

**Creación de objetos de la base de datos**

**Jorge Armando Toloza Carvajal**

**Análisis y desarrollo de software  
Ficha # 2627106**

**Instructor: Carlos Ernesto Lizarazo Sierra**

**SENA  
Regional Santa fe de Antioquia  
Julio 2023**

1. Crear un objeto JSON que modele los datos de un carro:

```
{  
  "placa": "ABC123",  
  "numero_serie": "123456789",  
  "modelo": "Sedan",  
  "marca": "Toyota",  
  "kilometraje": 50000,  
  "tipo": "Automóvil"  
}
```

2. Sentencia para insertar 5 datos en la colección:

```
[  
  {  
    "placa": "ABC123",  
    "numero_serie": "123456789",  
    "modelo": "Sedan",  
    "marca": "Toyota",  
    "kilometraje": 50000,  
    "tipo": "Automóvil"  
  },  
  {  
    "placa": "DEF456",  
    "numero_serie": "987654321",  
    "modelo": "SUV",  
    "marca": "Ford",  
    "kilometraje": 75000,  
    "tipo": "Camioneta"  
  },  
  {  
    "placa": "GHI789",  
    "numero_serie": "246813579",  
    "modelo": "Hatchback",  
    "marca": "Honda",  
    "kilometraje": 30000,  
    "tipo": "Automóvil"  
  },  
  {  
    "placa": "JKL012",  
    "numero_serie": "135792468",  
    "modelo": "Pickup",  
    "marca": "Chevrolet",  
    "kilometraje": 90000,  
    "tipo": "Camioneta"  
  },  
  {  
    "placa": "MNO345",  
    "numero_serie": "567891234",  
    "modelo": "SUV",  
    "marca": "Nissan",
```

```

    "kilometraje": 60000,
    "tipo": "Camioneta"
  }
]

```

3. Sentencia para consultar los datos ingresados en la colección:

`find({})` // Retorna todos los documentos de la colección "parque"

Actualizar los datos de la colección:

Puedes utilizar la función `updateOne()` para actualizar un documento específico en la colección. Por ejemplo, para actualizar el primer registro, puedes utilizar la siguiente sentencia:

The screenshot shows the JSONBuddy application interface. The main editor displays a JSON document with the following structure:

```

{
  // Crear una base de datos llamada "mi_base_de_datos"
  use mi_base_de_datos

  // Crear una colección de datos llamada "parque"
  db.createCollection("parque")

  // Insertar cinco documentos con la estructura JSON para representar los datos de los carros
  db.parque.insertMany([
    {
      "placa": "ABC123",
      "numero_serie": "123456789",
      "modelo": "Sedan",
      "marca": "Toyota",
      "kilometraje": 50000,
      "tipo": "Automóvil"
    },
    {
      "placa": "DEF456",
      "numero_serie": "987654321",
      "modelo": "SUV",
      "marca": "Ford",
      "kilometraje": 75000,
      "tipo": "Camioneta"
    },
    {
      "placa": "GHI789",
      "numero_serie": "246813579",
      "modelo": "Hatchback",
      "marca": "Honda",
      "kilometraje": 30000,
      "tipo": "Automóvil"
    },
    {
      "placa": "JKL012",
      "numero_serie": "135792468",
      "modelo": "Pickup",
      "marca": "Chevrolet",
      "kilometraje": 90000,
      "tipo": "Camioneta"
    },
    {
      "placa": "MNO345",
      "numero_serie": "567891234",
      "modelo": "SUV",
      "marca": "Nissan",
      "kilometraje": 60000,
      "tipo": "Camioneta"
    }
  ])

  // Actualizar los datos del primer y último registro
}

```

The interface includes a File Explorer on the left, a top menu bar with options like File, Edit, View, JSON, XML, File Explorer, Tools, and Help, and a bottom status bar showing 22 Errors, 0 Warnings, 0 Messages, and 0 Patches.