CREATE DATABASE IF NOT EXISTS centro;

USE centro;

-- Tabla persona (base)

CREATE TABLE persona (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

apellidos VARCHAR(150) NOT NULL,

direccion VARCHAR(255),

poblacion VARCHAR(100),

dni VARCHAR(20) UNIQUE,

fecha\_nacimiento DATE,

cp VARCHAR(10),

telefono VARCHAR(20)

);

-- Alumno y Profesor (relación 1:1 con persona)

CREATE TABLE alumno (

id INT AUTO\_INCREMENT PRIMARY KEY,

persona\_id INT NOT NULL UNIQUE,

FOREIGN KEY (persona\_id) REFERENCES persona(id) ON DELETE CASCADE

);

CREATE TABLE profesor (

id INT AUTO\_INCREMENT PRIMARY KEY,

persona\_id INT NOT NULL UNIQUE,

FOREIGN KEY (persona\_id) REFERENCES persona(id) ON DELETE CASCADE

);

-- Curso

CREATE TABLE curso (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL

);

-- Asignatura

CREATE TABLE asignatura (

id INT AUTO\_INCREMENT PRIMARY KEY,

codigo VARCHAR(50) NOT NULL UNIQUE,

nombre VARCHAR(150) NOT NULL,

horas\_semana INT,

profesor\_id INT NOT NULL,

curso\_id INT NOT NULL,

FOREIGN KEY (profesor\_id) REFERENCES profesor(id),

FOREIGN KEY (curso\_id) REFERENCES curso(id)

);

-- Aula

CREATE TABLE aula (

id INT AUTO\_INCREMENT PRIMARY KEY,

codigo VARCHAR(50) NOT NULL UNIQUE,

piso INT,

num\_pupitres INT

);

-- Relación asignatura <-> aula con horario (imparte)

CREATE TABLE imparte (

id INT AUTO\_INCREMENT PRIMARY KEY,

asignatura\_id INT NOT NULL,

aula\_id INT NOT NULL,

mes TINYINT NOT NULL,

dia TINYINT NOT NULL,

hora TIME NOT NULL,

FOREIGN KEY (asignatura\_id) REFERENCES asignatura(id),

FOREIGN KEY (aula\_id) REFERENCES aula(id)

);

-- Matricula (alumno - asignatura)

CREATE TABLE matricula (

id INT AUTO\_INCREMENT PRIMARY KEY,

alumno\_id INT NOT NULL,

asignatura\_id INT NOT NULL,

nota DECIMAL(4,2) NULL,

incidencias TEXT NULL,

FOREIGN KEY (alumno\_id) REFERENCES alumno(id),

FOREIGN KEY (asignatura\_id) REFERENCES asignatura(id)

);

-- Tutoria (curso -> profesor) (1 curso tiene 1 tutor, 1 profesor tutor de a lo sumo 1 curso)

CREATE TABLE tutoria (

curso\_id INT PRIMARY KEY,

profesor\_id INT UNIQUE,

FOREIGN KEY (curso\_id) REFERENCES curso(id),

FOREIGN KEY (profesor\_id) REFERENCES profesor(id)

);

Inserts:

INSERT INTO persona (nombre, apellidos, direccion, poblacion, dni, fecha\_nacimiento, cp, telefono)

VALUES

('Ana','Gomez','Calle 1','Ciudad','DNI001','2005-02-10','1000','111-1111'),

('Luis','Martinez','Calle 2','Ciudad','DNI002','1980-06-20','1000','222-2222'),

('María','Perez','Calle 3','Ciudad','DNI003','1990-07-15','1000','333-3333');

INSERT INTO alumno (persona\_id) VALUES (1);

INSERT INTO profesor (persona\_id) VALUES (2), (3);

INSERT INTO curso (nombre) VALUES ('1º A'), ('2º B');

INSERT INTO asignatura (codigo, nombre, horas\_semana, profesor\_id, curso\_id)

VALUES ('MAT101','Matemáticas I',4, 1, 1),

('FIS101','Física I',3, 2, 1);

INSERT INTO aula (codigo,piso,num\_pupitres) VALUES ('A1',1,30),('B2',2,25);

INSERT INTO imparte (asignatura\_id,aula\_id,mes,dia,hora)

VALUES (1,1,9,15,'09:00:00'), (1,2,9,17,'11:00:00'), (2,1,9,16,'10:00:00');

INSERT INTO matricula (alumno\_id,asignatura\_id,nota,incidencias)

VALUES (1,1,8.5,NULL);

Consultas:  
  
SELECT p.id AS persona\_id, p.nombre, p.apellidos, a.id AS asignatura\_id, a.nombre AS asignatura

FROM persona p

JOIN alumno al ON al.persona\_id = p.id

JOIN matricula m ON m.alumno\_id = al.id

JOIN asignatura a ON a.id = m.asignatura\_id;

SELECT a.nombre, COUNT(m.alumno\_id) AS num\_alumnos

FROM asignatura a

LEFT JOIN matricula m ON m.asignatura\_id = a.id

GROUP BY a.id

HAVING COUNT(m.alumno\_id) > 0;

SELECT nombre, apellidos, fecha\_nacimiento,

TIMESTAMPDIFF(YEAR, fecha\_nacimiento, CURDATE()) AS edad

FROM persona;

SELECT \* FROM persona WHERE apellidos LIKE 'Pe%';

SELECT p.nombre, p.apellidos, m.nota

FROM matricula m

JOIN alumno al ON al.id = m.alumno\_id

JOIN persona p ON p.id = al.persona\_id

WHERE m.nota IS NULL;

SELECT pr.id, p.nombre, p.apellidos

FROM profesor pr

JOIN persona p ON p.id = pr.persona\_id

WHERE pr.id NOT IN (SELECT profesor\_id FROM tutoria WHERE profesor\_id IS NOT NULL);

SELECT a.nombre, AVG(m.nota) AS media

FROM asignatura a

LEFT JOIN matricula m ON m.asignatura\_id = a.id

GROUP BY a.id;

SELECT \* FROM asignatura AS asg

WHERE asg.horas\_semana > (SELECT AVG(horas\_semana) FROM asignatura);