

Search Video 124

New Sort View ...

Desktop Downloads Home Gallery Harry - Personal Creative Cloud Files Desktop Documents Downloads Music Pictures Videos Harry. This PC volume(sdal) on tp link wifi.net OS (C:) New Volume (D:) Libraries Network Control Panel Recycle Bin New folder

Name Date modified Type Size

Today api-routes Fri 17:34 File folder

npm install

```
PS C:\Users\iitia\Downloads\Sigma Web Development Course\Sigma-Web-Dev-Course\Video 124> npx create-next-app@latest
? What is your project named? ... api-routes
? Would you like to use TypeScript? ... No / Yes
? Would you like to use ESLint? ... No / Yes
? Would you like to use Tailwind CSS? ... No / Yes
? Would you like to use 'src/' directory? ... No / Yes
? Would you like to use App Router? (recommended) ... No / Yes
? Would you like to customize the default import alias (@/*)? ... No / Yes
Creating a new Next.js app in C:\Users\iitia\Downloads\Sigma Web Development Course\Sigma-Web-Dev-Course\Video 124\api-routes.

Using npm.

Initializing project with template: app-tw

Installing dependencies:
- react
- react-dom
- next

Installing devDependencies:
- autoprefixer
- postcss
- tailwindcss
- eslint
- eslint-config-next

[██████████] - idealTree:next: sill Fetch manifest postcss-value-parser@^4.2.0
```

0 items |

22°C Partly sunny

Search

17:34 Fri

ENG IN

Subscribe



EXPLORER API-ROUTES ...

JS route.js X

app > api > add > JS route.js

1

JS route.js

favicon.ico

globals.css

JS layout.js

JS page.js

node\_modules

public

.eslintrc.json

.gitignore

jsconfig.json

next.config.mjs

package-lock.json

package.json

postcss.config.js

README.md

tailwind.config.js

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

node + ▾ □ ... ^ X

PS C:\Users\iitia\Downloads\Sigma Web Development Course\Sigma-Web-Dev-Course\Video 124\api-routes> npm run dev

> api-routes@0.1.0 dev

> next dev

▲ Next.js 14.1.0

- Local: http://localhost:3000

✓ Ready in 2.2s



Ln 1, Col 1 Spaces: 4 UTF-8 CRLF

22°C Partly sunny

Sigma Web Development Course - W | Create Next App | File Conventions: route.js | Next.js

nextjs.org/docs/app/api-reference/file-conventions/route

Sigma Web Developme... GitHub - CodeWithHarr... Learn to code online - C... EJS with Express

# ▲ / NEXT.js

Showcase Docs Blog Analytics Templates Enterprise

Search documentation... CtrlK Feedback Learn

## HTTP Methods

Using App Router  
Features available in /app

- Configuring
- Testing
- Authentication
- Deploying
- Upgrading

### API Reference

- Components
- File Conventions
  - default.js
  - error.js
  - instrumentation.js
  - layout.js
  - loading.js
  - middleware.js
  - not-found.js
  - page.js
  - route.js**
- Route Segment Config
- template.js
- Metadata Files
  - Functions
  - next.config.js Options
  - create-next-app

A **route** file allows you to create custom request handlers for a given route. The following [HTTP methods](#) are supported: `GET`, `POST`, `PUT`, `PATCH`, `DELETE`, `HEAD`, and `OPTIONS`.

```
route.js
```

```
1 export async function GET(request) {}  
2  
3 export async function HEAD(request) {}  
4  
5 export async function POST(request) {}  
6  
7 export async function PUT(request) {}  
8  
9 export async function DELETE(request) {}  
10  
11 export async function PATCH(request) {}  
12  
13 // If 'OPTIONS' is not defined, Next.js will automatically implement 'OPTION'  
14 export async function OPTIONS(request) {}
```

Good to know: Route Handlers are only available inside the `app` directory. You do not need to use API Routes (`pages`) and Route Handlers (`app`) together, as Route Handlers should be able to handle all use cases.

## Parameters

**request (optional)**

The `request` object is a `NextRequest` object, which is an extension of the Web

On this page

- HTTP Methods
- Parameters
  - request (optional)
  - context (optional)
- NextResponse
- Version History

Edit this page on GitHub



22°C Partly sunny

Search

ENG IN 17:30 Fri

Sigma Web Development Course - W | localhost:3000 | File Conventions: route.js | Next.js | Using the Fetch API - Web APIs | MDN

developer.mozilla.org/en-US/docs/Web/API/Fetch\_API/Using\_Fetch

Sigma Web Developme... GitHub - CodeWithHarr... Learn to code online - C... EJS with Express

mdn web docs References Guides Plus Blog Play AI Help BETA

Theme  Log in Sign up for free

English (US)

References > Web APIs > Fetch API > Using the Fetch API

Filter

Fetch API

Using the Fetch API

Fetch basic concepts

Cross-global fetch usage

Headers

Request

Response

Methods

fetch()

POST Req. →

The `fetch()` method can optionally accept a second parameter, an `init` object that allows you to control a number of different settings:

See `fetch()` for the full options available, and more details.

