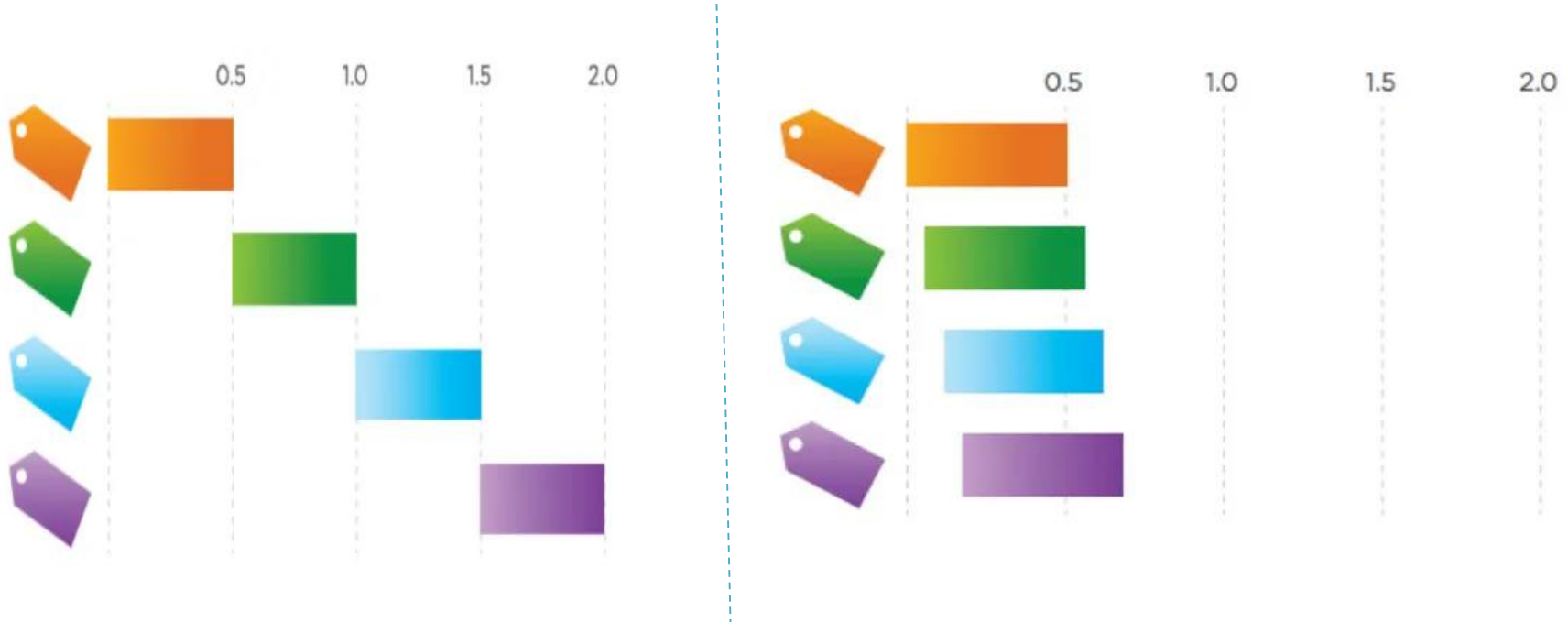


Asíncrono / Síncrono



- JavaScript **ejecuta el código secuencialmente** en orden descendente
- También **puede ejecutar código** después de **que ocurra otro evento en paralelo**



async / await

•Async:

- Palabra clave utilizada para definir una función asíncrona
- Indica que la función siempre devuelve una promesa
- Facilita el uso de await dentro de la función

•Await:

- Palabra clave que se utiliza para esperar la resolución de una promesa de tal forma que podamos volver nuestro código síncrono.
- Sólo puede usarse dentro de funciones async
- Detiene la ejecución de la función hasta que se resuelva la promesa, sin bloquear el hilo principal

Vamos a practicar

A yellow square containing the letters 'JS' in a bold, black, sans-serif font, representing JavaScript.

