

SKILL FACTORY

2022

by **AVALITH.**

Nombre y apellido

Nombre y apellido

Fake Store API

Para simular la base de datos en su lugar vamos a utilizar la API <https://fakestoreapi.com/> para que nos proporcione la funcionalidad de nuestros modelos, es imperativo que seamos capaces de usar las funcionalidades básicas de la API a través de nuestra API

Consigna

1. Crear un proyecto e instalar los módulos correspondientes, crear las carpetas correspondientes para modelos, controladores y middlewares
2. Crear un archivo app.js que levante el proyecto en localhost:3000, y utilice un Express.router
3. Usando la API proporcionada generar modelos para Producto, Carrito y Usuario
4. Crear un middleware para imprimir por consola la fecha en la que se hace cada request
5. Crear un middleware que maneje los errores 404
6. Crear los endpoints que se piden en la siguiente diapositiva y que cumplan sus funcionalidades

Endpoints

- Todos los endpoints básicos de la API de Fakestore se deben poder acceder a través de nuestra API, ej: /products, /products/:id, y sus análogos para Categorías, carritos y usuarios
- GET /products/categories debe devolver un array de objetos que tenga el nombre de la categoría y un array con todos los productos pertenecientes a la categoría
- GET /users/firsts debe devolver los primeros 3 usuarios ordenados por ID
- GET /prices debe devolver una lista de objetos que tengan id, título y precio de cada producto y que se pueda ordenar por precio y se pueda elegir si en orden ascendiente o descendiente a través de un query "order"
- GET /expensive debe devolver todos los productos que sean los más caros de su categoría
- GET /bigcarts debe devolver todos los carritos que tengan más de 2 productos y que diga el nombre del usuario que ordenó ese carrito

Pistas

- Pueden usar las queries limit y order de la API de fake store para replicarlas en la api nuestra
- Acuerdense que para crear nuevos objetos a traves de un array existe el metodo map() y luego existe el metodo sort() para ordenarlos por su cuenta, vean la documentaciòn oficial para entender como funcionan
- Cuando deban combinar request, prueben traer todo lo necesario a una variable en sus controladores usando la keyword AWAIT para tener todo en sus manos
- Acuerdense que en cualquier lugar que se use AWAIT se debe declarar la funcion como ASYNC incluso si es anonima
- Si se devuelven varias Promesas va a ser necesario utilizar Promise.all() asi que en caso de ver que se este devolviendo un array de promesas pending esta puede ser la respuesta

SKILL

SKILL FACTORY

by **AVALITH.**

FACTORY