

REC - PRIMER PARCIAL – LABORATORIO III – 1 cuat. 2023

Aclaración:

Las partes se corregirán de manera secuencial (ascendentemente). Si están bien todos los puntos de una parte, habilita la corrección de la parte posterior.

Se pueden usar templates o reutilizar código hecho, pero en ningún caso, se debe incluir código obsoleto o que no cumpla ninguna función dentro del parcial.

Se debe crear un archivo por cada entidad de TypeScript. Todos los métodos deben estar declarados dentro de clases.

Toda comunicación con el backend se realizará con AJAX. Todo el pasaje de datos se hará con JSON.

Las vistas (páginas .php o .html) deben respetar fielmente las formas, colores, iconos, etc.

Se deben respetar los nombres de los archivos, de las clases, de los métodos y de los parámetros de las peticiones.

Todas las referencias a archivos y/o recursos, deben estar de manera relativa.

La Api Rest está creada en Node.js. El script del servidor Node.js se entrega por parte del profesor, usted deberá transpilar, incluir los módulos de Node.js y agregar la estructura de directorios adecuada y NO modificarla.

La Api Rest tendrá que interactuar con la base de datos **garage_bd** (autos).

La Api Rest deberá estar separada del proyecto del recuperatorio (<http://localhost:2023>).

La Api Rest NO se deberá entregar, sólo el proyecto del recuperatorio.

Parte 1 FRONTEND – HTML5 y TypeScript (hasta 6)

Crear la siguiente clase en **Typescript** en el namespace **Apellido** (del alumno):

- **Auto**, posee como atributos protegidos:
 - patente(cadena)
 - marca(cadena)
 - color(cadena)
 - precio(flotante)

Un constructor (que inicialice los atributos), un método de instancia **toJSON()**, que retornará los datos de la instancia (en una cadena con formato **JSON**).

Crear en **TypeScript** la clase **Manejadora** (en el namespace **RecPrimerParcial**) que posea los siguientes métodos y funcionalidades:

AgregarAutoBD. Obtiene la **patente**, la **marca**, el **color** y el **precio** desde la página **auto_bd.html** y enviará por POST (utilizando **AJAX**) hacia el servidor ubicado en: **"http://localhost:2023/agregarAutoBD"**.

Informar por **consola** y **alert** el mensaje recibido.

Patente	<input type="text"/>
Marca	<input type="text"/>
Color	<input type="color" value="red"/>
Precio	<input type="text" value="100000"/>
<div><button>Agregar</button><button>Mostrar</button><button>Modificar</button></div>	

ListarAutosBD. Recuperará (por **AJAX**) todos los autos de la base de datos, invocando por GET al servidor ubicado en: **"http://localhost:2023/listarAutosBD"**.

Con el array recibido, crear una tabla HTML con cabecera (en el **FRONTEND**) que mostrará toda la información de cada uno de los autos.

Informar por **consola** el mensaje recibido y mostrar el listado en la página (*div id='divTabla'*).

NOTA:

La columna que indica el color del auto debe contener un input (*type="color"*) y estar deshabilitado (*disabled*).

Agregar una columna (Acciones) al listado de autos que permita: **Eliminar** y **Modificar** al auto elegido. Para ello, agregue dos botones (*input [type=button]*) que invoquen a las funciones (de la clase Manejadora), *EliminarAutoBD* y *ModificarAuto*, respectivamente.

Patente	Marca	Precio	Color	Acciones	
abc123	ford	150999	<input type="color" value="red"/>	Eliminar	Modificar
abc321	renault	150999	<input type="color" value="blue"/>	Eliminar	Modificar
cba123	fiat	250000	<input type="color" value="yellow"/>	Eliminar	Modificar
cba321	porche	650000	<input type="color" value="green"/>	Eliminar	Modificar

ModificarAuto. Mostrará todos los datos del auto que recibe por parámetro (objeto **JSON**), en el formulario de la página **auto_bd.html**. Permitirá modificar cualquier campo, a excepción de la **patente**, dejarlo como de sólo lectura.

Al pulsar el botón *"Modificar"*, se invocará por POST (utilizando **AJAX**) al servidor ubicado en:

"http://localhost:2023/modificarAutoBD". Realizar una función para tal fin.

