

Full Stack Bootcamp

Programación & Algorítmica & Proyectos Software

Introducción a BBDD

Bases de Datos avanzadas

Javascript avanzado & EcmaScript 6

MongoDB

MySQL – Subconsultas

- Consulta dentro de otra consulta

*SELECT * FROM ventas WHERE producto_id = (SELECT id FROM productos where nombre like '%Correa%')*

- Otras clausulas muy practicas: *ANY, ALL, IN, NOT, INTO y EXISTS*

MySQL – Subconsultas (Tabla Jugadores)

```
CREATE TABLE jugador(  
    id INT (11) NOT NULL AUTO_INCREMENT,  
    nombre VARCHAR(80) NOT NULL,  
    edad INT NOT NULL,  
    es_famoso INT NOT NULL DEFAULT 0,  
    id_videojuego INT NOT NULL,  
    FOREIGN KEY(id_videojuego) REFERENCES videojuego(id),  
    fecha_registro DATE NULL  
);
```

MySQL – Subconsultas (Tabla Videojuegos)

```
CREATE TABLE videojuego (  
    id INT (11) NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    titulo VARCHAR(100) NOT NULL,  
    lenguaje VARCHAR(5) NOT NULL,  
    resumen TEXT NULL,  
    nota DECIMAL (3,1),  
    total_jugadores INT DEFAULT 0  
);
```

MySQL – Ejercicios Subconsultas

- Recuperar el nombre de los jugadores que juegan a un videojuego que al menos juega 2 jugadores
- Realizar la consulta anterior con ANY
- Realizar la consulta anterior con SOME
- Recuperar todos los jugadores que juegan a un juego en Español
- Recuperar nombre de los jugadores que tienen la mayor edad. Obtener primero el jugador de mayor edad, luego los 3 jugadores de mayor y edad y por ultimo los 5 jugadores
- Seleccionar todos los videojuegos de uno de los lenguajes a los que juega algún jugador famoso.

MySQL – Tipos de Join

- Permite hacer consultas en múltiples tablas
- Se basa en conectar tablas por las columnas de identificadore (ID)
 - Tabla Productos con ID
 - Tablas Ventas con id_producto
 - Nos permite indicar producto.id = ventas.id_productos
- Varios tipos de join
 - Inner join: `SELECT * FROM productos p JOIN ventas v ON v.id_producto = p.id`
 - Right outer join: `SELECT * FROM productos p RIGHT JOIN ventas v ON v.id_producto = p.id`
 - Left outer join: `SELECT * FROM productos p LEFT JOIN ventas v on v.id_producto = p.id`

MySQL – Ejercicios

- Obtener el listado de ventas con nombre de usuario y el tipo de pago realizado
- Sacar por consola lista de productos con su nombre y la cantidad total vendida.
- Sacar por consola Lista de productos con el average del rating y el total de veces que se ha comprado cada producto
- Sacar por consola cual es el producto mas vendido y nombre de usuarios han comprado dicho producto
- Sacar por consola el producto con su nombre y tipo que mayor ganancia ha generado
- Sacar por consola el nombre de usuario que mas artículos ha comprado
- Sacar por consola el nombre de usuario que se ha gastado mas dinero. Incluir el símbolo €.

MySQL – Unique

- UNIQUE se aplica a una columna de una tabla de nuestra base de datos
- Indica que esa columna no puede tener el mismo valor en dos filas

ALTER TABLE t1 ADD UNIQUE (columna)

Ejercicio: En nuestra base de datos indicad que la columna email de la tabla usuarios es de tipo Unique. Añadir filas a nuestra con un mismo valor de email y ver resultados.

MySQL – Triggers

- Programa que se ejecuta cuando se cumple una circunstancia o se produce un evento determinado
- En MySQL los eventos disponibles son INSERT, UPDATE o DELETE
- Existen triggers a nivel fila o a nivel consulta, MySQL solo soporta triggers a nivel fila
- Ejemplos:
 - Llevar una auditoria de los cambios de mi base de datos
 - Calcular el rate de un producto tras cada votación

MySQL – Triggers (II)

- Para acceder a los datos de la fila que se esta insertando, actualizando o eliminando tenemos OLD y NEW tal que:

Trigger Event	OLD	NEW
INSERT	No	Yes
UPDATE	Yes	Yes
DELETE	Yes	No

MySQL – Triggers - Ejercicio

- Sintaxis:

```
CREATE TRIGGER before_employee_update
  BEFORE UPDATE ON employees
  FOR EACH ROW
  INSERT INTO employees_audit
  SET action = 'update',
      employeeNumber = OLD.employeeNumber,
      lastname = OLD.lastname,
      changedat = NOW();
```

Ejercicio: Implementar un trigger en nuestra tabla ventas para que baje el stock del producto comprador en la tabla productos

MySQL – Procedimientos almacenados

- Es un trozo de programa que se ejecuta cuando se llama (funciones)
- Similares a function en JavaScript pero se ejecuta a nivel de base de datos

```
CREATE PROCEDURE GetAllProducts()  
BEGIN  
    SELECT * FROM products;  
END
```

Ejercicio: crear un procedimiento almacenado para mostrar los productos mas vendidos