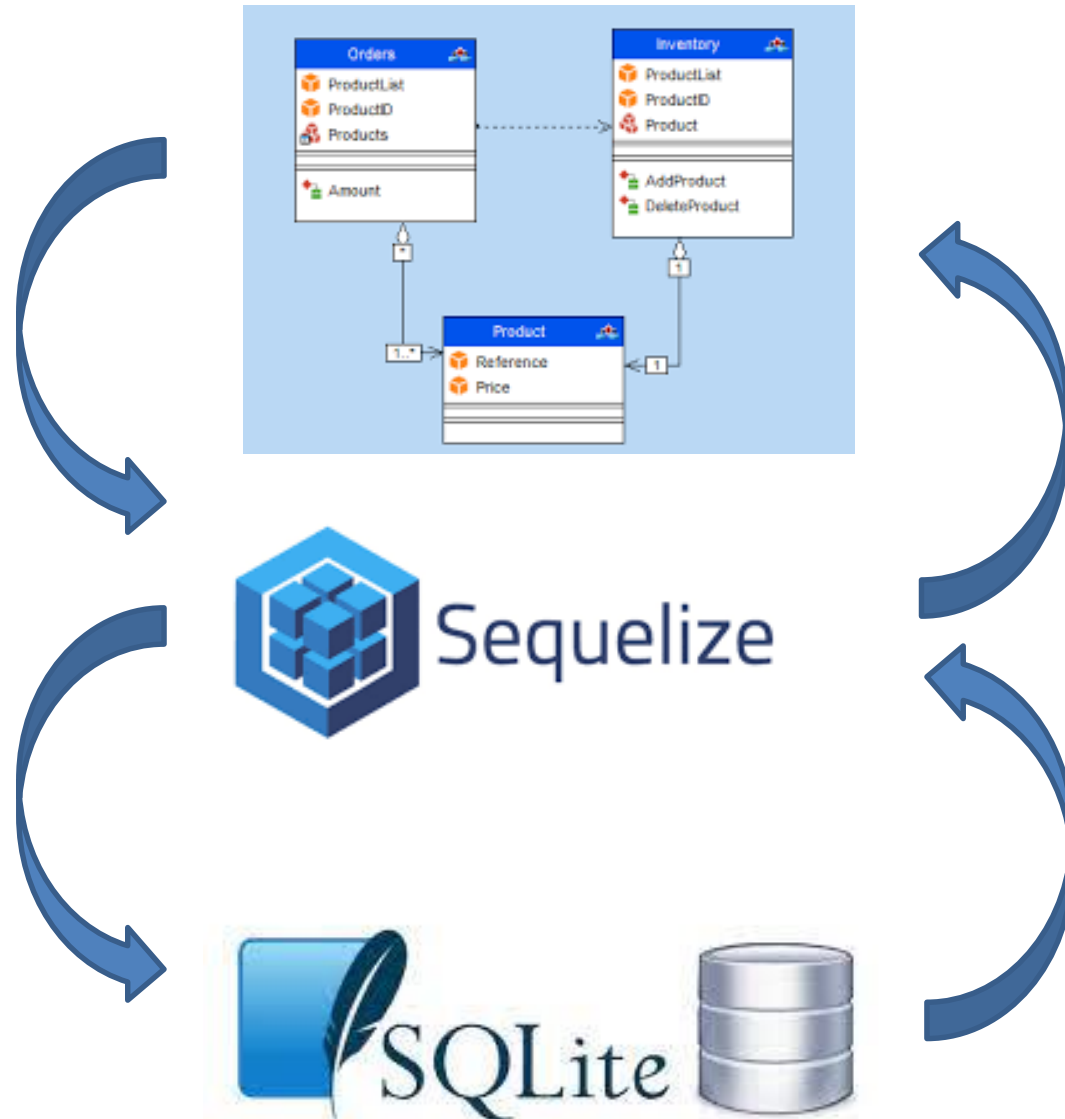
JS

{JSON} Placeholder

Caso práctico 04

- Desarrollar un script que permita recuperar los datos de los post publicados en la API JSON **Placeholder**
- Cargar los datos en un arreglo, identificar y codificar las clases de entidad que considere necesarias.
- Crear un archivo servicio.js que permita:
 - Consultar los post publicados por un usuario cuyo id es una valor entero
 - Obtener todos los post cuyo titulo contenga la palabra: 'eum'

ORM



SQLITE



SQLITE



DDL:

Lenguaje de **D**efinición de **D**atos
(CREATE, ALTER..)

Objetos de la base de datos

DML:

Lenguaje de **M**anipulación de **D**atos
(SELECT, INSERT, UPDATE,
DELETE)

Datos sobre las tablas

{JSON} Placeholder

Caso práctico 04 Sqlite

- Desarrollar un script que permita recuperar los datos de los post de una base datos: posts.db creada en Sqlite
- Cargar los datos en un arreglo, identificar y codificar las clases de entidad que considere necesarias.
- Crear un archivo servicio.js que permita:
 - Consultar los post publicados por un usuario cuyo id es una valor entero
 - Obtener todos los post cuyo titulo contenga la palabra: 'eum'

SQLITE

PASOS:

