ACÁMICA

Agenda

Daily

Programo: Middleware

Buenas prácticas: Compresión de respuesta

Break

Programamos: Validar tipo de dato

Programan: Autores y libros

Cierre

TEMA DEL DÍA

Middleware

Un middleware es un método que se ejecuta entre el request y la ruta.

Daily

Daily

Sincronizando...

Bitácora

¿Cómo te ha ido?
¿Obstáculos?
¿Cómo seguimos?

Challenge

¿Cómo te ha ido?
¿Obstáculos?
¿Cómo seguimos?

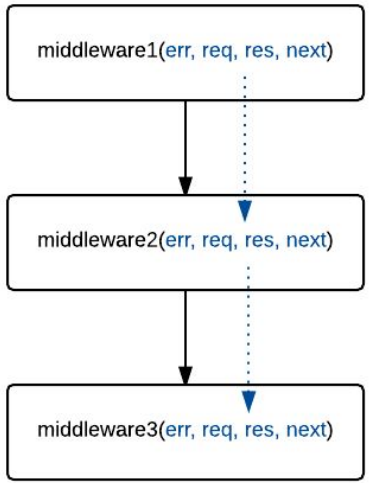
Middleware

Un **middleware** es una forma de agregar código entre nuestra ruta y el servidor de Express.

De esta forma podemos **encadenar** la ejecución de diferentes funciones previo a la ejecución de nuestras rutas.

Un middleware es...

Podemos pensar un **middleware** como un eslabón en una cadena, donde cada eslabón es el encargado de seguir con la cadena o cortarla



Utilidad de un middleware

Una de las utilidades más comunes es la de validar si un usuario está autenticado o si tiene acceso a determinado conjunto de recursos.

Pero también tenemos middlewares para darle formato a la respuesta, para manejar errores, etc.

Cortar la cadena de middlewares

Un middleware puede ser usado, entre cosas, para **detener** la ejecución de una ruta, por ejemplo, en el caso de que el usuario no tenga permisos.

Para esto, en lugar de llamar a la función next(), simplemente devolvemos un response.

```
function validarUsuario(req, res, next){
    if(req.query.usuario !== 'usuario') {
        res.json("Usuario inválido");
    } else {
        next();
    }
}

server.use(validarUsuario);
```

Un middleware para una sola ruta

Solo tenemos que agregar un método **antes** del callback de nuestra ruta específica.

```
function interceptar(req, res, next) {
    res.json("Acceso denegado!!!");
}

server.get('/demo', interceptar, (req, res) => {
    res.json("Hola mundo!!!");
});
```

Programo

mentores/as

Backend

Middleware

Pasemos a una demostración en vivo para entender cómo trabajar con Middlewares.

Buenas prácticas

Comprime tus respuesta

A través de la librería compression puedes comprimir tus respuestas para que las descomprima el cliente.
Todo se realiza de manera automática y puedes disminuir el tamaño de los paquetes que viajan entre cliente servidor.

```
var express = require('express');
var compression = require('compression');
var app = express();
app.use(compression());
```

¡BREAK!

Ph. Credit: Drew Coffmann

Programamos

todos/as

Backend

Programamos

Programemos la siguiente estructura:

Dado el siguiente array en pseudocódigo:

```
[
    {id:1, nombre: Pepe, email: pepe@nada.com }
    {id:2, nombre: Hugo, email: hugo@nada.com}
    {id:3, nombre: Juan, email: juan@nada.com}
]
```

- Crear un endpoint que devuelva una persona por ID (int)
- Crear un endpoint que busque un nombre (char)
- Validar los tipos de datos ingresados al Endpoint a través de un Middleware

Programan

estudiantes

Actividad

Escritores

Genera un array y que cada posición contenga un elemento tipo objeto como el siguiente:

```
{
    id: 1,
    nombre: "Jorge Luis",
    apellido: "Borges",
    fechaDeNacimiento: "24/08/1899",
    libros: [
        {
            id: 1,
            titulo: "Ficciones",
            descripcion: "Se trata de uno de sus más...",
            anioPublicacion: 1944
        },
        {
            id: 2,
            titulo: "El Aleph",
            descripcion: "Otra recopilación de cuentos...",
            anioPublicacion: 1949
        }
    ]
}
```

Actividad

Rutas / Acciones

/autores

- GET: devuelve todos los autores
- POST: crea un nuevo autor

/autores/:id

- GET: devuelve el autor con el id indicado
- DELETE: elimina el autor con el id indicado
- PUT: modifica el autor con el id indicado

Valida a través de un middleware que el escritor exista en tu array

Actividad

Rutas / Libros

/autores/:id/libros

- GET: devuelve todos los libros de un autor
- POST: agrega un nuevo libro al autor

Utiliza el mismo middleware para verificar que le autor exista

/autores/:id/libros/:idLibro

- GET: devuelve el libro con el id indicado del autor
- PUT: modifica el libro con el id indicado del autor
- DELETE: eliminar el libro con el id indicado del autor

Crea un nuevo middleware para verificar la existencia del libro y también que corresponda al autor

DWFS

Nos tomamos unos minutos para completar esta encuesta.

Queremos saber cómo valoran mi tarea hasta acá. ¡Les va a llevar solo un minuto!

Para la próxima

1) Termina el ejercicio del encuentro de hoy.
2) Lee la bitácora 43 y carga las dudas que tengas al Trello.
3) Resuelve el challenge.

En el encuentro que viene uno/a de ustedes será seleccionado para mostrar el ejercicio de hoy y otro/a mostrará cómo resolvió el challenge de la bitácora. De esta manera, ¡aprendemos todos/as de (y con) todos/as, así que vengan preparados/as!

ACÁMICA