

Taller Bases de datos YML

Jorge Esteban Herrera Jimenez – 833060

Corporación Universitaria Minuto de Dios

Bases de Datos Masivas

Ing. De Sistemas

William Alexander Matallana Porras

2025

1. Archivos .yaml vs json:

Archivos .yaml:

Los archivos yaml contienen el código fuente escrito en el lenguaje de programación YAML(no es un lenguaje de marcas). El código que contiene un archivo “.yaml” es legible por el ser humano y normalmente se usa para serializar datos. Con la ayuda de estos archivos se puede escribir y leer datos con independencia completa de los lenguajes de programación. Gracias a esto los archivos yaml pueden usarse junto con otros archivos contenedores de código fuente en varios lenguajes como podrían ser Java, PHP , C++ y varios más.

Archivos json:

Los archivos JSON es una herramienta que se usa en el desarrollo actual para compartir datos entre sistemas o entre distintas capas de una aplicación web. Sus siglas significan JavaScript Object Notation (JSON).

Estos archivos son un formato de almacenamiento e intercambio de datos que utiliza una estructura clave-valor que permite representar información de una manera legible tanto para humanos como para máquinas.

La estructura de un archivo JSON está compuesto por objetos {} y arreglos [], que son capaces de contener números, cadenas de texto, booleanos, null u otros objetos.

TABLA DE DIFERENCIAS

	JSON	YAML
Significado	Un formato para la serialización de los datos para hacer un intercambio de datos estructurados entre aplicaciones y servicios de software. Le da mas prioridad al uso de las aplicaciones	Es un formato de serialización de datos para intercambiar datos estructurados entre servicios y aplicaciones de software. Le da prioridad al uso humano antes que a las aplicaciones
Principales usos	Se usa de una manera amplia en plataformas, lenguajes, comunicaciones de software distribuido, aplicaciones web, archivos de configuración y APLI	Se usa en archivos de configuración en muchas herramientas y servicios de automatización, DevOps e infraestructura como código (IaC)
Tipos de datos	Números, cadena, matriz, booleano, nulo y objeto	Acepta todos los tipos de datos mediante colecciones de datos anidados que

		comprenden secuencias, escalares y asignaciones.
Admite objetos de datos como valores	Si	No

2. Docker-compose.yml – uso:

Es una herramienta que permite definir y gestionar aplicaciones multi-contenedor de forma sencilla. Con el Docker compose podremos describir la configuración del entorno de desarrollo en un archivo YAML, haciendo la especificación de los servicios, volúmenes y redes necesarios para la aplicación, seguidamente con un comando podremos crear y ejecutar todos los contenedores definidos en el archivo de configuración.

El uso de Docker Compose se hace cuando se necesita configurar un entorno de desarrollo que involucra varios servicios o contenedores. Esto nos permite definir la estructura de la aplicación y sus dependencias en un solo archivo YAML, haciendo más simple la creación y gestión del entorno de desarrollo.

Al usar el Docker Compose por encima de Docker, obtenemos la capacidad de definir volúmenes y redes personalizadas, la opción de escalar servicios y la facilidad de poder compartir y replicar el entorno de desarrollo con otros miembros del equipo. El uso de Docker Compose nos brinda una solución más eficiente para la configuración de aplicaciones multi-contenedor en nuestro entorno de desarrollo.

3. Como se crea un contenedor usando yaml:

Para crear un contenedor usando yml, creamos un archivo que se llame “docker-compose.yml” lo abrimos y debemos agregar el siguiente código:

```
► Run All Services
services:
  ► Run Service
  mysql:
    image: mysql:latest
    container_name: trabajoYML
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: "1234"
      MYSQL_DATABASE: "biblioteca"
      MYSQL_USER: "user"
      MYSQL_PASSWORD: "password"
    ports:
      - "3310:3306"
    volumes:
      - "mysql_db:/var/lib/mysql"
      - "./bd.sql:/docker-entrypoint-initdb.d/bd.sql"

volumes:
  mysql_db:
```

4. Crear base de datos que tenga tabla y le insertan registros:

```

CREATE DATABASE IF NOT EXISTS biblioteca;
USE biblioteca;

CREATE TABLE IF NOT EXISTS libros (
    id_libro INT AUTO_INCREMENT PRIMARY KEY,
    titulo VARCHAR (100) NOT NULL,
    autor VARCHAR (100) NOT NULL,
    anio_publicacion INT,
    genero VARCHAR (100),
    cantidad_disponible INT DEFAULT 1
);

CREATE TABLE IF NOT EXISTS usuarios(
    id_usuario INT AUTO_INCREMENT PRIMARY KEY,
    nombre VARCHAR (100) NOT NULL,
    apellido VARCHAR (100) NOT NULL,
    correo VARCHAR (100) UNIQUE NOT NULL,
    telefono VARCHAR (20)
);

CREATE TABLE IF NOT EXISTS prestamos(
    id_prestamo INT PRIMARY KEY AUTO_INCREMENT,
    id_usuario INT,
    id_libro INT,
    fecha_prestamo DATE,
    fecha_devolucion DATE,
    FOREIGN KEY (id_usuario) REFERENCES usuarios(id_usuario),
    FOREIGN KEY (id_libro) REFERENCES libros(id_libro)
);

INSERT INTO libros(titulo, autor, anio_publicacion, genero, cantidad_disponible) VALUES
('Cien años de soledad', 'Gabriel García Márquez', 1967, 'Realismo mágico', 3),
('1984', 'George Orwell', 1949, 'Distopía', 5),
('El principito', 'Antoine de Saint-Exupéry', 1943, 'Fábula', 2);

INSERT INTO usuarios(nombre, apellido, correo, telefono) VALUES
('Juan', 'Perez', 'juan@gmail.com', '3001234567' ),
('Maria', 'Lopez', 'maria@gmail.com', '3109876543'),
('Hector', 'Rodriguez', 'hector@gmail.com', '3158452342');

INSERT INTO prestamos(id_usuario, id_libro, fecha_prestamo, fecha_devolucion) VALUES
(1, 2, CURDATE(), NULL),
(2, 3, '2025-03-01', '2025-03-15');

