

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

6: Control Educativo

DDL

CREATE TABLE Profesor (

Id_profesor INT PRIMARY KEY AUTO_INCREMENT,
nombre_profesor VARCHAR (100) NOT NULL,
apellido_profesor VARCHAR (100) NOT NULL,
fecha_contratacion DATE,
email VARCHAR (150) NOT NULL UNIQUE,
telefono VARCHAR (20)
);

CREATE TABLE Asignatura (

Id_asignatura INT PRIMARY KEY AUTO_INCREMENT,
nombre_asignatura VARCHAR (100) NOT NULL UNIQUE,
credito INT NOT NULL,
descripcion_asignatura TEXT
);

CREATE TABLE Curso (

Id_curso INT PRIMARY KEY AUTO_INCREMENT,
nombre_curso VARCHAR (100) NOT NULL UNIQUE,
nivel_educativo VARCHAR (50),
duracion_meses INT,
Id_profesor_tutor INT,
FOREIGN KEY (Id_profesor_tutor) REFERENCES Profesor
(Id_profesor)
);

CREATE TABLE Estudiante (

Id_estudiante INT PRIMARY KEY AUTO_INCREMENT,
nombre_estudiante VARCHAR (100) NOT NULL,
apellido_estudiante VARCHAR (100) NOT NULL,
fecha_nacimiento DATE,
genero CHAR (1),
email VARCHAR (150) UNIQUE,
telefono_contacto VARCHAR (20),
Id_curso_actual INT,
FOREIGN KEY (Id_curso_actual) REFERENCES Curso (Id_curso)
);

CREATE TABLE Clase (

Id_clase INT PRIMARY KEY AUTO_INCREMENT,
Id_asignatura INT NOT NULL,
Id_profesor INT NOT NULL,
codigo_clase VARCHAR (50) NOT NULL UNIQUE,


```

año-académico INT NOT NULL,
Semestre VARCHAR (20),
Cupo-máximo INT NOT NULL,
aula-asignada VARCHAR (50)
FOREIGN KEY (id-asignatura) REFERENCES Asignatura
(id-asignatura),
FOREIGN KEY (id-profesor) REFERENCES (id-profesor)
);

```

```

CREATE TABLE Matricula (
id-matricula INT PRIMARY KEY AUTO-INCREMENT,
id-estudiante INT NOT NULL,
id-clase INT NOT NULL,
fecha-matricula DATETIME NOT NULL DEFAULT
CURRENT-TIMESTAMP,
estado-matricula VARCHAR (50) NOT NULL DEFAULT
'Activa',
Calificación-Final DECIMAL (4,2),
FOREIGN KEY (id-estudiante) REFERENCES Estudiante
(id-estudiante),
FOREIGN KEY (id-clase) REFERENCES Clase (id-clase),
UNIQUE (id-estudiante, id-clase)
);

```

```

CREATE TABLE Calificación (
id-calificación INT PRIMARY KEY AUTO-INCREMENT,
id-matricula INT NOT NULL,
Tipo-evaluación VARCHAR (50) NOT NULL,
fecha-evaluación DATE NOT NULL,
Nota-obtenida DECIMAL (4,2) NOT NULL,
peso-en-promedio DECIMAL (3,2) NOT NULL,
FOREIGN KEY (id-matricula) REFERENCES Matricula
(id-matricula)
);

```

```

INSERT INTO Profesor (nombre-profesor, apellido-
profesor, fecha-contratación, email, teléfono) VALUES
('Carlos', 'Ramirez', '2020-08-15', 'Carlos.ramirez@educativo.
com', '3101112233'),
('Laura', 'Martinez', '2021-02-01', 'Laura.martinez@educativo.
com', '3204445566');

```

```

INSERT INTO Asignatura (nombre-asignatura,
creditos, descripción-asignatura) VALUES
('Matemáticas VII', 4, 'Álgebra y Geometría avanzada
para Séptimo grado.').

```

```

INSERT INTO Curso (nombre-curso, nivel-educativo,
duración-meses, id-profesor-tutor) VALUES
('Séptimo Grado A', 'Secundaria', 10, 1),

```

```

INSERT INTO Estudiante (nombre-estudiante,
apellido-estudiante, fecha-nacimiento, género,
email, teléfono-contacto, id-curso-actual) VALUES
('Sofía', 'López', '2010-03-20', 'F', 'Sofia.Lopez@estudiante

```


3157778899', 1),

INSERT INTO Clase (id - asignatura, id - profesor, código - clase, año - académico, semestre, cupo - máximo, aula - asignada) VALUES
(1, 1, NOW(), 'Activa'),

INSERT INTO Matricula (id - estudiante, id - clase, fecha - matrícula, estado - matrícula) VALUES
(1, 1, NOW(), 'Activa'),

INSERT INTO Calificación (id - matrícula, tipo - evaluación, fecha - evaluación, nota obtenida, peso - en promedio) VALUES
(1, 'Quiz', '2025-07-20', 4.0, 0.2),

CONSULTAS

1. SELECT

SELECT
nombre - profesor,
email.

