

TALLER 1 DE BASE DE DATOS

1.

Base datos: bd_universidad

DICCIONARIO DE DATOS

TABLA : curso

NOMBRE	TIPO	LONGITUD	NULL	PK	FK	DESCRIPCION	DOMINIO
idCurso	MEDIUMINT			x		Llave primaria de la tabla curso	[0-9]
nombre	VARCHAR	50				Identificar el nombre de una categoria	[a-z][A-Z]
fechaInicio	DATE					Fecha inicio del curso	"aaaa-mm-dd"
fechaFin	DATE		x			Fecha fin del curso	"aaaa-mm-dd"
cantidadEstudiantes	MEDIUMINT					Indica la cantidad de estudiantes en el curso	[0-9]

TABLA : estudiante

NOMBRE	TIPO	LONGITUD	NULL	PK	FK	DESCRIPCION	DOMINIO
idEstudiante	MEDIUMINT			x		Llave foranea de la tabla examen	[0-9]
nombre	VARCHAR	80				Nombres y apellidos del estudiante	[a-z][A-Z]
email	VARCHAR	100				Correo electronico de estudiante	[a-z][A-Z][0-9][/_ _ # @ * , &]
telefono	VARCHAR	20	x			Telefono del estudiante	[0-9][+ -]
fkIdCurso	MEDIUMINT				x	Llave foranea de la tabla estudiante	[0-9]

TABLA : examen

NOMBRE	TIPO	LONGITUD	NULL	PK	FK	DESCRIPCION	DOMINIO
idExamen	MEDIUMINT			x		Llave primaria de la tabla examen	[0-9]
nota	TINYINT					Nota asignada al examen	[0-9].
fechaRealizacion	DATETIME					Fecha y hora de realización del examen	"aaaa-mm-dd H:is"
fkIdCurso	MEDIUMINT				x	Llave foranea de la tabla examen	[0-9]
fkIdEstudiante	MEDIUMINT				x	Llave foranea de la tabla examen	[0-9]

TABLA : pregunta

NOMBRE	TIPO	LONGITUD	NULL	PK	FK	DESCRIPCION	DOMINIO
idPregunta	MEDIUMINT			x		Llave primaria de la tabla pregunta	[0-9]
pregunta	VARCHAR	255				contenido de la pregunta	[a-z][A-Z][? - _ "][0-9]
respuesta	TEXT					Contenido de la respuesta	[a-z][A-Z][0-9][/_ _ # @ * , &] % !
tema	VARCHAR	100				identifica el tema de la pregunta	[a-z][A-Z][0-9][/_ _ # @ * , &] % !
idExamen	MEDIUMINT				x	Llave foranea de la tabla pregunta	[0-9]

2. CREATE DATABASE universidad;

```
MariaDB [(none)]> CREATE DATABASE universidad;
Query OK, 1 row affected (0.012 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| db_articulos |
| eventos |
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
| senacaldas |
| test |
| universidad |
+-----+
9 rows in set (0.015 sec)
```

3. CREATE TABLE curso(
idCurso MEDIUMINT PRIMARY KEY AUTO_INCREMENT,
nombre VARCHAR(50) NOT NULL,
fechaInicio DATE NOT NULL,
fechaFin DATE,
cantidadEstudiantes MEDIUMINT NOT NULL
);

```
MariaDB [universidad]> CREATE TABLE curso(  
-> idCurso MEDIUMINT PRIMARY KEY AUTO_INCREMENT,  
-> nombre VARCHAR(50) NOT NULL,  
-> fechaInicio DATE NOT NULL,  
-> fechaFin DATE,  
-> cantidadEstudiantes MEDIUMINT NOT NULL  
-> );  
Query OK, 0 rows affected (0.032 sec)
```

CREATE TABLE estudiante(
idEstudiante MEDIUMINT PRIMARY KEY AUTO_INCREMENT,
nombre VARCHAR(80) NOT NULL,
email VARCHAR(100) NOT NULL,
telefono VARCHAR(20),
fkIdCurso MEDIUMINT NOT NULL,
FOREIGN KEY (fkIdCurso) REFERENCES curso(idCurso)
);

```
MariaDB [universidad]> CREATE TABLE estudiante(  
-> idEstudiante MEDIUMINT PRIMARY KEY AUTO_INCREMENT,  
-> nombre VARCHAR(80) NOT NULL,  
-> email VARCHAR(100) NOT NULL,  
-> telefono VARCHAR(20),  
-> fkIdCurso MEDIUMINT NOT NULL,  
-> FOREIGN KEY (fkIdCurso) REFERENCES curso(idCurso)  
-> );  
Query OK, 0 rows affected (0.014 sec)
```

