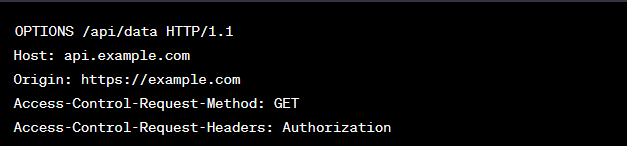
+Step-by-step guide to preflight CORS

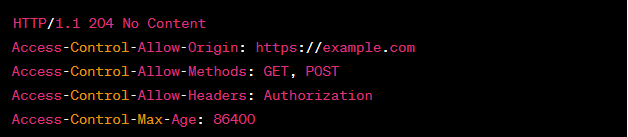
Suppose we have a web page at **https://example.com** that needs to make a cross-origin request to an API at **https://api.example.com**. The API requires authentication and only allows **GET** and **POST** methods.

1. The web page sends a preflight request to the API to determine whether the actual request will be allowed. The preflight request is an HTTP **OPTIONS** request that includes the following headers:



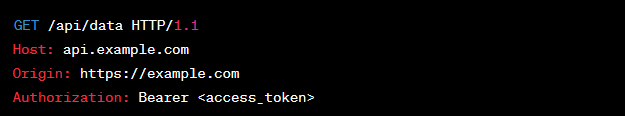
The **Origin** header indicates the origin of the web page, while the **Access-Control-** **Request-Method** and **Access-Control-Request-Headers** headers indicate the HTTP method and custom headers that will be used in the actual request.

1. The API server receives the preflight request and checks its CORS policy to determine whether to allow the request. The server should respond with the appropriate CORS headers to indicate whether the actual request is allowed.
2. The API server sends a preflight response to the web page with the appropriate CORS headers. The preflight response includes the following headers:



The **Access-Control-Allow-Origin** header indicates which origins are allowed to access the resource. In this case, it's set to the **Origin** header in the preflight request, which is **https://example.com**. The **Access-Control-Allow-Methods** header indicates which HTTP methods are allowed for the resource, and the **Access-Control-** **Allow-Headers** header indicates which HTTP headers are allowed for the resource. The **Access-Control-Max-Age** header indicates how long the results of the preflight request can be cached by the client.

1. If the preflight response indicates that the actual request is allowed, the web page sends the actual request to the API. The actual request is an HTTP **GET** request that includes the following headers:



The **Authorization** header includes an access token that the API can use to authenticate the request.

1. The API server receives the actual request and checks its CORS policy to determine whether to allow the request. If the CORS policy allows the request, the API processes the request and sends a response back to the web page with the appropriate CORS headers. If the CORS policy does not allow the request, the API should return a 403 Forbidden status code and the appropriate CORS headers to indicate that the request was not allowed.