SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `tpfinal` DEFAULT CHARACTER SET utf8 ;

-- -----------------------------------------------------

-- Schema tpfinal

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema tpfinal

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `tpfinal` DEFAULT CHARACTER SET utf8mb4 ;

USE `tpfinal` ;

CREATE DATABASE IF NOT EXISTS tpfinal;

USE tpfinal;

-- Usuarios

CREATE TABLE IF NOT EXISTS Usuarios (

idUsuario INT NOT NULL AUTO\_INCREMENT,

Usuario VARCHAR(45) NOT NULL,

Contraseña VARCHAR(45) NOT NULL,

Rol VARCHAR(45) NOT NULL,

Email VARCHAR(45) NOT NULL,

PRIMARY KEY (idUsuario)

) ENGINE = InnoDB;

-- Alumno

CREATE TABLE IF NOT EXISTS alumno (

idAlumno INT NOT NULL AUTO\_INCREMENT,

nombre VARCHAR(100),

apellido VARCHAR(100),

idUsuario INT NOT NULL,

Legajo INT UNIQUE,

PRIMARY KEY (idAlumno),

FOREIGN KEY (idUsuario) REFERENCES Usuarios(idUsuario)

ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE = InnoDB;

-- Docente

CREATE TABLE IF NOT EXISTS docente (

idDocente INT NOT NULL AUTO\_INCREMENT,

nombre VARCHAR(100),

apellido VARCHAR(100),

idUsuario INT NOT NULL,

Legajo INT UNIQUE,

PRIMARY KEY (idDocente),

FOREIGN KEY (idUsuario) REFERENCES Usuarios(idUsuario)

ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE = InnoDB;

-- Asistencia

CREATE TABLE IF NOT EXISTS asistencia (

idAsistencia INT NOT NULL AUTO\_INCREMENT,

estado VARCHAR(50),

PRIMARY KEY (idAsistencia)

) ENGINE = InnoDB;

-- Curso

CREATE TABLE IF NOT EXISTS curso (

idCurso INT NOT NULL AUTO\_INCREMENT,

Nombre VARCHAR(45),

Materias VARCHAR(100),

Titulo VARCHAR(45),

Proyecto TINYINT,

PRIMARY KEY (idCurso)

) ENGINE = InnoDB;

-- CursoInfo

CREATE TABLE IF NOT EXISTS curso\_info (

idCursoInfo INT NOT NULL AUTO\_INCREMENT,

FechaInicio DATE,

FechaFin DATE,

Descripcion VARCHAR(200),

idCurso INT NOT NULL,

PRIMARY KEY (idCursoInfo),

FOREIGN KEY (idCurso) REFERENCES curso(idCurso)

ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE = InnoDB;

-- Horario

CREATE TABLE IF NOT EXISTS horario (

idHorario INT NOT NULL AUTO\_INCREMENT,

horario TIME,

PRIMARY KEY (idHorario)

) ENGINE = InnoDB;

-- HorarioCurso

CREATE TABLE IF NOT EXISTS horario\_curso (

idCursoInfo INT NOT NULL,

idHorario INT NOT NULL,

PRIMARY KEY (idCursoInfo, idHorario),

FOREIGN KEY (idCursoInfo) REFERENCES curso\_info(idCursoInfo),

FOREIGN KEY (idHorario) REFERENCES horario(idHorario)

) ENGINE = InnoDB;

-- Pago

CREATE TABLE IF NOT EXISTS pago (

idPago INT NOT NULL AUTO\_INCREMENT,

monto DECIMAL(10,2),

fechaPago DATE,

PRIMARY KEY (idPago)

) ENGINE = InnoDB;

-- CursoAlumnos

CREATE TABLE IF NOT EXISTS curso\_alumnos (

idCursoAlumno INT NOT NULL AUTO\_INCREMENT,

idAlumno INT NOT NULL,

idCursoInfo INT,

idAsistencia INT NOT NULL,

PRIMARY KEY (idCursoAlumno),

FOREIGN KEY (idAlumno) REFERENCES alumno(idAlumno) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (idCursoInfo) REFERENCES curso\_info(idCursoInfo),

FOREIGN KEY (idAsistencia) REFERENCES asistencia(idAsistencia)

) ENGINE = InnoDB;

-- Inscripcion

CREATE TABLE IF NOT EXISTS inscripcion (

idInscripcion INT NOT NULL AUTO\_INCREMENT,

idPago INT,

idCursoAlumno INT NOT NULL,

PRIMARY KEY (idInscripcion),

FOREIGN KEY (idPago) REFERENCES pago(idPago),

FOREIGN KEY (idCursoAlumno) REFERENCES curso\_alumnos(idCursoAlumno)

) ENGINE = InnoDB;

-- DocenteCurso

CREATE TABLE IF NOT EXISTS docente\_curso (

idDocente INT NOT NULL,

idCursoInfo INT NOT NULL,

PRIMARY KEY (idDocente, idCursoInfo),

FOREIGN KEY (idDocente) REFERENCES docente(idDocente) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (idCursoInfo) REFERENCES curso\_info(idCursoInfo) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

PROCEDURE NECESARIO:

