

INSTRUCCIONES PARA LEVANTAR EL PROYECTO

PASO 1|

Al momento que tengamos el proyecto en un api, en este caso se usa visual studio code, desde la terminal ejecutamos el comando "npm install" para que se instalen todas las dependencias que requiera el proyecto.

```
PS D:\NelsonCifuentes-PE> npm install

up to date, audited 120 packages in 987ms

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS D:\NelsonCifuentes-PE> 
```

PASO 2|

Nos dirigimos a colocar en el archivo app.js en donde indica //middlewares, luego en password: "", colocamos la contraseña que se tenga en el equipo de MySQL.

```
// middlewares
app.use(morgan('dev'));
app.use(myConnection(mysql, {
  host: 'localhost',
  user: 'root',
  password: '',
  port: 3306,
  database: 'crudcar'
}, 'single'));
app.use(express.urlencoded({extended: false}));
```

PASO 3|

Una vez estemos dentro de la ruta del proyecto en la terminal, ejecutamos el comando: "mysql -u root -p" para iniciar el proceso de creación de la base de datos en MySQL, nos solicitará la contraseña que nosotros tengamos en nuestro usuario de MySQL, se coloca y se da enter

```
D:\NelsonCifuentes-PE>mysql -u root -p
Enter password: *****
```

Da como resultado esto:

```
D:\NelsonCifuentes-PE>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 654
Server version: 8.0.17 MySQL Community Server - GPL

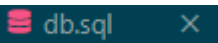
Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

PASO 4|

Una vez estemos dentro de MySQL en la terminal, de esta manera: `mysql>` , nos dirigimos al archivo llamado db.sql en el proyecto:  , para copiar todo el código que hay dentro del archivo;

```
1  -- creacion de base de datos
2  CREATE DATABASE crudcar;
3
4  -- Utilizando la base de datos
5  use crudcar;
6
7  -- creando la tabla
8  CREATE TABLE car(
9      id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
10     marca VARCHAR(50) NOT NULL,
11     modelo VARCHAR(50) NOT NULL,
12     año VARCHAR(50) NOT NULL,
13     placa VARCHAR(50),
14     estado VARCHAR(50) NOT NULL
15 );
16
17
18 -- mostrar todas las tablas
19
20 SHOW TABLES;
21
22
23 -- describir tabla
24 describe car;
25
```

Una vez tengamos copiado el código de creación de la base de datos, lo copiamos en la terminal desde `mysql>` y sin necesidad de oprimir enter se ejecutará. Creando así la base de datos y tabla que necesitamos:

```
mysql> describe car;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id    | int(6) unsigned | NO   | PRI | NULL    | auto_increment |
| marca | varchar(50)     | NO   |     | NULL    |                |
| modelo | varchar(50)     | NO   |     | NULL    |                |
| año   | varchar(50)     | NO   |     | NULL    |                |
| placa | varchar(50)     | YES  |     | NULL    |                |
| estado | varchar(50)     | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```


PASO 5|

Luego, desde la terminal una vez más en la ruta de nuestro proyecto ejecutamos el comando "npm run dev" para inicializar nuestro proyecto y nos mostrará un mensaje como este:

```
PS D:\NelsonCifuentes-PE> npm run dev
> nelsoncifuentes-pe@1.0.0 dev
> nodemon src/app.js

[nodemon] 2.0.19
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node src/app.js`
Servidor funcionando en el puerto 3000
```

PASO 6|

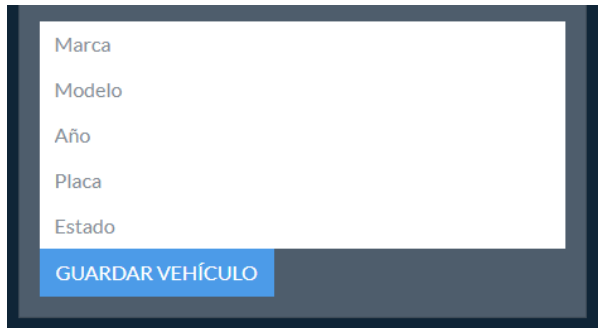
En nuestro navegador, procedemos a escribir en el url: localhost:3000,  localhost:3000 y presionamos enter, se desplegará nuestra aplicación.

