

UNIVERSIDAD PRIVADA “FRANZ TAMAYO”
FACULTAD DE INGENIERIA
CARRERA DE INGENIERÍA DE SISTEMAS



ESTUDIANTE:

EVER TICONA HUALLPA

CARRERA:

INGENIERÍA DE SISTEMAS

DOCENTE:

ING. WILLIAM RODDY BARRA PAREDES

ASIGNATURA:

BASE DE DATOS II

EL ALTO - LA PAZ

2022

```

CREATE DATABASE DIAGNOSTICO;
USE DIAGNOSTICO;

CREATE TABLE ESTUDIANTE
(
    id_est INTEGER AUTO_INCREMENT PRIMARY KEY NOT NULL,
    nombre VARCHAR(50) NOT NULL,
    apellido VARCHAR(50) NOT NULL,
    edad INTEGER
);

CREATE TABLE MATERIAS
(
    id_mat INTEGER AUTO_INCREMENT PRIMARY KEY NOT NULL,
    materia VARCHAR(50),
    cod_mat VARCHAR(10)
);

INSERT INTO ESTUDIANTE (nombre, apellido, edad)
VALUES ('EVER', 'TICONA', 21),
('PEDRO', 'MARTINES', 20);

INSERT INTO MATERIAS (materia, cod_mat)
VALUES ('MATEMATICAS', 'MAT-450'),
('FISICA', 'FIS-201');

SELECT *FROM MATERIAS;
DROP DATABASE DIAGNOSTICO;

CREATE DATABASE UNIVERSIDAD;
USE UNIVERSIDAD;

CREATE TABLE estudiantes
(
    id_est INTEGER AUTO_INCREMENT PRIMARY KEY NOT NULL,
    nombres VARCHAR(100),
    apellidos VARCHAR(100),
    edad INTEGER,
    fono INTEGER,
    email VARCHAR(50)
);

INSERT INTO estudiantes (nombres, apellidos, edad, fono, email)
VALUES
('EVER', 'TICONA', 21, 60614566, 'everticonal23@gmail.com'),
('PEDRO', 'MARTINES', 20, 60614567, 'pedromartines123@gmail.com'),
('KEVIN', 'URICHI', 21, 60614568, 'kevinuruchil23@gmail.com');

select est.*
from estudiantes as est;

#sirve para ver cual es el ultimo ID insertada
select last_insert_id();

# Agregar un nuevo campo a la tabla estudiantes
ALTER TABLE estudiantes
ADD COLUMN direccion VARCHAR(200);

```

```

ALTER TABLE estudiantes
ADD COLUMN fax VARCHAR(10),
ADD COLUMN sexo VARCHAR(10);

#sirve para eliminar una columna
ALTER TABLE estudiantes
DROP COLUMN fax;
#sirve para agregar un valor por defecto e
ALTER TABLE estudiantes
ADD COLUMN fax VARCHAR(10) DEFAULT 'FAX-123';

#Ejercicio
#Mostrar los registros de aquellos estudiantes que su nombre sea igual
a nombre4.
SELECT *
FROM estudiantes
WHERE estudiantes.nombres = 'EVER';

SELECT est.nombres, est.apellidos, est.edad
FROM estudiantes as est
WHERE est.edad >18

#Mostrar los registros donde cuyo ID sea PAR. (o IMPAR).
SELECT est.*
FROM estudiantes as est
WHERE est.id_est %2=0

SELECT est.*
FROM estudiantes as est
WHERE est.id_est %2=1

#Generar las tablas de acuerdo a la siguiente imagen.
#A) Crear una base de datos denominada Librería.
#B) Hacer uso de esa base de datos.

CREATE DATABASE LIBRERIA;
USE LIBRERIA;

CREATE TABLE categories
(
    category INTEGER PRIMARY KEY NOT NULL,
    name VARCHAR(50) NOT NULL
);
CREATE TABLE publishers
(
    published_id INTEGER PRIMARY KEY NOT NULL,
    name VARCHAR(50) NOT NULL
);
CREATE TABLE books
(
    book_id INTEGER AUTO_INCREMENT PRIMARY KEY NOT NULL,
    title VARCHAR(50) NOT NULL,
    isbn VARCHAR(40) NOT NULL,
    published_date DATE NOT NULL,
    description VARCHAR(50) NOT NULL,
    category INTEGER NOT NULL,

```

```
published_id INTEGER NOT NULL,  
FOREIGN KEY (category) REFERENCES categories(category),  
FOREIGN KEY (published_id) REFERENCES publishers(published_id)  
);
```