

Full Stack Bootcamp

Programación & Algorítmica & Proyectos Software

Introducción a BBDD

Bases de Datos avanzadas

Javascript avanzado & EcmaScript 6

MongoDB

Base de Datos – Qué es?

Según la Wikipedia:

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.

Base de Datos – Ejemplos

Ejercicio: Definir los datos necesarios para una biblioteca

Ejercicio: Definir los datos para una plataforma ecommerce.

- Tablas
- Tipos de datos
- Relaciones

Como se enlaza con la web?

Base de Datos – Ejemplos (II)

Ejercicio: Modelar los datos de IMDb

https://www.imdb.com/chart/top/?ref_=nv_mv_250

- Tablas
- Tipos de datos
- Relaciones

Base de Datos – Ejemplos (II)

- Almacenamiento de datos en servidor
- Guardado permanente
- Acceso a los datos en paralelo
- Inserción de datos
- Datos relacionados
- Búsqueda de datos -> Filtrado - filter()
- Ordenación de datos -> sort()



Base de Datos – Maria DB

Instalar Maria DB Server

<https://mariadb.org>

- Indicar usuario y contraseña
- Marcar resto de configuración por defecto
- Completar la instalación
- Inicio -> MySQL Client
- `mysql -u //user_name// -p -h //ip_address// //db_name//`

Base de Datos – Tipos

Sistemas transaccionales

- Basado en “transacciones”
- Aplicaciones en tiempo real
- Arquitectura normalizada
- Ejemplo: La mayoría de aplicaciones web

Ventajas / desventajas:

- Eliminación de bloqueos e interbloqueos (deadlocks)
- Difíciles de consultar desde el punto de vista analítico

Base de Datos – Tipos (II)

Sistemas analíticos:

- Para crear diagrama Relación-Entidad en estrella en sistema (OLAP)
- Usado masivamente para Data Warehouse
- Uso de procesos ETL

Ventajas / desventajas:

- Herramienta casi perfecta para la mejora continua de nuestro software
- No vale para uso del usuario final

Base de Datos – Tipos (III)

Sistemas relacionales:

- Ayudan a representar el modelo de datos con sus relaciones
- Operaciones de búsqueda e inserción controladas
- Uso de transacciones para gestionar los datos

Ventajas / desventajas:

- Fácil de entender y de mantener
- Con grandes volúmenes hace difícil su mantenimiento

Base de Datos – Tipos (IV)

Sistemas documentales:

- Para almacenar grandes volúmenes de datos
- Diferentes formatos para almacenar
- Uso de metadata

Ventajas / desventajas:

- Flexibilidad a la hora de almacenar y rapidez en la búsqueda
- Perdida de las relaciones y dificultad de mantenimiento

relevant.

MySQL – Historia y características



MySQL – Historia y características (II)

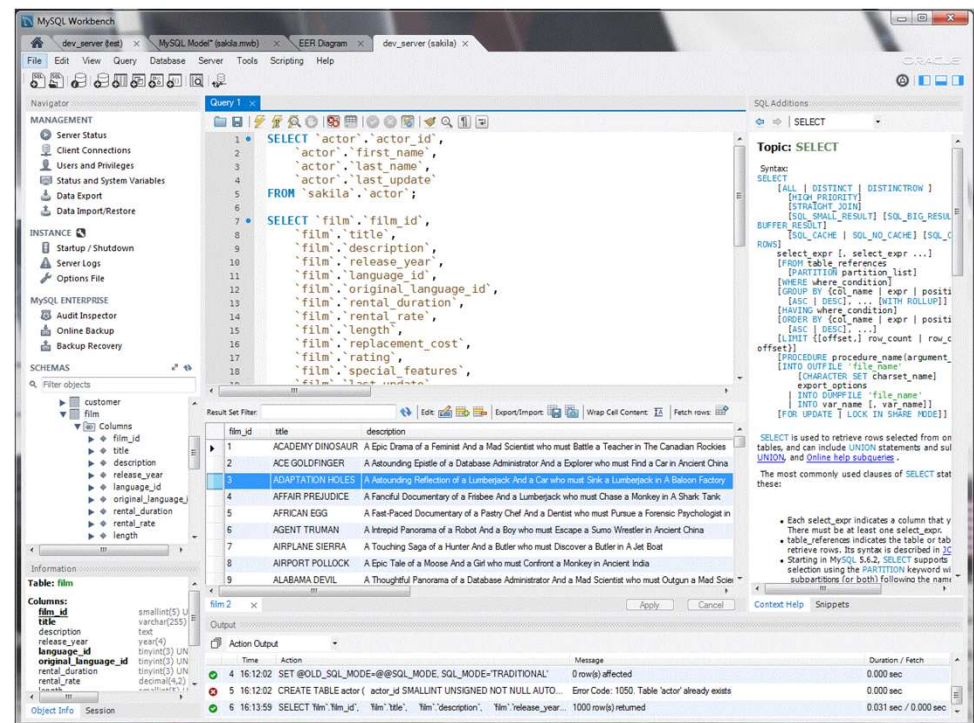
- Múltiples base de datos en un mismo servidor
- Base de datos relacionales
- Tipos de datos múltiples: entero, texto, decimal, flotante, etc.
- Basado en tablas y las relaciones entre dichas tablas