CREATE TABLE examen(
idExamen MEDIUMINT PRIMARY KEY AUTO_INCREMENT,
nota TINYINT NOT NULL,
fechaRealizacion DATETIME NOT NULL,
fkIdCurso MEDIUMINT NOT NULL,
fkIdEstudiante MEDIUMINT NOT NULL,
FOREIGN KEY (fkIdEstudiante) REFERENCES estudiante(idEstudiante),
FOREIGN KEY (fkIdCurso) REFERENCES curso(idCurso)

);

```
MariaDB [universidad]> CREATE TABLE examen(  
  -> idExamen MEDIUMINT PRIMARY KEY AUTO_INCREMENT,  
  -> nota TINYINT NOT NULL,  
  -> fechaRealizacion DATETIME NOT NULL,  
  -> fkIdCurso MEDIUMINT NOT NULL,  
  -> fkIdEstudiante MEDIUMINT NOT NULL,  
  -> FOREIGN KEY (fkIdEstudiante) REFERENCES estudiante(idEstudiante),  
  -> FOREIGN KEY (fkIdCurso) REFERENCES curso(idCurso)  
  -> );  
Query OK, 0 rows affected (0.015 sec)
```

```
CREATE TABLE pregunta(  
  idPregunta MEDIUMINT PRIMARY KEY AUTO_INCREMENT,  
  pregunta VARCHAR(255) NOT NULL,  
  respuesta TEXT NOT NULL,  
  tema VARCHAR(100) NOT NULL,  
  fkIdExamen MEDIUMINT NOT NULL,  
  FOREIGN KEY (fkIdExamen) REFERENCES examen(idExamen)  
);
```

```
MariaDB [universidad]> CREATE TABLE pregunta(  
  -> idPregunta MEDIUMINT PRIMARY KEY AUTO_INCREMENT,  
  -> pregunta VARCHAR(255) NOT NULL,  
  -> respuesta TEXT NOT NULL,  
  -> tema VARCHAR(100) NOT NULL,  
  -> fkIdExamen MEDIUMINT NOT NULL,  
  -> FOREIGN KEY (fkIdExamen) REFERENCES examen(idExamen)  
  -> );  
Query OK, 0 rows affected (0.026 sec)
```

```
MariaDB [universidad]> show tables;  
+-----+  
| Tables_in_universidad |  
+-----+  
| curso                  |  
| estudiante             |  
| examen                 |  
| pregunta               |  
+-----+  
4 rows in set (0.000 sec)
```

4. # Insertar Datos en la tabla curso

```
INSERT INTO curso (nombre,fechaInicio,fechaFin,cantidadEstudiantes)  
VALUES ('Curso 1','2024-02-12','2025-07-24','10');
```

```
INSERT INTO curso (nombre,fechaInicio,fechaFin,cantidadEstudiantes)
VALUES ('Curso 2','2024-03-05','2025-09-30','10');
```

```
INSERT INTO curso (nombre,fechaInicio,fechaFin,cantidadEstudiantes)
VALUES ('Curso 3','2024-04-15','2025-11-20','10');
```

```
INSERT INTO curso (nombre,fechaInicio,fechaFin,cantidadEstudiantes)
VALUES ('Curso 4','2024-05-30','2026-01-15','10');
```

```
MariaDB [universidad]> SELECT * FROM curso;
```

idCurso	nombre	fechaInicio	fechaFin	cantidadEstudiantes
1	Curso 1	2024-02-12	2025-07-24	10
2	Curso 2	2024-03-05	2025-09-30	10
3	Curso 3	2024-04-15	2025-11-20	10
5	Curso 4	2024-05-30	2026-01-15	10

```
4 rows in set (0.000 sec)
```

Insertar Datos en la tabla estudiante

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
VALUES ('Daniel','daniel@gmail.com','31123212',1);
```

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
VALUES ('Isabela','isa@gmail.com','312315434',2);
```

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
VALUES ('Viviana','vivi@gmail.com','356736539',3);
```

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
VALUES ('Sofia','sofi@gmail.com','312374323',5);
```

```
MariaDB [universidad]> SELECT * FROM estudiante;
```

idEstudiante	nombre	email	telefono	fkIdCurso
1	Daniel	daniel@gmail.com	31123212	1
2	Isabela	isa@gmail.com	312315434	2
3	Viviana	vivi@gmail.com	356736539	3
4	Sofia	sofi@gmail.com	312374323	5

4 rows in set (0.000 sec)