Refrescar el listado sólo si se pudo modificar, caso contrario, informar (por alert y consola) de lo acontecido.

EliminarAutoBD. Recibe como parámetro al objeto **JSON** que se ha de eliminar. Pedir confirmación, mostrando la **patente** y **marca**, antes de eliminar.

Si se confirma se invocará por POST (utilizando **AJAX**) al servidor ubicado en:

"http://localhost:2023/eliminarAutoBD".

Informar por consola y alert lo acontecido. Refrescar el listado para visualizar los cambios.

Parte 2 FRONTEND – HTML5 y TypeScript (hasta 8)

Agregar la siguiente clase en **Typescript**, en el namespace **Apellido** (del alumno):

- **AutoFoto**, hereda de Auto, posee como único atributo **foto**(cadena). Un constructor para inicializar los atributos.

En la clase **ManejadoraAutoFotos** (en el namespace **RecPrimerParcial**), agregar los siguientes métodos:

AgregarAutoFotoBD, MostrarAutoFotosBD, EliminarAutoFotoBD y ModificarAutoFotoBD.

Al cargar la página **auto_foto_bd.html**, se deberá cargar el listado de autos con sus respectivas fotos obtenidos desde la base de datos, para ello se invocará al método **MostrarAutoFotosBD** que enviará por GET (utilizando **AJAX**) hacia el servidor ubicado en: **"http://localhost:2023/listarAutoFotosBD"**.

Con el array de AutoFoto recibido, armar una tabla (HTML) para mostrar los datos.

Preparar la tabla (HTML) con una columna extra para que muestre la imagen de la foto (50px por 50px).

Mostrar el listado en la página (**div id='divTablaAutoFotos'**).

NOTA:

La columna que indica el color del auto debe contener un input (type="color") y estar deshabilitado (disabled).

Agregar una columna (Acciones) al listado de autos que permita: **Eliminar** y **Modificar** al auto con foto elegido. Para ello, agregue dos botones (input [type=button]) que invoquen a las funciones EliminarAutoFotoBD y Modificar, respectivamente.

Patente	Marca	Precio	Color	Foto	Acciones	
abc123	ford	150999	<input type="color" value="red"/>		Eliminar	Modificar
abc321	renault	150999	<input type="color" value="blue"/>		Eliminar	Modificar
cba123	fiat	250000	<input type="color" value="yellow"/>		Eliminar	Modificar
cba321	porche	650000	<input type="color" value="green"/>		Eliminar	Modificar

AgregarAutoFotoBD. Obtiene los datos del auto (incluyendo la foto) desde la página *auto_foto_bd.html* y se enviará por POST (utilizando **AJAX**) hacia “*http://localhost:2023/agregarAutoFotoBD*”.

Informar por **consola** y **alert** el mensaje recibido.

Refrescar el listado de los autos con fotos.

Modificar. Mostrará todos los datos del auto que recibe por parámetro (objeto **JSON**), en el formulario, incluida la foto (mostrarla en “*imgFoto*”).

Permitirá modificar cualquier campo, a excepción de la **patente**, dejarlo como de sólo lectura.

Al pulsar el botón “*ModificarBD*”, invocará al método **ModificarAutoFotoBD**, que llamará por POST (utilizando **AJAX**) hacia “*http://localhost:2023/modificarAutoFotoBD*”.

Refrescar el listado sólo si se pudo modificar, caso contrario, informar (por alert y consola) de lo acontecido.

Patente	<input type="text" value="abc321"/>
Marca	<input type="text" value="renault"/>
Color	<input type="color" value="#4169E1"/>
Precio	<input type="text" value="150999"/>
Foto	<input type="button" value="Seleccionar archivo"/> Sin archivos seleccionados



EliminarAutoFotoBD. Recibe como parámetro al objeto **JSON** que se ha de eliminar. Pedir confirmación, mostrando la **patente** y **marca**, antes de eliminar.

Si se confirma se invocará por POST (utilizando **AJAX**) hacia “*http://localhost:2023/eliminarAutoFotoBD*”.

Informar por consola y alert lo acontecido. Refrescar el listado para visualizar los cambios.