```

Ahora iniciamos los servicios definidos del archivo docker-compose.yml con el siguiente comando:

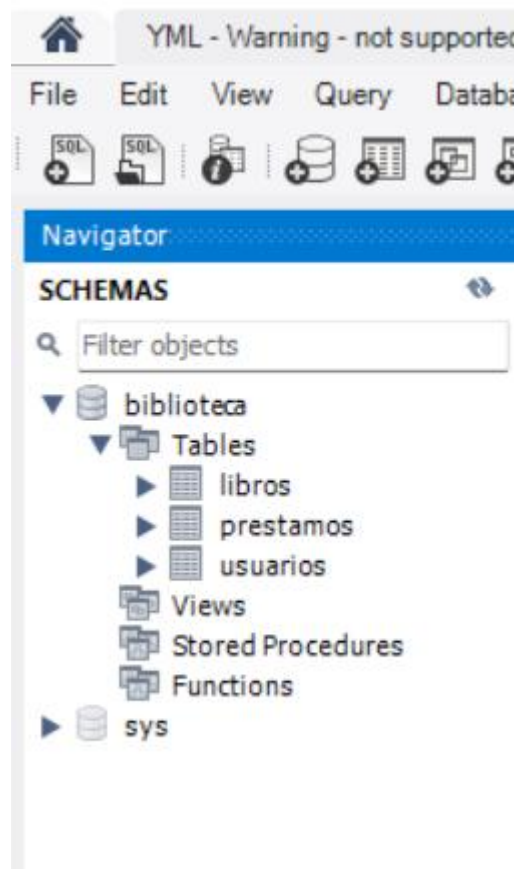
```
C:\Users\jorge\OneDrive\Escritorio\prueba_yaml>docker-compose up -d
[+] Running 3/3
✔ Network prueba_yaml_default      Created           0.3s
✔ Volume "prueba_yaml_mysql_db"    Created           0.1s
✔ Container trabajoYML             Started           0.9s
C:\Users\jorge\OneDrive\Escritorio\prueba_yaml>
```

Y revisamos que este corriendo en el Docker:

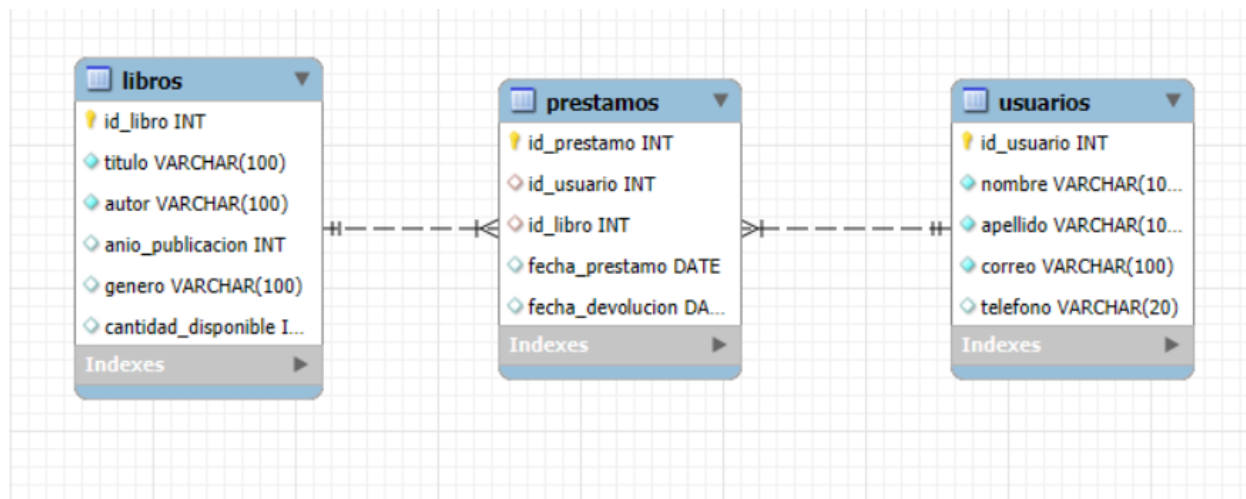
<input type="checkbox"/>	<input type="checkbox"/>	prueba_yaml	-	-	-	0.92%	8 minutes ago			
<input type="checkbox"/>	<input checked="" type="checkbox"/>	trabajoYML	290efcae7e12	mysql:latest	3310:3306	0.92%	8 minutes ago			

Showing 5 items

Seguidamente vamos a enlazar y revisar que aparezcan las tablas en workbench:



Para finalmente que no muestre el diagrama:



Referencias:

1. <https://www.online-convert.com/es/formato-de-archivo/yml>
 - [https://www.arsys.es/blog/archivo-json-que-es-y-para-que-sirve#:~:text=Un%20archivo%20JSON%20\(JavaScript%20Object,para%20humanos%20como%20para%20m%C3%A1quinas.](https://www.arsys.es/blog/archivo-json-que-es-y-para-que-sirve#:~:text=Un%20archivo%20JSON%20(JavaScript%20Object,para%20humanos%20como%20para%20m%C3%A1quinas.)
 - <https://aws.amazon.com/es/compare/the-difference-between-yaml-and-json/>
2. <https://imaginaformacion.com/tutoriales/que-es-docker-compose>