**Exercise 8: Custom middleware**

# Objective

In this exercise, you'll learn how to write middleware in an Express application that is a crucial skill for developing efficient, scalable, and maintainable web applications. Middleware functions can perform a variety of tasks like logging, parsing, authentication, and more. They have access to the request (req), the response (res), and the next function in the application’s request-response cycle. The next function is a function in the Express router which, when invoked, executes the middleware succeeding the current middleware.

# Exercises

A screen shot of a computer program

AI-generated content may be incorrect.

## Exercise 1: create a middleware function that checks the validation of the request body for the articles endpoint

A screenshot of a computer

AI-generated content may be incorrect.

## Exercise 2: create a middleware function that validate the format of the date field.

A screenshot of a computer

AI-generated content may be incorrect.

## Exercise 3: create a middleware function that ensure a text field meets certain length requirements

A screenshot of a computer

AI-generated content may be incorrect.

## Exercise 5: Separating middleware into different files can enhance the modularity and maintainability of your Express.js application, especially as it grows. This approach allows you to organize your code logically, making it easier to understand, debug, and test.

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

# Conclusion

This exercise aims to deepen your understanding of how to build a middleware with Express framework. Middleware functions are powerful tools in Express, allowing you to handle requests efficiently and perform various operations like logging, authentication, and error handling. By writing custom middleware, you can significantly enhance your application's functionality and maintainability. Experiment with different types of middleware to see how they can solve common problems in your applications.