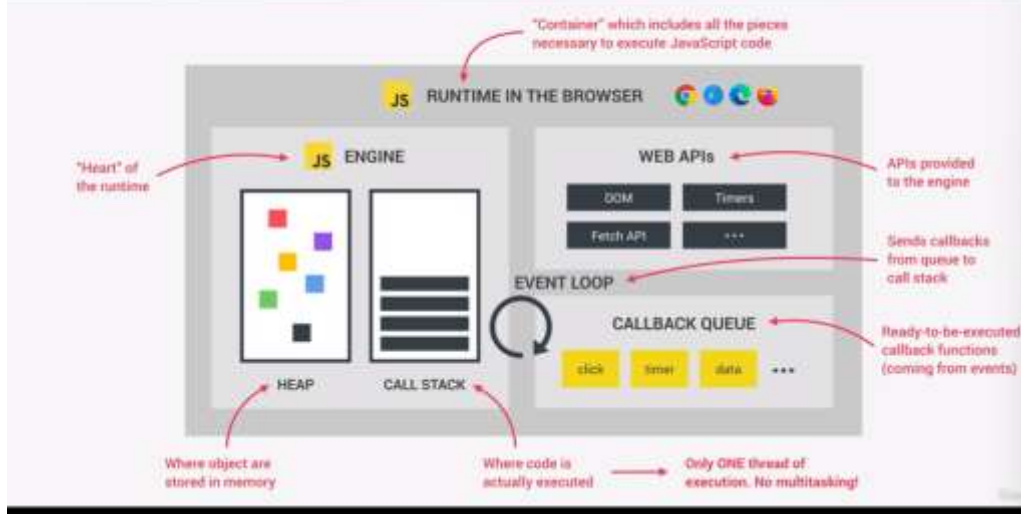
Lottery ticket Promise that I will receive money if I guess correct outcome

- I buy lottery ticket (promise) right now
- ↓
- Lottery draw happens asynchronously
- ↓
- If correct outcome, I receive money, because it was promised