JS

```
// Example POST method implementation:  
async function postData(url = "", data = {}) {  
  // Default options are marked with *  
  const response = await fetch(url, {  
    method: "POST", // *GET, POST, PUT, DELETE, etc.  
    mode: "cors", // no-cors, *cors, same-origin  
    cache: "no-cache", // *default, no-cache, reload, force-cache, only-if-cached  
    credentials: "same-origin", // include, *same-origin, omit  
    headers: {  
      "Content-Type": "application/json",  
      // 'Content-Type': 'application/x-www-form-urlencoded',  
    },  
    redirect: "follow", // manual, *follow, error  
    referrerPolicy: "no-referrer", // no-referrer, *no-referrer-when-downgrade, origin, origin-when-cross-origin, same-origin, strict-origin, strict-origin-when-cross-origin, unsafe-url  
    body: JSON.stringify(data), // body data type must match "Content-Type" header  
  });  
  return response.json(); // parses JSON response into native JavaScript objects  
}  
  
postData("https://example.com/answer", { answer: 42 }).then((data) => {  
  console.log(data); // JSON data parsed by `data.json()` call  
});
```

Note that `mode: "no-cors"` only allows a limited set of headers in the request:

Accept

In this article

Supplying request options

- Abort a fetch
- Sending a request with credentials included
- Uploading JSON data
- Uploading a file
- Uploading multiple files
- Processing a text file line by line
- Checking that the fetch was successful
- Supplying your own request object
- Headers
- Response objects
- Body
- Feature detection
- Differences from `jQuery.ajax()`

See also



22°C Partly sunny

Search

ENG IN

17:35 Fri

The screenshot shows a code editor with a dark theme. The top bar includes icons for File, Edit, Selection, View, Go, and a search bar containing "api-routes". The left sidebar has icons for node, page.js, and route.js, with route.js currently selected. The main editor area displays the following code:

```
1 import { NextResponse } from "next/server";  
2  
3 export async function POST(request) {  
4     let data = await request.json()  
5     console.log(data)  
6     return NextResponse.json({success: true, data: "yes"})  
7 }  
8
```

Handwritten annotations are present in the code:

- A red arrow points from the word "POST" in the third line to the first line, indicating the import of the NextResponse module.
- A blue arrow points from the "console.log(data)" line to the fifth line, indicating the output of "data" to the console.
- A large blue arrow points upwards from the "On console" text to the "console.log(data)" line.
- Handwritten text "getting data on backend" is written vertically next to the blue arrow.
- Handwritten text "On console" is written below the blue arrow.

File Edit Selection View Go ... ⏪ ⏩ api-routes

node page.js route.js

app > page.js > Home > handleClick > data > role

```
1 import Image from "next/image";
2
3 export default function Home() {
4     const handleClick = async () => {
5         let data = {
6             name: "Harry",
7             role: "Coder"
8         }
9         let a = await fetch("/api/add", {
10             method: "POST", headers: {
11                 "Content-Type": "application/json",
12             },
13             body: JSON.stringify(data),
14         })
15         let res = await a.json()
16         console.log(res)
17     }
18
19     return (
20         <div>
21             <h1 className="text-xl font-bold">Next.js Api routes demo</h1>
22             <button onClick={handleClick}>click me</button>

```

Add <'use-client'>



Ln 7, Col 19 Spaces: 2 UTF-8 LF

22°C Partly sunny

Search

17:30 Fri

Σ node X JS page.js JS route.js



node page.js route.js x

app > api > add > JS route.js > POST > [o] data

```
1 import { NextResponse } from "next/server";
2
3 export async function POST(request) {
4     let data = await request.json()
5     console.log(data)
6     return NextResponse.json({success: true, data: data})
7 }
8
```



0 0 0 0

GBP/INR +0.23%

Sigma Web Development Course - W Create Next App File Conventions: route.js | Next.js Using the Fetch API - Web APIs | MDN

localhost:3000

Sigma Web Developme... GitHub - CodeWithHarr... Learn to code online - C... EJS with Express

## Next.js Api routes demo

click me

Elements Console Sources Network Performance Security Memory > Default levels ▾ No Issues

react-dom.development.js:32366

Download the React DevTools for a better development experience: <https://reactjs.org/link/react-dev-tools>

page.js:17

▼ Object i  
  data: "yes"  
  success: true  
  ► [[Prototype]]: Object

[Fast Refresh] rebuilding

hot-reloader-client.js:181

► {success: true, data: 'yes'}

page.js:17

② [Fast Refresh] rebuilding

hot-reloader-client.js:181

► {success: true, data: {...}} i  
  ► data: {name: 'Harry', role: 'Coder'}  
  success: true  
  ► [[Prototype]]: Object

page.js:17



GBP/INR +0.23%

Search

17:45 Fri