MySQL Workbench

Características:

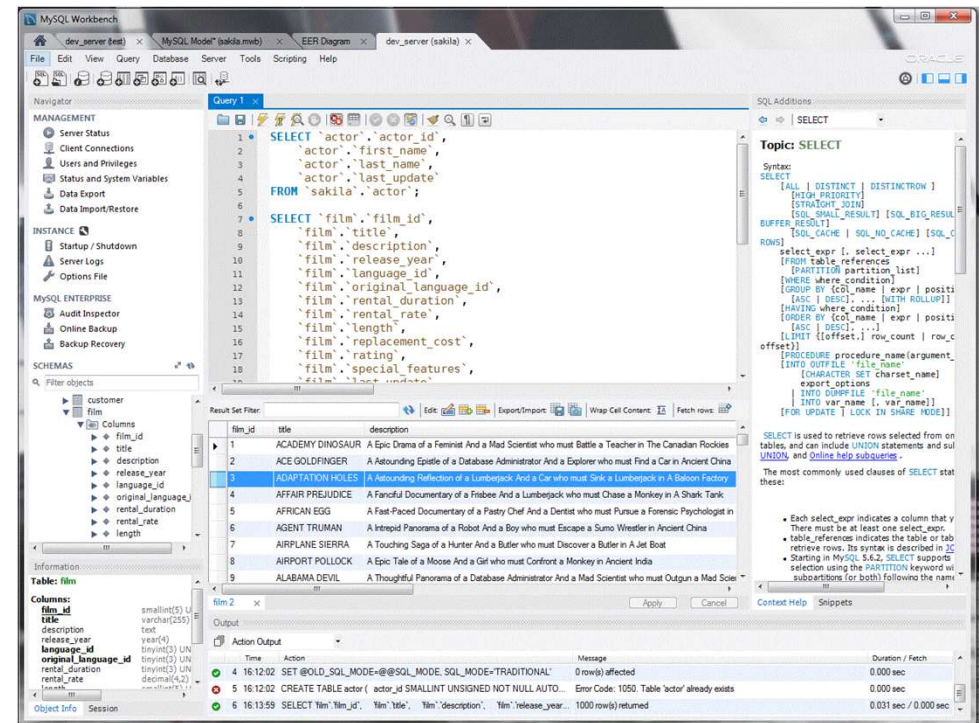
- Nos ayuda a visualizar los datos
- Permite crear consultas de forma fácil
- Podemos “debuggear” nuestros scripts de SQL
- Crear tablas y sus relaciones de forma visual



MySQL Workbench

Instalación:

- <https://www.mysql.com/products/workbench/>
- Conectar con nuestro servidor Maria DB
- Ver base de datos y tablas



MySQL – Crear Base de Datos

Comando CREATE DATABASE

CREATE DATABASE nombre_base_de_datos

Ejemplos:

CREATE DATABASE miBaseDeDatos

CREATE DATABASE test

CREATE DATABASE QWERTY_

MySQL - Tablas

Comando CREATE TABLE

CREATE TABLE nombre_tabla (nombre_columna TIPO_DATO, ...)

Ejemplos:

CREATE TABLE t1 (a INT);

CREATE TABLE t2 (b INT);

*CREATE TABLE student_tests (
name CHAR(10), test CHAR(10),
score TINYINT, test_date DATE
);*

Ejercicio: crear una base de datos de nombre imdb y crear su table de usuario.