FROM
Profesor;

2. JOIN

SELECT
E.nombre - estudiante,
E.apellido - estudiante,
C.nombre - curso

FROM
Estudiante E

JOIN
curso C ON E.id - curso - actual = C.id - curso;

3. GROUP BY y COUNT.

SELECT
C.nombre - curso,
COUNT(E.id - estudiante) AS total - estudiantes

FROM
Estudiante E

JOIN
curso C ON E.id - curso - actual = C.id - curso

GROUP BY
C.nombre - curso

ORDER BY
total - estudiantes DESC;

4. HAVING Y COUNT

SELECT

A.nombre_asignatura
COUNT (DISTINCT C.id_profesor) AS total_
profesores_impartiendo

FROM

Asignatura A

JOIN

Clase C ON A.id_asignatura = C.id_asignatura

GROUP BY

A.nombre_asignatura

HAVING

COUNT (DISTINCT C.id_profesor) > 1;

5. LIMIT Y ORDER BY

SELECT

M.id_estudiante,
A.nombre_asignatura,
CL.codigo_clase,
C.nota_obtenida,
C.fecha_evaluación,

FROM

Calificación C

JOIN

Matricula M ON C.id_matricula = M.id_matricula

JOIN

Clase CL ON M.id_clase = CL.id_clase

JOIN

Asignatura A ON CL.id_asignatura = A.id_asignatura

ORDER BY

C.fecha_evaluación DESC

LIMIT 2

6. AVG

SELECT

AVG (ca.nota_obtenida) AS promedio_
quizzes_matematicas

FROM

Calificación Ca

JOIN
Matricula M ON Ca.id - matricula = M.id -
matricula

JOIN
Clase CL ON M.id - clase = CL.id - clase

JOIN
Asignatura A ON A.id - asignatura = A.id -
asignatura

WHERE
A.nombre - asignatura = 'Matemáticas VII' AND
Ca.tipo - evaluación = 'Quiz';

7. DISTINCT

SELECT DISTINCT
tipo - evaluación

FROM

Calificación;

8. Subconsulta

SELECT
E.nombre - estudiante,
E.apellido - estudiante

FROM
Estudiante E

WHERE

E.id - estudiante IN (
SELECT M.id - estudiante
FROM Matricula M
JOIN Clase CL ON M.id - clase = CL.id - clase
WHERE CL.codigo - clase = 'MATFA-2025' AND
M.estado - matricula = 'Activo'

);

- Sentencia

1. UPDATE

Objetivo: Actualizar el estado de la matricula
de un estudiante en una clase a 'Completada'.

SQL


```

UPDATE Matricula
SET estado-matricula = 'completada', calificación
    - final = 4.2
WHERE id-estudiante = 1 AND id-clase = 1;

```

Procedimiento

```

CREATE PROCEDURE MatricularEstudiante (
    IN p-id-estudiante INT,
    IN p-id-clase INT,
    OUT p-mensaje VARCHAR (255)
)

```

BEGIN

```

DECLARE V-cupo-actual INT;
DECLARE V-matriculas-actuales INT;
DECLARE V-estudiante-existe INT;
DECLARE V-clase-existe INT;
DECLARE V-ya-matriculado INT;

```

```

SELECT COUNT (*) INTO V-estudiante-existe
FROM Estudiante WHERE id-estudiante = p-id-estudiante;
SELECT COUNT (*) INTO V-clase-existe FROM Clase
WHERE id-clase = p-id-clase;

```

```

IF V-estudiante-existe = 0 THEN
    SET p-mensaje = 'Error: EL ID de estudiante
    no existe.';
ELSE IF V-clase-existe = 0 THEN
    SET p-mensaje = 'Error: EL ID de clase no
    existe.';

```

ELSE

```

SELECT COUNT (*) INTO V-ya-matriculado FROM
matricula WHERE id-estudiante = p-id-
estudiante AND id-clase = p-id-clase;
IF V-ya-matriculado > 0 THEN
    SET p-mensaje = 'Error: El estudiante
    ya está matriculado en esta clase.';

```

ELSE

```

SELECT cupo-maximo INTO V-cupo-actual
FROM Clase WHERE id-clase = p-id-clase;
SELECT COUNT (*) INTO V-matriculas-actuales
FROM Matricula WHERE id-clase = p-id-clase
AND estado-matricula = 'Activa';

```

```

IF V-matriculas-actuales < V-cupo-actual
FROM Clase WHERE id-clase = p-id-clase;
SELECT COUNT (*) INTO V-matriculas-
actuales FROM Matricula WHERE id-clase = p-
id-clase AND estado-matricula = 'Activa';

```



```
IF V-matriculas-actuales < V-cupo-actual THEN
  INSERT INTO Matricula (id-estudiante,
    id-clase, fecha-matricula,
    VALUES (p-id-estudiante, p-id-clase, now()),
    'Activa');
  SET p-mensaje = 'Estudiante matriculado con
    éxito.';
ELSE
  SET p-mensaje = 'Error: No hay cupo disponible en
    esta clase.';
END IF;
END IF;
END IF;
END //
```


