

# eslint-nestjs-apiresponse

---

A simple ESLint plugin that enforces the use of `HttpStatus` enum from `@nestjs/common` in NestJS applications.

## Features

- Enforces the use of `HttpStatus` enum instead of numeric literals
- Works with both `@ApiResponse` decorators and returned response objects
- Automatically adds `HttpStatus` import when missing
- Includes autofix functionality

## Installation

```
npm install eslint-nestjs-apiresponse --save-dev
```

## Usage

For ESLint 9+ (flat config), add to your `eslint.config.js` or `eslint.config.mjs`:

```
import nestjsApiResponse from "eslint-nestjs-apiresponse";

export default [
  {
    plugins: {
      "eslint-nestjs-apiresponse": nestjsApiResponse
    },
    rules: {
      "eslint-nestjs-apiresponse/require-http-status": "error"
    }
  }
];
```

For ESLint 8 and below (`.eslintrc`):

```
{
  "plugins": ["eslint-nestjs-apiresponse"],
  "rules": {
    "eslint-nestjs-apiresponse/require-http-status": "error"
  }
}
```

Or use the recommended configuration:

```
{
  "extends": ["plugin:eslint-nestjs-apiresponse/recommended"]
}
```

## Running from CLI

To run the plugin directly from command line:

```
# Check files
npx eslint --rule 'eslint-nestjs-apiresponse/require-http-status: error'
src/

# With auto-fix
npx eslint --rule 'eslint-nestjs-apiresponse/require-http-status: error'
--fix src/
```

## Examples

✗ Incorrect:

```
// In decorators
@ApiResponse({ status: 200, description: 'Success' })

// In response objects
return { status: 200, data: result };
```

✓ Correct:

```
import { HttpStatus } from '@nestjs/common';

// In decorators
@ApiResponse({ status: HttpStatus.OK, description: 'Success' })

// In response objects
return { status: HttpStatus.OK, data: result };
```

## Requirements

- Node.js >=14.0.0
- ESLint >=9.0.0 (for flat config) or >=7.0.0 (for legacy config)
- @nestjs/common >=8.0.0
- @nestjs/swagger >=5.0.0

# Authors

- Cline (Senior TypeScript Developer)
- 1MaJKeL1 (Project Supervisor)

# License

ISC