1. Instalar paquete `sqlite3` o descargar Sqlite desde el sitio de descarga
2. Crear la base de datos mediante un script:

```
import * as sqlite3 from 'sqlite3';

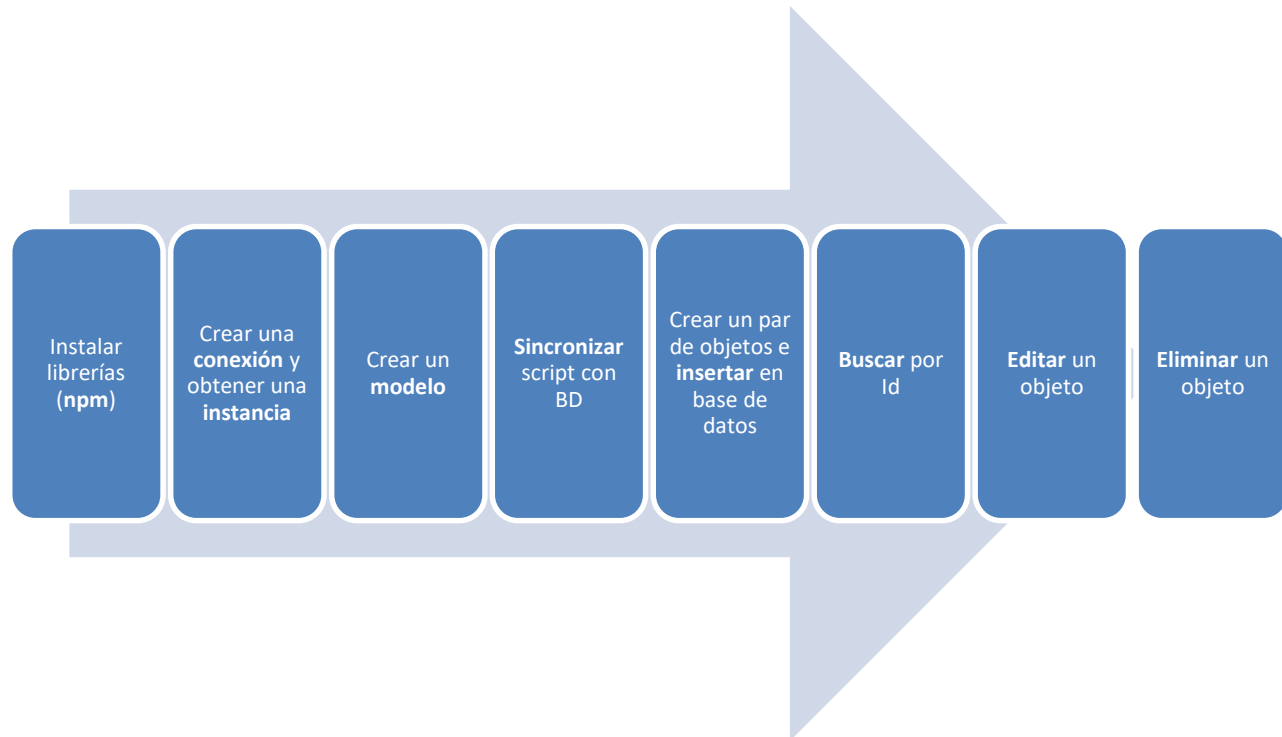
const db = new sqlite3.default.Database("./posts.db",sqlite3.OPEN_READWRITE,
(err)=>{
  if (err) return console.error(err.message);
})
```

3. Utilizar DB Browser for SQLite para crear una tabla y definir los campos
4. Poblar la BD desde la herramienta o mediante sentencias SQL
5. Ejecutar las consultas SQL necesarias según los requerimientos del ejercicio.

Sequelize



- Es un **ORM** (Object-Relational Mapping) de Node.js que permite interactuar con bases de datos relacionales utilizando objetos JavaScript. Herramienta útil para desarrolladores de Node.js que necesitan **interactuar con bases de datos relacionales de una manera más fácil**, segura y escalable.



Sequelize - Métodos



Nombre del método	Funcionalidad
new Sequelize(base, user, pass, {})	Permite crear una instancia de conexión con la BD
sequelize. define (nombre, {})	Permite definir un modelo que representa una tabla en la BD
sequelize. sync () .then(()=>{}) .catch(()=>{})	Permite sincronizar el modelo con la base de datos. Puede eliminar y recrear la tabla si se pasa el parámetro <i>force: true</i> como opción
Modelo.bulkCreate(datos, opciones);	Se utiliza para crear varios registros en la base de datos al mismo tiempo.
Modelo.create({})	Para crear un solo registro
Modelo.findOne({ where: { id: x } })	Permite recuperar un fila por Id

Sequelize - Métodos



Nombre del método	Funcionalidad
instancia. save()	Podemos actualizar una entidad que ya hemos recuperado previamente
instancia. destroy()	De forma análoga a la modificación, podemos eliminar una instancia que ha sido recuperada por una operación findOne().
Otro método comúnmente utilizado es Modelo. findAll({}) que permite recuperar todos los registros de una tabla según los criterios que se pasa como argumento.	

{JSON} Placeholder

Caso práctico 05 Sequelize

- Desarrollar un script que permita recuperar los datos de los post de una base datos: posts.db creada en Sqlite
- Inicializar la base con un script de inserte 5 posts en la base.
- Crear un archivo servicio.js que permita:
 - Editar el cuerpo y titulo de un post a partir de un Id.
 - Eliminar un post a partir de Id.
 - Listar todos los posts de un usuario cuyo Id se recibe como argumento.