# Tìm hiểu về Axios

## Axios là gì

Axios là một HTTP client được viết dựa trên Promises được dùng để hỗ trợ cho việc xây dựng các ứng dụng API từ đơn giản đến phức tạp và có thể được sử dụng cả ở trình duyệt hay Node.js.

## Cách cài đặt axios

Kéo thư viện này từ một CDN như theo cách truyền thống

<script src="https://unpkg.com/axios/dist/axios.min.js"></script>

## Các phương thức trong axios

**Get**

Lấy dữ liệu từ server bằng HTTP GET

Thường nhận tham số là một ***url***

|  |  |
| --- | --- |
| |  | | --- | | axios.get(“/user”) | |

**Post**

Đưa dữ liệu lên server

Post nhận vào hai tham số là ***url*** và ***data***

|  |  |
| --- | --- |
| |  | | --- | | axios.post(“/user”, {“name”:”Le Thanh Huyen”, “age”:21, “school”: “HNUE”) | |

**Put**

Put thường được dùng để ***THAY ĐỔI*** dữ liệu trên server

Put cũng nhận hai tham số đầu vào là ***url*** và ***data***

|  |  |
| --- | --- |
| |  | | --- | | axios.put(“/user/name=Le Thanh Huyen”,{“school”: “HANU”) | |

**Delete**

Là phương pháp xóa dữ liệu trên server

Nhân vào một tham số là ***url***

|  |  |
| --- | --- |
| |  | | --- | | axios.delete(“/user/id=3”) | |

## Giá trị trả về sau khi request bằng axios

Do axios được xây dựng dựa trên promise nên kết quả trả về sẽ là một promise. Để xem được giá trị bên trong promise, ta dùng hàm .then() và .catch()

|  |  |
| --- | --- |
| |  | | --- | | axios.get(“/user”)  .then((response) => {  if(response.name[0]===”H”)  {  console.log(response))  }  )  .catch(error => console.log(error)) | |

## So sánh axios và fetch

Ta xét cụ thể hai ví dụ sau

|  |  |
| --- | --- |
| |  | | --- | | axios.post(<https://jsonplaceholder.typicode.com/users>, {  “name”: “Bret”,  “phone” : 821 794 384  })  .then((response) => console.log(response.data))  .catch(error => console.log(error)) | |

|  |  |
| --- | --- |
| |  | | --- | | fetch(<https://jsonplaceholder.typicode.com/users>, {  method: “POST”,  body: JSON.stringify({  “name”: “Bret”,  “phone” : 821 794 384  })  )  .then(res => {  if(!response.ok) throw Error(response.statusText);  return respone.json();  })  .then(data => console.log(data))  .catch(error => console.log(error)); | |

- Khi sử dụng **fetch** thì chúng ta mất 2 lần promise thì mới nhận được kết quả, còn với **axios** thì nó lấy trực tiếp khi promise return

- Với **fetch**, ta phải chuyển kiểu dữ liệu sang *json.stringify*, điều đó không cần thiết với **axios**

- Ta có thể kiểm tra lỗi với **axios** một cách trực tiếp, còn với **fetch** thì phải kiểm tra mã lỗi

## Axios config

Dùng để chỉ định các cấu hình (config) thích hợp cho aixos

|  |  |
| --- | --- |
| url:’/user/’ | Là đích đến của request |
| method : ‘get’ | Mặc định là “get” |
| baseURL: ‘https://some-domain/com/api/’ | Được gán trước vào url nếu url là một đường dẫn tương đối |
| transformRequest: [function(data, headers)  {  //các hàm sử lý  return data;  }] | Thay đổi dữ liệu trước khi gửi lên server |
| transformResponse : [function(data)  {  //các hàm xử lý  return data;  }] | Thay đổi dữ liệu trả về trước khi truyền vào then, catch |
| params:{  ID: 7264  } | Các tham số URL sẽ được gửi lên cùng request |
| data: “  fullName: “Le Thanh Huyen”  } | Dữ liệu được gửi lên theo body của request |
| timeout:1000 | Chỉ số mili giây, khi request vượt qua thời gian này thì sẽ bị hủy |
| responseType: ‘json’ | Chỉ định kiểu dữ liệu mà server trả về, có thể là arraybuffer, document, json, text, stream |

Không sử dụng axios config

|  |  |
| --- | --- |
| |  | | --- | | axios.post(<https://jsonplaceholder.typicode.com/users>, {  “name”: “Bret”,  “phone” : 821 794 384  })  .then((response) => console.log(response.data))  .catch(error => console.log(error)) | |

Sử dụng axios config

|  |  |
| --- | --- |
| |  | | --- | | axios.post([‘/users](https://jsonplaceholder.typicode.com/users)’, {  baseURL :’ <https://jsonplaceholder.typicode.com>’;  param: {  “ID” : 1724  }  data:{  “name”: “Bret”,  “phone” : 821 794 384  }  })  .then((response) => console.log(response.data))  .catch(error => console.log(error)) | |

## Global axios defaults

Việc sử dụng axios bằng global axios default có nghĩa nó sẽ cài cấu hình mặc định cho mỗi request

|  |  |
| --- | --- |
| |  | | --- | | axios.defaults.baseURL = 'https://api.example.com';  axios.defaults.headers.common['Authorization'] = AUTH\_TOKEN; | |