



# What are middlewares?

Middlewares are code that runs before / after your request handler.

It's commonly used for things like

1. Analytics
2. Authentication
3. Redirecting the user

## ▼ Code

```
import express from "express";  
  
const app = express();  
  
let requestCount = 0;  
  
app.use(  
  function middleware(req, res, next) {
```

[Back to home](#)[Jump To ↗](#)[◀ Prev](#)[Next ▶](#)[Go to Top ↑](#)

{}

);



```
Middlewares in next 4 of 5  eq, res) => {
    res.send("Hello world");
}

app.get("/requestCount", (req, res) => {
    res.json({
        requestCount
    })
})

app.listen(3000);
```

## ▼ Code

```
import express from "express";
import jwt from "jsonwebtoken";

const app = express();

// @ts-ignore
async function authMiddleware(req, res, next) {
    const token = req.headers.authorization.split(" ")[1];
    const decoded = jwt.verify(token, "secret");
    // ...
}

} else {
```

```
res.status(401).send("Unauthorised");
```



## Middlewares in next 4 of 5

```
app.get("/", authMiddleware, (req, res) => {
  res.send("You are logged in");
})  
  
app.listen(3000);
```

# middlewares + Next

Ref - <https://nextjs.org/docs/app/building-your-application/routing/middleware>

Middleware allows you to run code before a request is completed.

se by



## 1. rewriting

### Middlewares in next 4 of 5

3. modifying the request or response headers
4. or responding directly.

## Use cases

- Authentication and Authorization: Ensure user identity and check session cookies before granting access to specific pages or API routes.
- Logging and Analytics: Capture and analyze request data for insights before processing by the page or API.



## Create a request count middleware

- Create an empty NextJS project

```
npx create-next-app
```

```
→ Projects npx create-next-app
Need to install the following packages:
create-next-app@14.1.4
Ok to proceed? (y) y
✓ What is your project named? ... middleware-test
✓ 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 /Users/harkiratsingh/Projects/middleware-test.
```

- Create middleware.ts in the root folder

**Note:** While only one `middleware.ts` file is supported per project, you can still organize your middleware logic modularly. Break out middleware functionalities into separate `.ts` or `.js` files and import them into your main `middleware.ts` file. This allows for cleaner management of route-specific middleware, aggregated in the `middleware.ts` for centralized control. By enforcing a single middleware file, it simplifies configuration, prevents potential conflicts, and



## Middlewares in next 4 of 5 : track the number of requests

```
import { NextResponse } from 'next/server'  
import type { NextRequest } from 'next/server'  
  
let requestCount = 0;  
export function middleware(request: NextRequest) {  
  requestCount++;  
  console.log("number of requests is " + requestCount);  
  return NextResponse.next()  
}
```



- Try visiting the website

## Code #2

/api

start with

- Add a dummy API route ( `api/user/route.ts` )



## Middlewares in next 4 of 5

```
  response } from "next/server";
```

```
export function GET() {
  return NextResponse.json({
    message: "Hi there"
  })
}
```

- Update `middleware.ts`

```
import { NextResponse } from 'next/server'
import type { NextRequest } from 'next/server'
```

```
let requestCount = 0;
export function middleware(request: NextRequest) {
  requestCount++;
  console.log("number of requests is " + requestCount);
  return NextResponse.next()
}
```

```
// See "Matching Paths" below to learn more
export const config = {
  matcher: '/api/:path*',
```

```
}
```



# Selectively running middlewares

```
import { NextResponse } from 'next/server'
import type { NextRequest } from 'next/server'

export function middleware(request: NextRequest) {
  console.log(request.nextUrl.pathname)
  if (request.nextUrl.pathname.startsWith('/admin')) {
    return NextResponse.redirect(new URL('/signin', request.url))
  }

  if (request.nextUrl.pathname.startsWith('/dashboard')) {
    return NextResponse.next()
  }
}
```



Middlewares in next 4 of 5 com/code100x/cms/blob/main/src/middleware.ts