DELIMITER $$

CREATE PROCEDURE InscribirAlumnoEnCurso\_ConUsuario(

IN p\_idUsuario INT,

IN p\_idCursoInfo INT,

IN p\_idPago INT

)

BEGIN

DECLARE v\_idAlumno INT;

DECLARE v\_idAsistencia INT;

DECLARE v\_idCursoAlumno INT;

SELECT idAlumno INTO v\_idAlumno

FROM alumno

WHERE idUsuario = p\_idUsuario;

INSERT INTO asistencia (estado) VALUES ('Ausente');

SET v\_idAsistencia = LAST\_INSERT\_ID();

INSERT INTO curso\_alumnos (idAlumno, idCursoInfo, idAsistencia)

VALUES (v\_idAlumno, p\_idCursoInfo, v\_idAsistencia);

SET v\_idCursoAlumno = LAST\_INSERT\_ID();

INSERT INTO inscripcion (idPago, idCursoAlumno)

VALUES (p\_idPago, v\_idCursoAlumno);

END$$

DELIMITER ;

INSERTS

-- 👤 Usuarios (5 alumnos + 5 docentes)

INSERT INTO Usuarios (Usuario, Contraseña, Rol, Email) VALUES

('juan123', 'pass123', 'Alumno', 'juan@example.com'),

('maria456', 'pass456', 'Alumno', 'maria@example.com'),

('pedro789', 'pass789', 'Alumno', 'pedro@example.com'),

('laura321', 'pass321', 'Alumno', 'laura@example.com'),

('carlos654', 'pass654', 'Alumno', 'carlos@example.com'),

('prof.luis', 'pass111', 'Docente', 'luis@example.com'),

('prof.ana', 'pass222', 'Docente', 'ana@example.com'),

('prof.pablo', 'pass333', 'Docente', 'pablo@example.com'),

('prof.marta', 'pass444', 'Docente', 'marta@example.com'),

('prof.sofia', 'pass555', 'Docente', 'sofia@example.com');

-- 🎓 Alumnos (legajos desde 1000)

INSERT INTO alumno (nombre, apellido, idUsuario, Legajo) VALUES

('Juan', 'Pérez', 1, 1000),

('María', 'Gómez', 2, 1001),

('Pedro', 'Rodríguez', 3, 1002),

('Laura', 'Fernández', 4, 1003),

('Carlos', 'Ramírez', 5, 1004);

-- 👨‍🏫 Docentes (legajos desde 100)

INSERT INTO docente (nombre, apellido, idUsuario, Legajo) VALUES

('Luis', 'Martínez', 6, 100),

('Ana', 'López', 7, 101),

('Pablo', 'Suárez', 8, 102),

('Marta', 'Gutiérrez', 9, 103),

('Sofía', 'Díaz', 10, 104);

-- 📚 Cursos

INSERT INTO curso (Nombre, Materias, Titulo, Proyecto) VALUES

('Programación I', 'Algoritmos, Lógica', 'Tecnicatura', 1),

('Bases de Datos', 'SQL, Modelado', 'Tecnicatura', 1),

('Matemática I', 'Álgebra, Análisis', 'Tecnicatura', 0),

('Inglés Técnico', 'Inglés nivel 1', 'Tecnicatura', 0),

('Diseño Web', 'HTML, CSS', 'Tecnicatura', 1);

-- 🗓️ Curso\_Info

INSERT INTO curso\_info (FechaInicio, FechaFin, Descripcion, idCurso) VALUES

('2025-08-01', '2025-12-15', 'Programación I - 1er Cuatrimestre', 1),

('2025-08-01', '2025-12-15', 'Bases de Datos - 1er Cuatrimestre', 2),

('2025-08-01', '2025-12-15', 'Matemática I - 1er Cuatrimestre', 3),

('2025-08-01', '2025-12-15', 'Inglés Técnico - 1er Cuatrimestre', 4),

('2025-08-01', '2025-12-15', 'Diseño Web - 1er Cuatrimestre', 5);

-- ⏰ Horario

INSERT INTO horario (horario) VALUES

('08:00:00'),

('10:00:00'),

('12:00:00'),

('14:00:00'),

('16:00:00');

-- ⏳ Horario\_Curso

INSERT INTO horario\_curso (idCursoInfo, idHorario) VALUES

(1, 1),

(1, 2),

(2, 3),

(3, 4),

(4, 5);

-- ✅ Asistencia

INSERT INTO asistencia (estado) VALUES

('Presente'),

('Ausente'),

('Presente'),

('Ausente'),

('Presente');

-- 💸 Pago

INSERT INTO pago (monto, fechaPago) VALUES

(20000.00, '2025-07-20'),

(25000.00, '2025-07-21'),

(15000.00, '2025-07-22'),

(18000.00, '2025-07-23'),

(22000.00, '2025-07-24');

-- 📖 Curso\_Alumnos

INSERT INTO curso\_alumnos (idAlumno, idCursoInfo, idAsistencia) VALUES

(1, 1, 1),

(2, 2, 2),

(3, 3, 3),

(4, 4, 4),

(5, 5, 5);

-- 📝 Inscripcion

INSERT INTO inscripcion (idPago, idCursoAlumno) VALUES

(1, 1),

(2, 2),

(3, 3),

(4, 4),

(5, 5);