Nombre de Tabla: Profesor

Fecha de creación 10/07/2023

Descripción: Entidad que almacenaran la información del profesor

Clave	Campo	Tipo de dato	Tamaño	Descripción:
PK	Id - profesor	INT		Identificador Único del profesor
	nombre - profesor	VARCHAR	100	Nombre del profesor
	Apellido - profesor	VARCHAR	100	Apellido del profesor
	Fecha - Contratación	DATE		Fecha de contratación
	email -	VARCHAR	150	Email del profesor
	Telefono	VARCHAR	20	Telefono del profesor

Nombre de tabla Asignatura

Fecha creacion 10/01/2025

Descripción. Entidad que almacenará la información de la asignatura

Clave	Campo	Tipo de dato	Tamaño	Descripción.
PK	ID-Asignatura	INT		Identificador unico de asignatura
	nombre-Asignatura	VARCHAR	100	nombre de la asignatura
	creditos	INT		creditos
	descripcion-asignatura	TEXT		descripcion de la asignatura.

Nombre de tabla CURSO

Fecha creacion 10/07/2025

Descripcion Entidad que almacenará a curso

CLAVE campo

Tipo de dato

Tamaño

Descripcion

PK

Id - Curso

INT

nombre -
Curso

VARCHAR

100

nombre del curso

nivel -
educativo

VARCHAR

50

nivel educativo

duracion -
meses

INT

duracion del curso

Id - Profesor -
tutor

INT

Nombre de tabla: Estudiante

Fecha de creación 10/07/2023

Descripción: Entidad que almacenará la información del estudiante

Clave	Campo	Tipo de dato	Tamaño	Descripción
PK	id - estudiante	INT		Identificador único del estudiante
	nombre - estudiante	VARCHAR	100	nombre del estudiante
	Apellido - estudiante	VARCHAR	100	Apellido del estudiante
	Fecha - nacimiento	DATE		fecha de nacimiento del estudiante.
	Genero	CHAR		el estudiante
	email	VARCHAR	150	email del estudiante
	Telefono - contacto	VARCHAR	20	telefono del estudiante
	id - Curso - actual	INT		Curso actual del estudiante.

Nombre de tabla: Clase

Fecha de creación: 10/07/2023

Descripción Entidad que almacenará la clase

Clave	Campo	Tipo de Dato	Tamaño	Descripción
PK	Id - Clase	INT		Identificador Único de Clase
FK	Id - asignatura	INT		
FK	Id - profesor	INT		
	Código - Clase	VARCHAR	50	Código de clase
	años - académicos	INT	10	años académicos
	Semestre	VARCHAR	50	Semestre de la Clase
	Cupo - máximo	INT		Cupo máximo
	aula - asignada	VARCHAR	50	aula asignada.

Nombre de tabla matricula

Fecha creación: 10/01/2025

Descripción entidad que almacenara la matricula

Clave campo tipo de dato Tamaño

Descripción

PK id - matricula INT

Identificador unico de matricula

FK id - estudiante INT

FK id - clase INT

Fecha -
matricula DATE TIME

Fecha de matricula

estado -
matricula VARCHAR 50

Estado matricula

calificación -
final DECIMAL 4.2

calificación final

Nombre de Tabla: Calificación

Fecha creación: 10/07/2025

Descripción: Entidad almacenará la calificación

Clave	Campo	Tipo de Dato	Tamaño	Descripción
PK	Id - Calificación	INT		Identificador único de calificación
FK	Id - matrícula	INT		
	Tipo - Evaluación	VARCHAR	50	Tipo de evaluación
	Fecha - evaluación	DATE		Fecha de evaluación
	Nota - Obtenido	DECIMAL	4,2	Nota obtenida
	Peso - procedimiento	DECIMAL	3,2	Peso en promedio

Profesor
id-profesor (PK)
nombre-profesor
apellido-profesor
fecha-Contratación
email
telefono

Estudiante
id-estudiante (PK)
nombre-estudiante
apellido-estudiante
fecha-nacimiento
genero
email
telefono-contacto
id-Curso-actual (FK)

Clase
id-clase (PK)
id-assignatura (FK)
id-profesor (FK)
Codigo-clase
año-academico
semestre
cupo-maximo
aula-asignada

Asignatura
id-assignatura (PK)
nombre-assignatura
credito
descripcion-assignatura

1:1

N:M

1:N

Curso
id-curso (PK)
nombre-curso
nivel-educativo
duracion-meses
id-profesor (FK)

1:N

Calificación
id-calificacion (PK)
id-matricula (FK)
tipo-evaluacion
fecha-evaluacion
nota-obtenida
peso-en-promedio

N:M

Matricula
id-matricula (PK)
id-estudiante (FK)
id-clase (FK)
fecha-matricula
estado-matricula
calificacion-final