Insertar Datos en la tabla examen

```
INSERT INTO examen (fechaRealizacion,fkIdCurso,fkIdEstudiante)
VALUES ('2024-03-15 12:00:00',1,1);
```

```
INSERT INTO examen (fechaRealizacion,fkIdCurso,fkIdEstudiante)
VALUES ('2024-04-10 10:00:00',2,2);
```

```
INSERT INTO examen (fechaRealizacion,fkIdCurso,fkIdEstudiante)
VALUES ('2024-06-20 4:00:00',5,3);
```

```
MariaDB [universidad]> SELECT * FROM examen
-> ;
```

idExamen	nota	fechaRealizacion	fkIdCurso	fkIdEstudiante
1	NULL	2024-03-15 12:00:00	1	1
2	NULL	2024-04-10 10:00:00	2	2
4	NULL	2024-06-20 04:00:00	5	3

3 rows in set (0.000 sec)

Insertar Datos en la tabla pregunta

```
INSERT INTO pregunta (pregunta,tema,fkIdExamen)
VALUES ('Cuanto es 1+1','Matematicas',1);
```

```
INSERT INTO pregunta (pregunta,tema,fkIdExamen)
VALUES ('Cual es la segunda ley de Newton','Fisica',1);
```

```
INSERT INTO pregunta (pregunta,tema,fkIdExamen)
VALUES ('Cual es la capital de Colombia','Sociales',2);
```

```
INSERT INTO pregunta (pregunta,tema,fkIdExamen)
VALUES ('Para que se utiliza la tilde','Español',2);
```

```
INSERT INTO pregunta (pregunta,tema,fkIdExamen)
VALUES ('Para que funciona if en dart','Programación',4);
```

```
INSERT INTO pregunta (pregunta,tema,fkIdExamen)
VALUES ('¿Qué es RAM?','MODELADO de software',4);
```

MariaDB [universidad]> SELECT * FROM pregunta;






idPregunta	pregunta	respuesta	tema	fkIdExamen
1	Cuanto es 1+1	NULL	Matematicas	1
2	Cual es la segunda ley de Newton	NULL	Fisica	1
3	Cual es la capital de Colombia	NULL	Sociales	2
4	Para que se utiliza la tilde	NULL	Español	2
5	Para que funciona if en dart	NULL	Programación	4
6	¿Qué es RAM?	NULL	MODELADO de software	4

6 rows in set (0.000 sec)

5. Ejecutar las siguientes consultas







- a. Seleccionar los exámenes con nota superior a 3

```
SELECT *
FROM examen
WHERE nota >=3;
```

	idExamen	nota	fechaRealizacion	fkIdCurso	fkIdEstudiante
<input type="checkbox"/>   	1	3	2024-03-15 12:00:00	1	1
<input type="checkbox"/>   	2	5	2024-04-10 10:00:00	2	2







- b. Seleccionar los exámenes con fecha de realización mayor a 5 de Julio de 2024

```
SELECT *
FROM examen
WHERE fechaRealizacion > "2024-04-01";
```

	idExamen	nota	fechaRealizacion	fkIdCurso	fkIdEstudiante
<input type="checkbox"/>   	2	5	2024-04-10 10:00:00	2	2
<input type="checkbox"/>   	4	2	2024-06-20 04:00:00	5	3

- c. Seleccionar solo los nombres de cursos, con fecha de finalización menor a 1 de Octubre de 2024

```
SELECT nombre, fechaFin
FROM curso
WHERE fechaFin <= "2025-10-1";
```

<div><div><div><div></div><div></div><div></div></div><div></div></div></div>					nombre	fechaFin		
<input type="checkbox"/>		Editar		Copiar		Borrar	Curso 1	2025-07-24
<input type="checkbox"/>		Editar		Copiar		Borrar	Curso 2	2025-09-30

- d. Seleccionar los estudiantes cuya nota sea mayor a 3 (Utilizando INNER JOIN)
 Seleccionar los estudiantes cuya nota sea mayor a 3 (Utilizando INNER JOIN)

```
SELECT nombre, email, nota, fechaRealizacion
FROM estudiante INNER JOIN examen
ON estudiante.idEstudiante = examen.fkIdEstudiante;
```

nombre	email	nota	fechaRealizacion
Daniel	daniil@gmail.com	3	2024-03-15 12:00:00
Isabela	isa@gmail.com	5	2024-04-10 10:00:00
Viviana	viviana@gmail.com	2	2024-06-20 04:00:00

