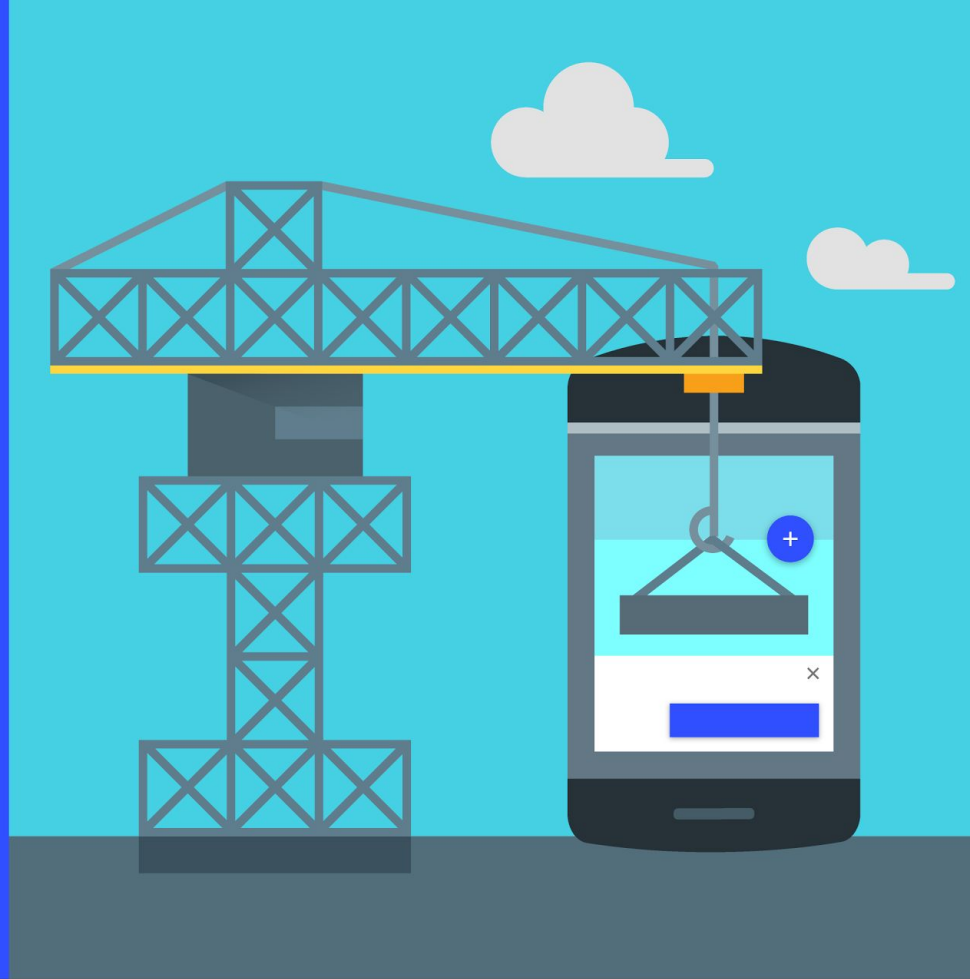# Fetch API

A simple interface for fetching resources

What you'll learn

- What is Fetch?
- Making a request
- Reading the response
  - Example: Fetching images
  - Example: Fetching HTML
- HEAD and POST requests
  - Example: HEAD requests
  - Example: POST requests
  - Example: Advanced use case

What is fetch?

What is fetch?

The Fetch API is a simple interface for making HTTP calls

fetch("path/to/resource") ⟹ Result

What is fetch? (cont.)

