

Request & Responses

Introduction

- How an API works is based on requests and responses
- Front End requests data to the server
- Back end corresponds by sending data back to the client appropriate to the request.

What is an API Request?

- An API request happens when the Client makes a call to the server hence “Request”
 - Requesting data from the server.
- The Client calls the server through a URL which then the API corresponds by running a url specific function
- The URL is what's known as a route

- Requests can also carry data to the server
 - Username and password for login
 - Product name for fetching a specific data
- One of the methods for carrying data from the client is known as request query parameters or req.query for short.



```
const API_URL = "http://localhost:2000";
```

```
const API_URL_QUERIES = "http://localhost:2000?name=lian&job=coder&online=yes";
```

Request Query Parameter

- req.query or request query parameter is the data that comes after the “?” in your url

```
"name=lian&job=coder&online=yes"
```

```
"name=lian&job=coder&online=yes"
```

- Given the string we can see that
 - name = lian
 - job = coder
 - online = yes
- Each property is separated by a "&"



```
req.query = {  
  name : 'lian',  
  job : 'coder',  
  online : 'yes',  
};
```

- When parsed, the queries from the url will form as an object
- Which then will be used accordingly to the endpoint called

Header


“A HTTP header lets the client / server pass additional information about the resource being fetched or about the data being sent as a response”

Headers

- Headers is generally used for sending additional information such as
 - Content - Type : Type of data
 - application/json
 - text/plain
 - text/html
 - multipart/form-data
 - Content - Length : size of data given in decimal of bytes
 - Authorization : Contains credentials to authenticate user with the server
 - Headers are available for requests and response

What is an API Response?

- An API response happens when the server gives the corresponding protocol to the API request
 - Reacting / Responding
- The server will then fetch the requested resources which then is sent back to the client



```
const API_URL = "http://localhost:2000";

// When a client is requesting data the following code is written
axios.get(API_URL)
  .then(res => {
    // the response by the api is represented the "res" object
    // data sent by the server is inside the data property of res
    console.log(res.data);
  })
  .catch(err => {
    // error handler
    console.log(err)
  })
```

HTTP Status Code

“HTTP response status codes indicate whether a specific request has been completed”

Groups of Response Codes

- 1xx Informational
- 2xx Successful
- 3xx Redirect
- 4xx Client Errors
- 5xx Server Error

HTTP status codes in a nutshell:

Or, what a 404 error
& all of its friends mean
in simple human words.

1xx Hold on.

2xx Here you go.

3xx Go away.

4xx You screwed up.

5xx I screwed up.



1XX Informational		4XX Client Error Continued	
100	Continue	409	Conflict
101	Switching Protocols	410	Gone
102	Processing	411	Length Required
2XX Success		412	Precondition Failed
200	OK	413	Payload Too Large
201	Created	414	Request-URI Too Long
202	Accepted	415	Unsupported Media Type
203	Non-authoritative Information	416	Requested Range Not Satisfiable
204	No Content	417	Expectation Failed
205	Reset Content	418	I'm a teapot
206	Partial Content	421	Misdirected Request
207	Multi-Status	422	Unprocessable Entity
208	Already Reported	423	Locked
226	IM Used	424	Failed Dependency
3XX Redirectional		426	Upgrade Required
300	Multiple Choices	428	Precondition Required
301	Moved Permanently	429	Too Many Requests
302	Found	431	Request Header Fields Too Large
303	See Other	444	Connection Closed Without Response
304	Not Modified	451	Unavailable For Legal Reasons
305	Use Proxy	499	Client Closed Request
307	Temporary Redirect	5XX Server Error	
308	Permanent Redirect	500	Internal Server Error
4XX Client Error		501	Not Implemented
400	Bad Request	502	Bad Gateway
401	Unauthorized	503	Service Unavailable
402	Payment Required	504	Gateway Timeout
403	Forbidden	505	HTTP Version Not Supported
404	Not Found	506	Variant Also Negotiates
405	Method Not Allowed	507	Insufficient Storage
406	Not Acceptable	508	Loop Detected
407	Proxy Authentication Required	510	Not Extended
408	Request Timeout	511	Network Authentication Required
		599	Network Connect Timeout Error

HTTP STATUS CODES

When a browser requests a service from a web server, an error may occur.
This is a list of HTTP status messages that might be returned.

Request & Responses