

Desafío - Citas médicas

En este desafío validaremos nuestros conocimientos de manejo de dependencias y el uso de paquetes en Node. Para lograrlo, necesitarás aplicar tus habilidades en el uso de NPM, concretamente en el conocimiento de paquetes que no procesan datos, así como en la utilización de paquetes NPM diseñados específicamente para el procesamiento de datos.

Lee todo el documento antes de comenzar el desarrollo **individual**, para asegurarte de tener el máximo de puntaje y enfocar bien los esfuerzos.

Descripción

La clínica DENDE Spa ha tenido un problema con su software de gestión de citas y necesita urgentemente una aplicación capaz de registrar a los nuevos usuarios almacenando su nombre, apellido, sexo, hora en la que fue registrado e indispensablemente un código identificador.

Deberás crear un servidor que disponibilice una ruta para la consulta de todos los usuarios registrados.

Mujeres:

1. Nombre: Vicenta - Apellido: Marin - ID: 37adc3 - Timestamp: November 4th 2021, 7:32:35 pm
2. Nombre: Manuela - Apellido: Ramos - ID: 27db77 - Timestamp: November 4th 2021, 7:32:37 pm
3. Nombre: Nadia - Apellido: Marie - ID: eb23a1 - Timestamp: November 4th 2021, 7:32:47 pm
4. Nombre: Anaïs - Apellido: Meunier - ID: a51620 - Timestamp: November 4th 2021, 7:32:54 pm
5. Nombre: Grace - Apellido: Green - ID: ea4c12 - Timestamp: November 4th 2021, 7:32:57 pm
6. Nombre: Vildan - Apellido: Tahincioğlu - ID: 30a5c6 - Timestamp: November 4th 2021, 7:33:00 pm
7. Nombre: Silvia - Apellido: Westhof - ID: 107d62 - Timestamp: November 4th 2021, 7:33:03 pm

Hombres:

1. Nombre: Luis - Apellido: Langø - ID: 38eec1 - Timestamp: November 4th 2021, 7:32:40 pm
2. Nombre: Cooper - Apellido: Wood - ID: 7d67fa - Timestamp: November 4th 2021, 7:32:43 pm
3. Nombre: Patrick - Apellido: Lee - ID: 8b1aa2 - Timestamp: November 4th 2021, 7:33:06 pm
4. Nombre: Tracy - Apellido: Reynolds - ID: 37cc96 - Timestamp: November 4th 2021, 7:33:10 pm

Requerimientos

1. El registro de los usuarios debe hacerse con la [API Random User](#) usando axios para consultar la data. **(2 Puntos)**
2. Cada usuario registrado debe tener un campo id único generado por el paquete UUID. **(2 Puntos)**
3. Cada usuario debe tener un campo timestamp almacenando la fecha de registro obtenida y formateada por el paquete Moment. **(2 Puntos)**
4. Por cada consulta realizada al servidor, se debe devolverle al cliente una lista con los datos de todos los usuarios registrados usando Lodash para dividir el arreglo en 2 separando los usuarios por sexo. **(2 Puntos)**
5. En cada consulta también se debe imprimir por la consola del servidor la misma lista de usuarios pero con fondo blanco y color de texto azul usando el paquete Chalk. **(1 Punto)**
6. El servidor debe ser levantado con el comando Nodemon. **(1 Punto)**



¡Mucho éxito!