MySQL – Integridad de datos

- Permite asegurar que los datos siempre guardan una consistencia
- Importante relación con el código que usa la base de datos, comunicación de errores entre ambas partes
- Usuario sin nombre o contraseña no puede existir -> obligar a que al guardar en base de datos sean correctos
- Uso de NOT NULL

CREATE TABLE t1 (a INT NOT NULL);

Ejercicio: borrar table usuario y crearla aplicando integridad de datos requerida.

MySQL – Tipos de datos

- **Numericos**
 - INT, TINYINT, SMALLINT, MEDIUMINT, BIGINT, FLOAT, DOUBLE, DECIMAL
- **Fecha y hora**
 - DATE, DATETIME, TIMESTAMP, TIME, YEAR
- **Cadenas**
 - CHAR, VARCHAR, BLOB, TINYBLOB, MEDIUMBLOB, LONGBLOB, ENUM

Ejercicio: crear varias tablas con varios tipos de datos

MySQL – Tipos de datos – El valor NULL

- Representa en valor vacío o no conocido
- Por ejemplo, un usuario crea una cuenta pero no indica sus hobbies porque no es un dato obligatorio

Ejercicio: crear varias tablas con varios tipos de datos, algunos null y otros not null

MySQL – INSERT

- Inserta nuevas filas en la tabla

INSERT INTO nombre_tabla VALUES (valor_columna, ...)

INSERT INTO t1 VALUES (1), (2), (3);

INSERT INTO t2 VALUES (2), (4);

INSERT INTO student_tests

(name, test, score, test_date) VALUES

('Chun', 'SQL', 75, '2012-11-05')

Ejercicio: Insertar filas en tabla de usuarios que ya hemos creado

Tip: usar sentencia `SELECT * FROM nombre_tabla` para ver datos ya creados

MySQL – UPDATE

- Actualiza datos de filas existentes en la tabla

UPDATE nombre_tabla SET nombre_columna = valor_nuevo

UPDATE books

SET title = 'Amerika'

Ejercicio: Actualizar datos de las filas de la tabla usuario que ya hemos creado

MySQL – DELETE

- Elimina datos de filas existentes en la tabla

DELETE FROM nombre_tabla WHERE condicion

DELETE FROM books

Ejercicio: Borrar filas de la tabla usuario que hemos creado anteriormente

MySQL – SELECT

- Selecciona filas existentes en la tabla

SELECT * FROM nombre_tabla WHERE condicion

SELECT * FROM Books

SELECT name, test FROM student_tests

Ejercicio: Seleccionar filas de la tabla Usuarios que ya hemos creado

MySQL – DISTINCT

- Selecciona filas distintas existentes en la tabla

SELECT DISTINCT nombre_columna FROM nombre_tabla

SELECT DISTINCT topic FROM Books

SELECT DISTINCT test FROM student_tests

Ejercicio: Crear filas nuevas en la tabla Contenidos, varios contenidos con mismo géneros. Selecciona las filas distintas por genero.

MySQL – COUNT

- Cuenta el numero de filas en una tabla

```
SELECT COUNT(nombre_columna) FROM nombre_tabla
```

```
SELECT COUNT(topic) FROM Books
```

```
SELECT COUNT(test) FROM student_tests
```

Ejercicio: Cuenta el numero de filas de la tabla Usuarios y Contenidos.

MySQL – GROUP BY

- Agrupa una función de agregado por la columna indicada

```
SELECT COUNT(nombre_columna) FROM nombre_tabla GROUP BY  
nombre_otra_columna
```

```
SELECT COUNT(name) FROM Books GROUP BY topic
```

```
SELECT COUNT(name) FROM student_tests GROUP BY test
```

Ejercicio: Cuenta el numero de filas de la tabla Usuarios y Contenidos agrupando por población y por genero.

MySQL – HAVING

- Filtra los resultados de GROUP BY usando el filtrado indicado

```
SELECT nombre_columna FROM nombre_tabla GROUP BY
nombre_otra_columna HAVING condicion
```

```
SELECT topic, count(name) as total FROM Books GROUP BY topic
HAVING total > 10
```

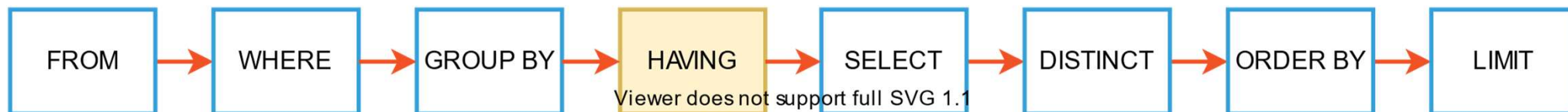
```
SELECT name, AVG(score) as total_score FROM student_tests
GROUP BY name HAVING total_score > 4
```

Ejercicio: Contar numero de contenidos por genero y solo mostrar cuando haya mas de 3 contenidos por genero.