- e. Seleccionar las preguntas que tiene un examen específico

```
SELECT *
FROM pregunta
WHERE fkIdExamen = 2;
```

				idPregunta	pregunta	respuesta	tema	fkIdExamen
<input type="checkbox"/>		Editar		Copiar		Borrar		
				3	Cual es la capital de Colombia	NULL	Sociales	2
<input type="checkbox"/>		Editar		Copiar		Borrar		
				4	Para que se utiliza la tilde	NULL	Español	2

- f. Seleccionar la cantidad de preguntas que tiene cada tema. Para este punto se debe mostrar una lista con solamente: el tema, la cantidad de preguntas y la fecha de realización del examen (Utilizando GROUP BY)

```
SELECT tema, COUNT(*) AS cantidadTema
FROM pregunta
GROUP BY tema;
```

tema	cantidadTema
Fisica	1
MODELADO de software	1
Sociales	2

6. #Modificar el nombre y la fecha de inicio de dos cursos

```
UPDATE curso
```

```
SET nombre = 'Curso 5'
```

```
WHERE nombre= 'Curso 4';
```

```
UPDATE curso
```

```
SET nombre = 'Curso 3-1'
```

```
WHERE nombre = 'Curso 3';
```

#Modificar la fecha de inicio de dos cursos

```
UPDATE curso
```

```
SET fechaInicio = '2024-06-10'
```

```
WHERE fechaInicio = '2024-05-30';
```

```
UPDATE curso
```

```
SET fechaInicio = '2024-04-30'
```

```
WHERE fechaInicio = '2024-04-15';
```

```
MariaDB [universidad]> SELECT * FROM curso;
```

idCurso	nombre	fechaInicio	fechaFin	cantidadEstudiantes
1	Curso 1	2024-02-12	2025-07-24	10
2	Curso 2	2024-03-05	2025-09-30	10
3	Curso 3-1	2024-04-30	2025-11-20	10
5	Curso 5	2024-06-10	2026-01-15	10

```
4 rows in set (0.000 sec)
```

#Modificar email de dos estudiantes

```
UPDATE estudiante
```

```
SET email = 'daniiii@gmail.com'
```

```
WHERE email = 'daniel@gmail.com';
```

```
UPDATE estudiante
```



```
SET email = 'viviana@gmail.com'
WHERE email = 'vivi@gmail.com';
```

#Modificar telefono de dos estudiantes

```
UPDATE estudiante
SET telefono = '313456545'
WHERE telefono = '3134565456';
```

```
UPDATE estudiante
SET telefono = '3124654157'
WHERE telefono = '331274323';
```

```
MariaDB [universidad]> SELECT * FROM estudiante;
```

idEstudiante	nombre	email	telefono	fkIdCurso
1	Daniel	daniiii@gmail.com	313456545	1
2	Isabela	isa@gmail.com	312315434	2
3	Viviana	viviana@gmail.com	356736539	3
4	Sofia	sofi@gmail.com	312374323	5

```
4 rows in set (0.000 sec)
```

7. #Eliminar dos exámenes

```
DELETE FROM examen
WHERE idExamen = 1;
```

```
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`universidad`.`pregunta`, CONSTRAINT `pregunta_ibfk_1` FOREIGN KEY (`fkIdExamen`) REFERENCES `examen` (`idExamen`))
MariaDB [universidad]> Bye
Ctrl-C -- exit!
```

No es posible eliminar los exámenes ya que tienen llaves foráneas asociadas, y su llave primaria esta como llave foránea en la tabla pregunta

#Eliminar dos examenes

```
DELETE FROM examen
WHERE idExamen = 1;
```

#Elimintar dos preguntas

```
DELETE FROM pregunta
WHERE idPregunta = 1;
```

```
DELETE FROM pregunta
WHERE idPregunta = 5;
```

```
MariaDB [universidad]> SELECT * FROM pregunta;
```




idPregunta	pregunta	respuesta	tema	fkIdExamen
2	Cual es la segunda ley de Newton	NULL	Fisica	1
3	Cual es la capital de Colombia	NULL	Sociales	2
4	Para que se utiliza la tilde	NULL	Español	2
6	¿Qué es RAM?	NULL	MODELADO de software	4

```
4 rows in set (0.000 sec)
```

Puntos adicionales

1. Ordenar las preguntas por el campo “respuesta” de forma descendente y ascendente (utilizando ORDER BY)

```
SELECT respuesta
FROM pregunta
ORDER BY respuesta ASC;
```