THE PROMISE LIFECYCLE

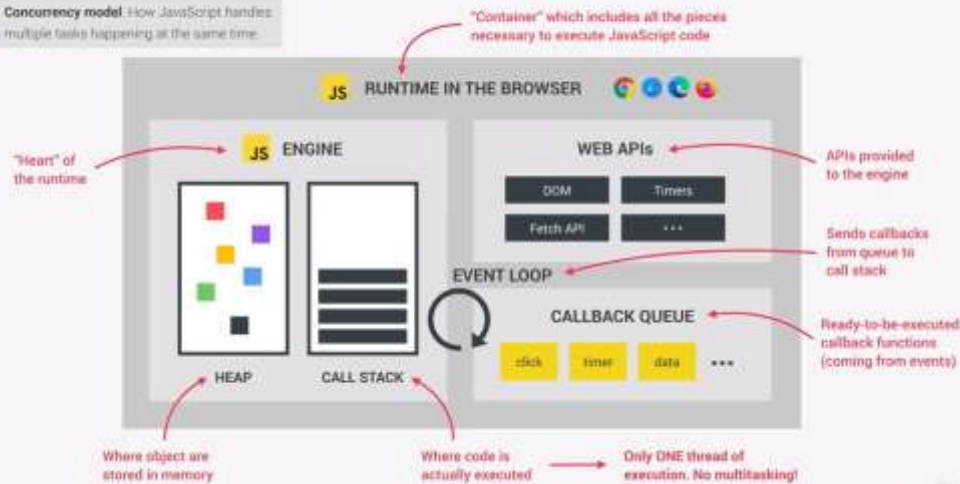


REVIEW: JAVASCRIPT RUNTIME



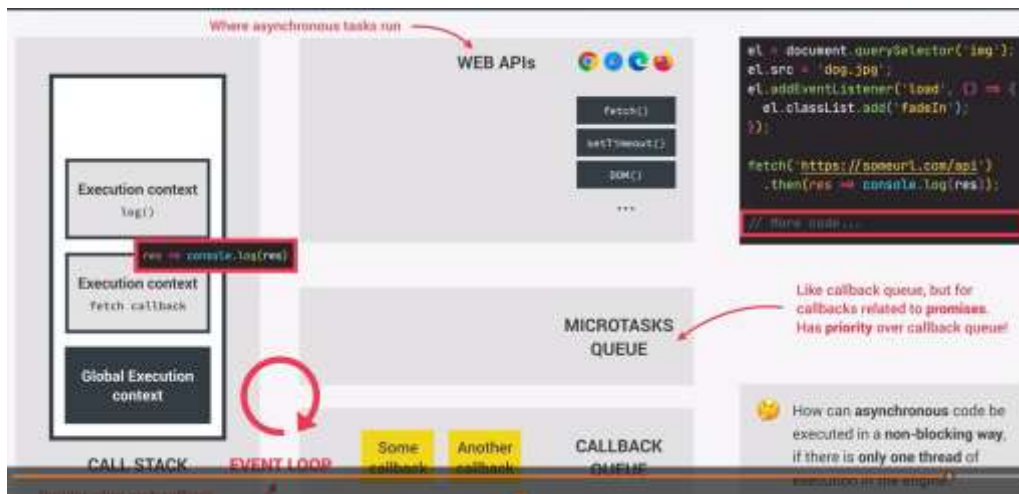
REVIEW: JAVASCRIPT RUNTIME

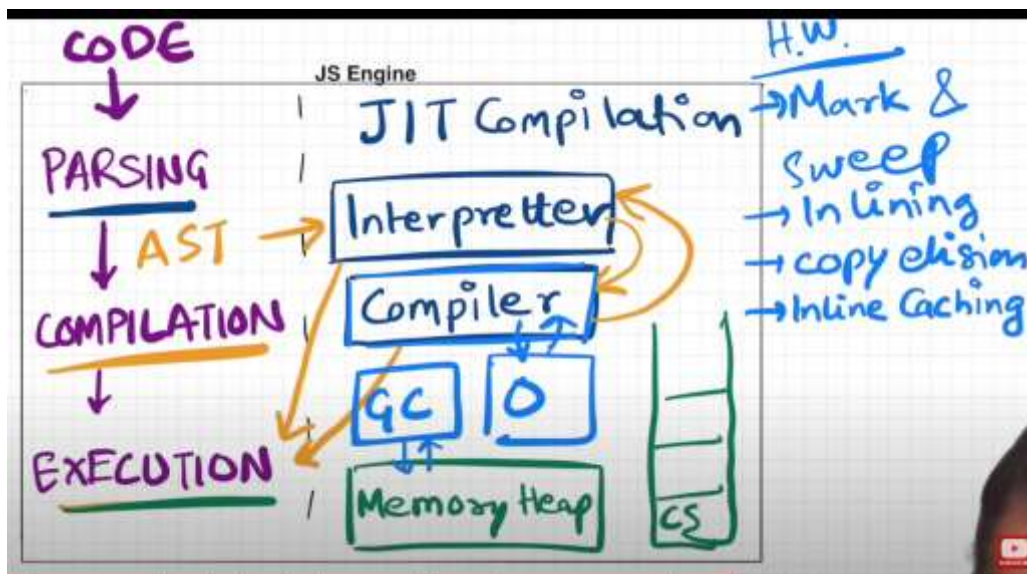
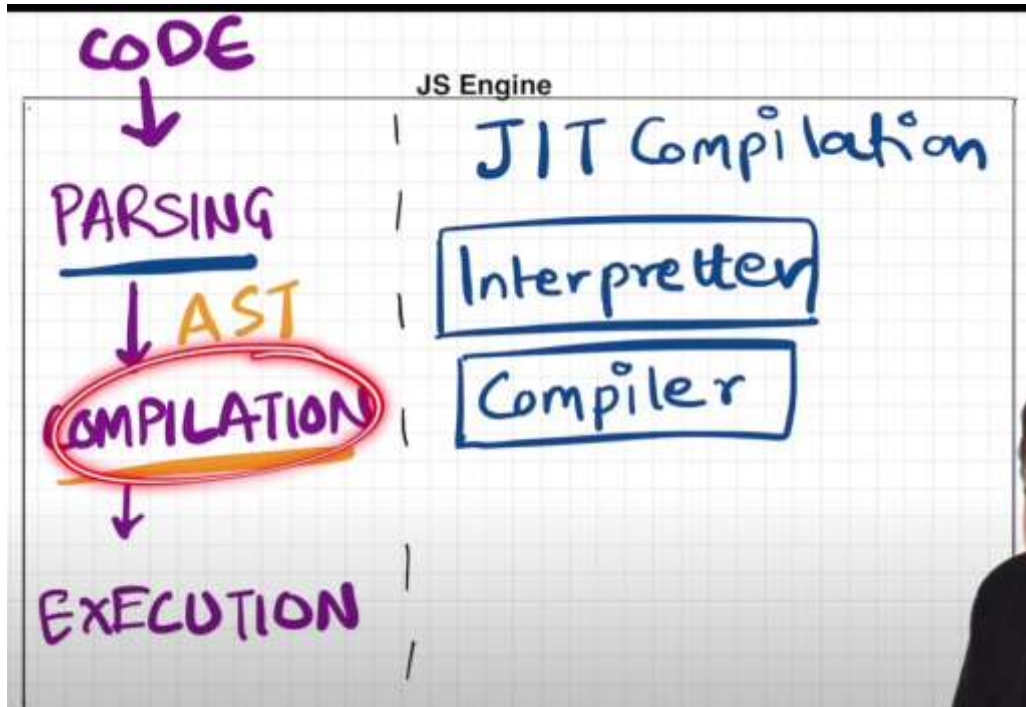
Concurrency model How JavaScript handles multiple tasks happening at the same time.



HOW ASYNCHRONOUS JAVASCRIPT WORKS BEHIND THE SCENES







V8 JS Engine