- Modern replacement for XMLHttpRequest
  - Automatically handles redirection, etc.
  - Promise based
  - Can be used with service workers
  - Supports Cross Origin Resource Sharing (CORS)

Cross Origin Resource Sharing (CORS)

- Browsers enforce the *single origin model*:
  - HTTP requests <u>must</u> match page's scheme, hostname, and port
  - (Exception: images, scripts, video/audio, embeds)
  - Ex: www.example.com requests JSON from www.json.com, fails
- Browser can request cross-origin access via CORS
  - Adds **origin** header on request, server sends **access-control-allow-origin** if allowed

Making a request

Make a request

```
fetch('/examples/example.json')

.then(function(response) {

  // Handle the response

})

.catch(function(error) {

  console.log('Fetch failed', error);

});
```

Response objects

Properties

headers

status

statusText

type

url

…

Methods

blob( )

json( )

text( )

arrayBuffer( )

formData( )

…

Validate responses

```
fetch('non-existent.json')
  .then(function(response) {
    // Handle the response
  })
  .catch(function(error) {
    console.log('Fetch failed', error);
  });
```

Validate responses (cont.)

```javascript
fetch('non-existent.json')
  .then(function(response) {
    if (!response.ok) {
      throw Error(response.statusText);
    }
    // Code using the response
  })
  .catch(function(error) {
    console.log(error);
  });
```

Reading the response object

Read the response object

```javascript
fetch('/examples/example.json')
.then(function(response) {
  if (!response.ok) {
    throw Error(response.statusText);
  }
  return response.json(); // Read the response
})
.then(function(json) {
  console.log(json); // Do stuff with the read response
})
.catch(function(error) {...});
```

Step 1: fetch the response

```
fetch('/examples/example.json')
.then(readResponseAsJSON)
.then(logResult)
.catch(logError);
```

Step 2: read the response

```
fetch('/examples/example.json')
.then(readResponseAsJSON)
.then(logResult)
.catch(logError);
```

```
function readResponseAsJSON(response) {
  if (!response.ok) {
    throw Error(response.statusText);
  }
  return response.json();
}
```

Step 3: log the data

```
fetch('/examples/example.json')
.then(readResponseAsJSON)
.then(logResult)
.catch(logError);
```

```
function logResult(json) {
  console.log(json);
}
```

Example: fetch images

```
fetch('/images/kitten.jpg')
.then(readResponseAsBlob)
.then(showImage)
.catch(logError);
```

Example: fetch images (cont.)

```
fetch('/images/kitten.jpg')
.then(readResponseAsBlob)
.then(showImage)
.catch(logError);
```

```
function readResponseAsBlob(response) {
    if (!response.ok) {
        throw Error(response.statusText);
    }
    return response.blob();
}
```

Example: fetch images (cont.)

```
fetch('/images/kitten.jpg')
.then(readResponseAsBlob)
.then(showImage)
.catch(logError);
```

```
function showImage(imgBlob) {
  var myImage = document.querySelector('img');
  var url = URL.createObjectURL(imgBlob);
  myImage.src = url;
}
```

Example: fetch HTML

```
fetch('/pages/offline.txt')
.then(readResponseAsText)
.then(updatePage)
.catch(logError);
```

Example: fetch HTML (cont.)

```
fetch('/pages/offline.html')
.then(readResponseAsText)
.then(updatePage)
.catch(logError);
```

```
function readResponseAsText(response) {
  if (!response.ok) {
    throw Error(response.statusText);
  }
  return response.text();
}
```

Example: fetch HTML (cont.)

```
fetch('/pages/offline.html')
.then(readResponseAsText)
.then(updatePage)
.catch(logError);
```

```
function updatePage(responseAsText) {
  var msg = document.querySelector('#msg')
  msg.textContent = responseAsText;
}
```

HEAD and POST requests

Example: HEAD Requests

```
fetch('/examples/example.html', {
  method: 'HEAD'
})
```

Example: POST Requests

```
fetch('someurl/login', {

  method: 'POST',

  body: 'username=john&password=12345'

})
```

Example: Advanced use case

```javascript
fetch('examples/example.json')

.then(function(response) {

  if (!response.ok) {

    fetch('backup.html').then(readResponseAsText).then(updateHtml);

  } else {

    readResponseAsJSON(response).then(logResult);

  }

})

.catch(logError);
```

Resources

- CORS
- Setting up a local server
- Fetch Polyfill
- Browsers that support fetch
- Fetch documentation
- Response objects

- XMLHttpRequests
- Promises
- Readable streams
- Blobs
- URL object's
- HTTP methods