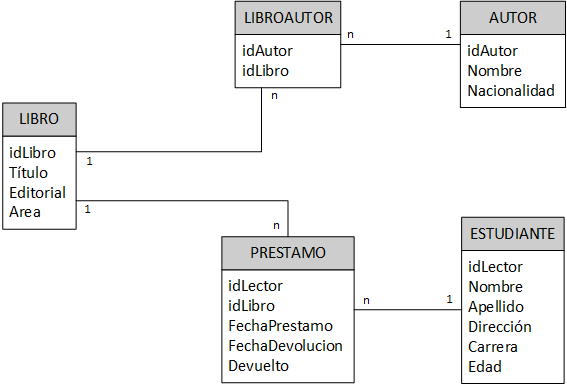
Bases de Datos Relacionales

 Consultas SQL Avanzadas  
// Práctica Grupal  
****Ejercicio 1

Se tiene el siguiente DER que corresponde al esquema que presenta la base de datos de una “biblioteca”.



En base al mismo, **plantear las consultas SQL** para resolver los siguientes requerimientos:

1. Listar los datos de los autores.

SELECT \* FROM autor;



1. Listar nombre y edad de los estudiantes

SELECT nombre, edad FROM estudiante;



1. ¿Qué estudiantes pertenecen a la carrera informática?

SELECT nombre, apellido

FROM estudiante

WHERE carrera = 'Informática';



1. ¿Qué autores son de nacionalidad francesa o italiana?

SELECT \*

FROM autor

WHERE nacionalidad IN ('francesa', 'italiana');



1. ¿Qué libros no son del área de internet?

SELECT \*

FROM libro

WHERE area != 'internet';



1. Listar los libros de la editorial Salamandra.

SELECT \*

FROM libro

WHERE editorial = 'Salamandra';



1. Listar los datos de los estudiantes cuya edad es mayor al promedio.

SELECT \*

FROM estudiante

WHERE edad > (

SELECT AVG(edad) FROM estudiante

);



1. Listar los nombres de los estudiantes cuyo apellido comience con la letra G.

SELECT nombre

FROM estudiante

WHERE apellido LIKE 'G%';



1. Listar los autores del libro “El Universo: Guía de viaje”. (Se debe listar solamente los nombres).

/\* Usando JOINS \*/

SELECT autor.nombre

FROM libro\_autor

INNER JOIN libro ON libro\_autor.id\_libro = libro.id\_libro

INNER JOIN autor ON libro\_autor.id\_autor = autor.id\_autor

WHERE libro.titulo = 'El Universo: Guía de viaje';



1. ¿Qué libros se prestaron al lector “Filippo Galli”?

/\* Usando Joins \*/

SELECT libro.\*

FROM prestamo

INNER JOIN estudiante ON prestamo.id\_lector = estudiante.id\_lector

INNER JOIN libro ON prestamo.id\_libro = libro.id\_libro

WHERE estudiante.nombre = 'Filippo' AND estudiante.apellido = 'Galli';



1. Listar el nombre del estudiante de menor edad.

SELECT nombre

FROM estudiante

WHERE edad = (

SELECT MIN(edad) FROM estudiante

);



1. Listar nombres de los estudiantes a los que se prestaron libros de Base de Datos.

/\* Usando Joins \*/

SELECT estudiante.nombre, estudiante.apellido

FROM prestamo

INNER JOIN libro ON prestamo.id\_libro = libro.id\_libro

INNER JOIN estudiante ON prestamo.id\_lector = estudiante.id\_lector

WHERE libro.area = 'Base de Datos';



1. Listar los libros que pertenecen a la autora J.K. Rowling.

SELECT libro.\*

FROM libro\_autor

INNER JOIN libro ON libro\_autor.id\_libro = libro.id\_libro

INNER JOIN autor ON libro\_autor.id\_autor = autor.id\_autor

WHERE autor.nombre = 'J.K. Rowling';



1. Listar títulos de los libros que debían devolverse el 16/07/2021.

SELECT libro.titulo

FROM prestamo

INNER JOIN libro ON prestamo.id\_libro = libro.id\_libro

WHERE prestamo.fecha\_devolucion = '2021-07-16';



****Ejercicio 2

Implementar la base de datos en ***PHPMyAdmin*** o ***MySQL Workbench***, cargar cinco registros en cada tabla y probar algunas consultas planteadas en el Ejercicio 1.