PASO 7|

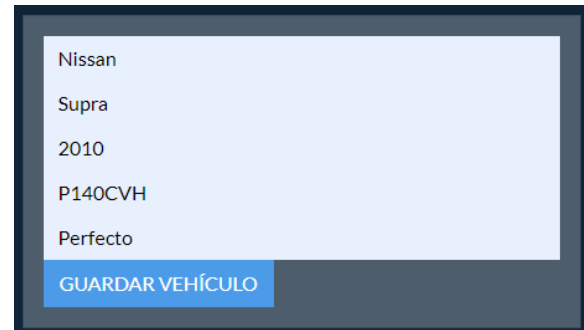
Dentro de la aplicación, podemos guardar, eliminar, editar o simplemente visualizar los vehículos que hemos registrado.

GUARDAR:

Para guardar debemos rellenar todos los datos solicitados del vehículo y presionar el botón de guardar vehículo.



Formulario para guardar un vehículo. Campos: Marca, Modelo, Año, Placa, Estado. Botón: GUARDAR VEHÍCULO.



Formulario para guardar un vehículo. Campos rellenos: Nissan, Supra, 2010, P140CVH, Perfecto. Botón: GUARDAR VEHÍCULO.

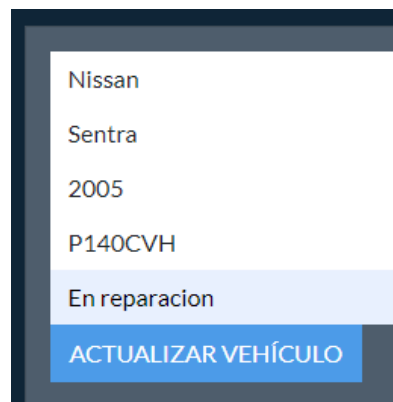
Se guardará y mostrará en pantalla:

| No. | Marca | Modelo | Año | Placa | Estado | Actions | |
|-----|--------|--------|------|---------|----------|---------|----------|
| 1 | Nissan | Sentra | 2010 | P140CVH | Perfecto | Editar | Eliminar |

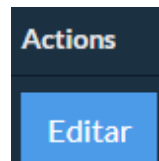
EDITAR:

Para editar cualquier vehículo y cualquier dato, presionamos el botón de editar:

En esta parte corregimos los datos del vehículo a editar y presionamos el botón de actualizar vehículo.



Formulario para actualizar un vehículo. Campos rellenos: Nissan, Sentra, 2005, P140CVH, En reparacion. Botón: ACTUALIZAR VEHÍCULO.



Botón: Editar

Se actualizarán y mostrarán los datos:

| No. | Marca | Modelo | Año | Placa | Estado | Actions | |
|-----|--------|--------|------|---------|---------------|---------|----------|
| 1 | Nissan | Sentra | 2005 | P140CVH | En reparacion | Editar | Eliminar |

ELIMINAR:

Para eliminar cualquier vehículo, debemos presionar el botón de eliminar:

A red rectangular button with the word "Eliminar" in white text.

Y el vehículo se estará eliminando:

| No. | Marca | Modelo | Año | Placa | Estado | Actions |
|-----|-------|--------|-----|-------|--------|---------|
| | | | | | | |

Por último para dar de baja a nuestra aplicación, desde la terminal presionamos las teclas ctrl y la letra c al mismo tiempo y se estará dando de baja al servidor.

```
¿Desea terminar el trabajo por lotes (S/N)?  
PS D:\NelsonCifuentes-PE> ^C  
PS D:\NelsonCifuentes-PE> █
```