← T →					respuesta ▲ 1
<input type="checkbox"/>		Editar		Copiar	 Borrar bogotá
<input type="checkbox"/>		Editar		Copiar	 Borrar ley de la fuerza
<input type="checkbox"/>		Editar		Copiar	 Borrar RAM es la memoria que usa el ordenador para trabaj...
<input type="checkbox"/>		Editar		Copiar	 Borrar tilde se usa para marcar la sílaba tónica en palab...

```
SELECT respuesta
FROM pregunta
ORDER BY respuesta DESC;
```

	respuesta	▼	1
<input type="checkbox"/>	Editar	Copiar	Borrar tilde se usa para marcar la sílaba tónica en palab...
<input type="checkbox"/>	Editar	Copiar	Borrar RAM es la memoria que usa el ordenador para trabaj...
<input type="checkbox"/>	Editar	Copiar	Borrar ley de la fuerza
<input type="checkbox"/>	Editar	Copiar	Borrar bogotá

- Ingresar estudiantes para que haya un total de 7 registros en la tabla estudiante.
Traer solamente los 4 estudiantes con mejor nota (utilizando LIMIT y ORDER BY)

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
```

```
VALUES ('Manuel','gonzales@gmail.com','3342452344',1);
```

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
```

```
VALUES ('jose','joseee@gmail.com','3546544677',1);
```

```
INSERT INTO estudiante (nombre,email,telefono,fkIdCurso)
```

```
VALUES ('Sara','ssssara@gmail.com','3226067640',1);
```

<div><div><div></div><div></div><div></div></div></div>				idEstudiante	nombre	email	telefono	fkIdCurso
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1	Daniel	daniii@gmail.com	313456545	1
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	2	Isabela	isa@gmail.com	312315434	2
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	3	Viviana	viviana@gmail.com	356736539	3
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	4	Sofia	sofi@gmail.com	312374323	5
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	5	Manuel	gonzales@gmail.com	3342452344	1
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	6	jose	joseee@gmail.com	3546544677	1
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	7	Sara	ssssara@gmail.com	3226067640	1

```
SELECT *
```

```
FROM examen
```

```
ORDER BY nota DESC
```

```
LIMIT 4;
```

			idExamen	nota	1	fechaRealizacion	fkIdCurso	fkIdEstudiante
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	2	5	2024-04-10 10:00:00	2	2
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	7	4	2024-03-15 12:00:00	1	7
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	5	4	2024-03-15 12:00:00	1	5
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	6	4	2024-03-15 12:00:00	1	6

3. Consultar los estudiantes (solamente los campos: nombre, email y teléfono) que pertenezcan a cursos donde la cantidad de estudiantes sea mayor a 2. Traer también el nombre del curso

```

SELECT e.nombre, e.email, e.telefono, c.nombre
FROM estudiante e
JOIN curso c ON e.fkIdCurso = c.idCurso
WHERE e.fkIdCurso IN (
    SELECT fkIdCurso
    FROM estudiante
    GROUP BY fkIdCurso
    HAVING COUNT(*) > 2
);

```

nombre	email	telefono	nombre
Daniel	daniiii@gmail.com	313456545	Curso 1
Manuel	gonzales@gmail.com	3342452344	Curso 1
jose	joseeee@gmail.com	3546544677	Curso 1
Sara	sssssara@gmail.com	3226067640	Curso 1

4. Consultar estudiantes que (utilizando LIKE): - Nombre empiece por la letra "S" - Nombre termine con la letra "I" - Nombre tenga una letra "n" en la mitad

```

SELECT nombre
FROM estudiante
WHERE nombre LIKE 'S_%';

```

				nombre
<input type="checkbox"/>	Editar	Copiar	Borrar	Sofia
<input type="checkbox"/>	Editar	Copiar	Borrar	Sara

```
SELECT nombre
FROM estudiante
WHERE nombre LIKE '%_l';
```

				nombre
<input type="checkbox"/>	Editar	Copiar	Borrar	Daniel
<input type="checkbox"/>	Editar	Copiar	Borrar	Manuel

```
SELECT nombre
FROM estudiante
WHERE nombre LIKE '%_n_%';
```

				nombre
<input type="checkbox"/>	Editar	Copiar	Borrar	Daniel
<input type="checkbox"/>	Editar	Copiar	Borrar	Viviana
<input type="checkbox"/>	Editar	Copiar	Borrar	Manuel

5. Aplicar cada una de las siguientes funciones de MySQL al taller realizado

```
AVG()

SELECT AVG(nota) AS promedio_notas
FROM examen;
```

promedio_notas
3.6667

SUM()

```
SELECT SUM(nota) AS suma_notas
```

```
FROM examen;
```

suma_notas

22

MAX()

```
SELECT MAX(nota) AS nota_maxima
```

```
FROM examen;
```

nota_maxima

5

MIN()

```
SELECT MIN(nota) AS nota_minima
```

```
FROM examen;
```

nota_minima

2
