

## Práctica SQL

### MER

Profesor (id\_profesor, DNI, nombre, direccion\_CP, direccion\_calle, direccion\_nro)

Teléfono (id\_telefono, telefono, id\_profesor)

Módulo (cod\_modulo, nombre, id\_profesor)

Matriculado (id\_matriculado, cod\_modulo, cod\_alumno)

Alumno (nro\_expediente, nombre, apellido, fecha\_nac, nro\_expediente\_delegado)

#Creamos la base de datos

**CREATE DATABASE IF NOT EXISTS** instituto;

**USE** instituto;

#Creamos las tablas

**CREATE TABLE** Profesor (  
id\_profesor **INT PRIMARY KEY AUTO\_INCREMENT**,  
dni **INT UNIQUE NOT NULL**,  
nombre **VARCHAR(20) NOT NULL**,  
direccion\_cp **INT NOT NULL**,  
direccion\_calle **VARCHAR(40) NOT NULL**,  
direccion\_nro **INT NOT NULL**);

**CREATE TABLE** Modulo (  
cod\_modulo **INT PRIMARY KEY AUTO\_INCREMENT**,  
nombre **VARCHAR(20) NOT NULL**,  
id\_profesor **INT NOT NULL**,  
**FOREIGN KEY** (id\_profesor) **REFERENCES** Profesor(id\_profesor));

**CREATE TABLE** Alumno (  
nro\_expediente **INT PRIMARY KEY AUTO\_INCREMENT**,  
nombre **VARCHAR(20) NOT NULL**,  
apellido **VARCHAR(20) NOT NULL**,  
fecha\_nac **DATE NOT NULL**,  
nro\_expediente\_delegado **INT**,  
**FOREIGN KEY** (nro\_expediente\_delegado) **REFERENCES** Alumno(nro\_expediente));

**CREATE TABLE** Matriculado (  
id\_matriculado **INT PRIMARY KEY AUTO\_INCREMENT**,  
cod\_modulo **INT NOT NULL**,  
cod\_alumno **INT NOT NULL**,  
**FOREIGN KEY** (cod\_modulo) **REFERENCES** Modulo(cod\_modulo),  
**FOREIGN KEY** (cod\_alumno) **REFERENCES** Alumno(nro\_expediente));

# Insertamos datos

**INSERT INTO** Profesor (dni, nombre, direccion\_cp, direccion\_calle, direccion\_nro)  
**VALUES** (13567890, "Daniel", 1714, "Av. Ratti", 2560),  
(28570434, "Marisa", 1712, "Gob. Arias", 1756),  
(36032333, "Javier", 1740, "Av. San Martin", 3569);

**INSERT INTO** Modulo (nombre, id\_profesor)  
**VALUES** ("Programación I", 1),  
("Programación 2", 1),  
("Matemática", 2),  
("Álgebra", 2),  
("Base de Datos", 3);

**INSERT INTO** Alumno (nro\_Expediente, nombre, apellido, fecha\_nac)  
**VALUES** (1020, "Lucía", "Jimenez", "2000-08-23"),

```
(1021, "Lautaro", "Robles", "1999-04-25");
```

```
INSERT INTO Alumno (nombre, apellido, fecha_nac, nro_expediente_delegado)
VALUES ("Sofia", "Alvarez", "2003-10-14", 1021),
("Marcos", "Diaz", "2000-05-28", 1021),
("Mario", "Gomez", "2001-06-30", 1020),
("Ivana", "Robles", "2002-11-20", 1020),
("Ramiro", "Casablanca", "2002-02-05", 1020);
```

```
INSERT INTO Matriculado (cod_modulo, cod_alumno)
VALUES (1, 1020),
(1, 1024),
(1, 1025),
(1, 1026),
(2, 1021),
(2, 1022),
(2, 1023),
(3, 1024),
(3, 1026),
(4, 1021),
(4, 1025),
(5, 1020);
```

#1) Cree la tabla Telefono.

```
CREATE TABLE Telefono (
id_telefono INT PRIMARY KEY AUTO_INCREMENT,
telefono VARCHAR(12) NOT NULL, #Puede ser INT también
id_profesor INT NOT NULL,
FOREIGN KEY (id_profesor) REFERENCES profesor(id_profesor));
```

#2) Ingrese registros en la tabla Telefono (al menos un profesor debe tener más de un teléfono).

```
INSERT INTO Telefono (telefono, id_profesor)
VALUES ("1122334455", 1),
("1155443322", 2),
("1199887766", 2);
```

#3) Cambie la calle y el número de la dirección del profesor con id 3 a Av. Rosas y 3405, respectivamente.

```
UPDATE profesor
SET direccion_calle = "Av. Rosas",
direccion_nro = 3405
WHERE id_profesor = 3;
```

#4) Listar el nombre y apellido de los alumnos que hayan nacido entre el 2001 y el 2003, ordenados por apellido de manera ascendente y por nombre de manera descendente.

```
SELECT nombre, apellido FROM alumno
WHERE fecha_nac BETWEEN "2001-01-01" AND "2003-12-31"
ORDER BY apellido ASC, nombre DESC;
```

#5) Listar el nombre de los profesores en el que su calle empieza con "av".

```
SELECT nombre FROM profesor
WHERE direccion_calle LIKE 'av%';
```

#6) Listar el id y nombre de los alumnos que son delegados.

```
SELECT nro_expediente, nombre FROM alumno
WHERE nro_expediente_delegado IS NULL;
```

#7) Listar el nombre y apellido de los alumnos que hayan nacido en el mes de Junio.

```
SELECT nombre, apellido FROM alumno
WHERE MONTH(fecha_nac) = 6;
```

#8) Listar los datos de los alumnos que tengan como delegada a Lucía.

```
SELECT A.* FROM alumno A
JOIN alumno AL ON A.nro_expediente_delegado = AL.nro_expediente
WHERE AL.nombre = "Lucía";
```

#9) Listar en una columna el nombre completo del delegado junto al nombre de quién es delegado.

```
SELECT CONCAT(D.nombre, " ", D.apellido) AS nombre_completo_delegado, AL.nombre
FROM alumno D
JOIN alumno AL ON D.nro_expediente = AL.nro_expediente_delegado;
```

#10) Listar el nombre de los profesores junto con el nombre del/los módulo/s que imparten.

```
SELECT P.nombre, M.nombre FROM profesor P
JOIN modulo M ON P.id_profesor = M.id_profesor;
```

#11) Listar los nombres de los alumnos que están matriculados en el módulo "Programación I".

```
SELECT A.nombre FROM alumno A
JOIN Matriculado MA ON A.nro_expediente = MA.cod_alumno
JOIN modulo M ON MA.cod_modulo = M.cod_modulo
WHERE M.nombre = "Programación I";
```

#12) Listar los nombres de los profesores junto con los números de su/s teléfono/s.

```
SELECT P.nombre, T.telefono FROM profesor P
JOIN Telefono T ON T.id_profesor = P.id_profesor;
```

#13) Listar el número de expediente y el nombre de los alumnos que estén matriculados en un módulo impartido por Javier.

```
SELECT A.nro_expediente, A.nombre FROM alumno A
JOIN Matriculado MA ON A.nro_expediente = MA.cod_alumno
JOIN modulo M ON MA.cod_modulo = M.cod_modulo
JOIN profesor P ON P.id_profesor = M.id_profesor
WHERE P.nombre = "Javier";
```

#14) Listar el id del profesor que tenga al menos un teléfono.

```
SELECT DISTINCT id_profesor FROM Telefono;
```