1. Implementación de la base de datos:

CREATE DATABASE biblioteca;

USE biblioteca;

CREATE TABLE autor (

id\_autor INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

nacionalidad VARCHAR(50) NOT NULL

);

CREATE TABLE libro (

id\_libro INT AUTO\_INCREMENT PRIMARY KEY,

titulo VARCHAR(100),

editorial VARCHAR(50),

area VARCHAR(50)

);

CREATE TABLE estudiante (

id\_lector INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50),

apellido VARCHAR(50),

direccion VARCHAR(100),

carrera VARCHAR(50),

edad INT

);

CREATE TABLE libro\_autor (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

id\_autor INT NOT NULL,

id\_libro INT NOT NULL,

FOREIGN KEY (id\_autor) REFERENCES autor(id\_autor),

FOREIGN KEY (id\_libro) REFERENCES libro(id\_libro)

);

CREATE TABLE prestamo (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

id\_libro INT NOT NULL,

id\_lector INT NOT NULL,

fecha\_prestamo DATE,

fecha\_devolucion DATE,

devuelto BOOLEAN,

FOREIGN KEY (id\_libro) REFERENCES libro(id\_libro),

FOREIGN KEY (id\_lector) REFERENCES estudiante(id\_lector)

);



1. Llenado de tablas con 5 registros cada una:

INSERT INTO autor (nombre, nacionalidad) VALUES

('Gabriel García Márquez', 'Colombia'),

('Mario Vargas Llosa', 'Perú'),

('J.K. Rowling', 'Reino Unido'),

('Paulo Coelho', 'Brasil'),

('Isabel Allende', 'Chile');

INSERT INTO libro (titulo, editorial, area) VALUES

('Cien años de soledad', 'Editorial Sudamericana', 'Literatura'),

('La ciudad y los perros', 'Seix Barral', 'Literatura'),

('Harry Potter y la piedra filosofal', 'Bloomsbury Publishing', 'Literatura'),

('El alquimista', 'HarperCollins', 'Autoayuda'),

('La casa de los espíritus', 'Plaza & Janés', 'Literatura');

INSERT INTO estudiante (nombre, apellido, direccion, carrera, edad) VALUES

('Juan', 'Gómez', 'Av. Principal 123', 'Ingeniería de Sistemas', 21),

('María', 'Martínez', 'Calle Los Pinos 456', 'Informática', 19),

('Pedro', 'García', 'Av. Las Palmas 789', 'Ingeniería Civil', 23),

('Ana', 'Rodríguez', 'Calle Las Rosas 1010', 'Ingeniería de Sistemas', 22),

('Luis', 'González', 'Av. La Marina 222', 'Informática', 20);

INSERT INTO libro\_autor (id\_autor, id\_libro) VALUES

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5);

INSERT INTO prestamo (id\_libro, id\_lector, fecha\_prestamo, fecha\_devolucion, devuelto) VALUES

(1, 1, '2021-03-01', '2021-03-08', true),

(2, 2, '2022-01-15', '2022-01-22', true),

(3, 3, '2022-02-10', '2022-02-17', false),

(4, 4, '2021-11-20', '2021-11-27', true),

(5, 5, '2023-01-01', '2023-01-08', false);



1. Ejecución de algunas de las consultas propuestas (item 1).

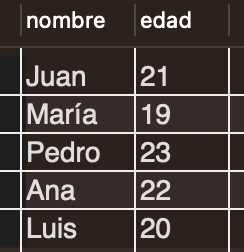
SELECT \* FROM autor;





SELECT nombre, edad FROM estudiante;





SELECT \* FROM libro WHERE area != 'Internet';