MySQL – Orden de ejecucion

- Filtra los resultados de GROUP BY usando el filtrado indicado

**SELECT DISTINCT nombre_columna FROM nombre_tabla
GROUP BY nombre_otra_columna HAVING condición LIMIT
numero**



MySQL – LIMIT

- Limita el numero de filas a devolver por una sentencia SELECT

SELECT nombre_columna FROM nombre_tabla LIMIT numero

SELECT topic, count(name) FROM Books LIMIT 10

SELECT name, AVG(score) FROM student_tests LIMIT 5

Ejercicio: Mostrar las 5 filas de las tablas Usuarios y Contenidos.

MySQL – Vistas

- Muestra los resultados de una query / sentencia de MySQL
- Es de solo lectura y no se puede insertar datos
- Al insertar datos en la tabla fuente la vista se modifica

***CREATE VIEW nombre_vista AS <query> (p.e. SELECT name
from student_tests)***

Ejercicio: Crear una vista con el listado de contenidos que sean los 5 primeros de la lista (mayor score).

MySQL – ALTER TABLE

- Incluir nueva columna en una tabla existente

ALTER TABLE t3 ADD COLUMN (d INT)

- Modificar una columna existente en una tabla existente

ALTER TABLE t3 MODIFY COLUMN d VARCHAR

- Eliminar una columna existente en una tabla existente

ALTER TABLE t3 DROP COLUMN d

Ejercicio: Anadir nueva columna en la tabla Usuarios y Contenidos, Modificar las columnas creadas y luego eliminarlas.

MySQL – Secuencias

- Listado de números enteros en orden ascendente (1, 2, 3, ...)
- Se usan para generar números únicos en base de datos
- Permite hacer referencia a una fila en concreto
- Se suele usar con clave primaria

```
CREATE TABLE t3 (id INT UNSIGNED NOT NULL AUTO_INCREMENT, c  
INT)
```

Ejercicio: Incluir una columna ID en la tabla Usuarios y Contenidos.

MySQL – Secuencias (II)

- Reenumerar secuencia en una tabla

```
ALTER TABLE t3 DROP id;
```

```
ALTER TABLE t3 ADD id INT UNSIGNED NOT NULL AUTO_INCREMENT  
FIRST, ADD PRIMARY KEY (id);
```

- Comenzar la secuencia en un numero concreto

```
CREATE TABLE t3 (id INT UNSIGNED NOT NULL AUTO_INCREMENT = 100,  
c INT)
```

Ejercicio: Modificar las secuencias de las tablas de Usuarios y Contenidos.

MySQL – Recopilación

- **CREATE DATABASE test;**
- **CREATE TABLE t1 (a INT);**
- **INSERT INTO t1 VALUES (1);**
- **UPDATE books SET title = 'A';**
- **DELETE FROM Books;**
- **SELECT * FROM Books;**
- **SELECT DISTINCT topic FROM Books;**
- **SELECT COUNT(topic) FROM Books;**
- **SELECT COUNT(name) FROM Books GROUP BY topic**
- **SELECT topic, count(name) as total FROM Books GROUP BY topic HAVING total > 10**
- **SELECT topic, count(name) FROM Books LIMIT 10**
- **ALTER TABLE t3 ADD COLUMN (d INT)**
- **ALTER TABLE t3 MODIFY COLUMN d VARCHAR**
- **ALTER TABLE t3 DROP COLUMN d**
- **CREATE TABLE t3 (id INT UNSIGNED NOT NULL AUTO_INCREMENT, c INT)**

MySQL – Practicar

Ejercicio: Trabajar sobre la base de datos IMDb para completar los requisitos.

- *Mostrar las 5 películas con mayor score*
- *Listado del numero de películas por genero*
- *Listado de los géneros disponibles*
- *Media de score por genero*
- *Tabla de votaciones de usuarios*

MySQL – Practicar

Ejercicio: Crear base de datos para eCommerce.

- *Trabajar en grupo para el diseño de la base de datos*
- *Crear todas las tablas requeridas*
- *Usar columna ID para cada tabla*
- *Crear las sentencias para obtener datos necesarios*