

Repaso CRUD (populado)

Referencia endpoints

Id	Method	Path	Description
1	get	/parks/new	Muestra el formulario para crear un parque
2	post	/parks/new	Guarda en la BBDD un parque
3	get	/coasters/new	Muestra el formulario para crear una montaña rusa
4	post	/coasters/new	Guarda en la BBDD una montaña rusa
5	get	/coasters/	Muestra la lista de montañas rusas
6	get	/coasters/:id	Muestra los detalles de una montaña rusa
7	get	/coasters/delete?id=xxx	Elimina de la BBDD la montaña rusa
8	get	/coasters/edit?id=xxx	Muestra el formulario para editar una montaña rusa
9	post	/coasters/edit?id=xxx	Edita en la BBDD la montaña rusa

Iteraciones

0.- Comienza a trabajar [sobre este repo](#)

1.- Crea un modelo `Park` con el esquema:

```
name:      Str
description: Str
active:    Bool
```

2.- Crea un modelo `Coaster` con el esquema:

```
name:      Str
description: Str
inversions: Num
length:    Num
active:     Bool
park:       ObjectId (referenciando al modelo Park)
```

3.- Utiliza los modelos para poblar la BBDD mediante el `seed.js` alojado en `/bin`

4.- Crea la ruta 1 en el servidor y muestra la vista de `parks/new-park.hbs`

5.- Crea y configura la ruta 2 en el servidor para guardar parques en la BBDD mediante el modelo `Park`, **testea mediante Postman su validez**. A continuación configura la vista `new-park.hbs` (ruta 1) cuyo formulario te permitirá crear nuevos parques. No olvides modificar los atributos `action` y `method` de su formulario, así como el atributo `name` de cada campo para conectarlo adecuadamente con el controlador.

6.- Testea en la BBDD que el formulario crea parques sin incidencia.

7.- Crea la ruta 3 en el servidor y muestra en el cliente la vista `coasters/new-coaster.hbs`

8.- En el formulario, las opciones del desplegable deben contener los nombres de cada uno de los parques de la BBDD (debes enviarlos desde el controlador que genera la vista haciendo uso del modelo `Park`). Añade a cada `option` el atributo `value` con el `_id` de ese parque.

Ejemplo:

```
<select name="park" class="form-control" id="park-list">
  <option>Seleccionar</option>
  <option value="5d408c5eff558796d4cf681e">Port Aventura</option>
  <option value="5d408c5eff558796d4cf6894">Isla Mágica</option>
</select>
```

9.- Crea y configura la ruta 4 en el servidor para guardar montañas rusas en la BBDD mediante el modelo `Coaster`, testea mediante Postman su validez. A continuación configura la vista `new-coaster.hbs` (ruta 3) cuyo formulario te permitirá crear nuevas montañas. No olvides modificar los atributos `action` y `method` del formulario, así como el atributo `name` de cada campo para conectarlo adecuadamente con el controlador. El ID del parque al que pertenece llegará al controlador identificado con el valor `name` de la lista de selección.

10.- Testea en la BBDD que el formulario crea montañas rusas sin incidencia.

11.- Crea la ruta 5 en el cliente y muestra la vista de `coasters/index-coasters.hbs`

12.- Adecúa la vista para que muestre todas las montañas rusas, y para que cada botón enlace con la URL `/coasters/id_de_la_montaña_rusa`. Ejemplo:

`http://localhost:3000/coasters/5d408c5eff558796d4cf681e`

13.- Crea la ruta 6 y en el cliente muestra la vista de `coasters/coaster-details.hbs`

14.- Adecúa la vista para que muestre toda la información de la montaña rusa, incluyendo el nombre y descripción del parque al que pertenece.

15.- Incluye en la vista un botón adicional que permita eliminar la montaña rusa, tal y como se solicita en la ruta 7. Recuerda que necesitarás un *endpoint* para ello.

16.- Incluye en la vista un botón adicional que permita editar la montaña rusa, tal y como se solicita en las rutas 8 y 9. Recuerda que necesitarás dos *endpoints* para ello, y recuerda que en la vista de edición (que no se incluye en el *repo*, créala) que tendrás que renderizar la lista de selección donde el usuario pueda seleccionar cualquiera de los parques existentes en el sistema.

¡A por ello!

Bonus I

Crea las vistas restantes del CRUD (índice de parques, detalles de parque...) y las acciones restantes del CRUD (editar parque, eliminar parque...)

Bonus II

Desarrolla **validación de formularios**: en caso de detectar una edición de montaña rusa con cualquier campo vacío, o si en la lista desplegable de parques está la primera opción “Seleccionar” activa, evita que se pueda crear el ítem y muestra un mensaje de error personalizado en la vista.