

Instituto Tecnológico de Mexicali

Ingeniería Sistemas Computacionales Fundamentos De Base De Datos

Actividad:

"Examen Final de Unidad 4"

Alumno:

Delgado Rodriguez Jonhatan Eduardo

No. Control: C21490757

Profesor:

José Ramón Bogarin Valenzuela

Mexicali, B.C. a 23 de Mayo de 2025

Problema Técnico: Gestionando la Información de una Universidad

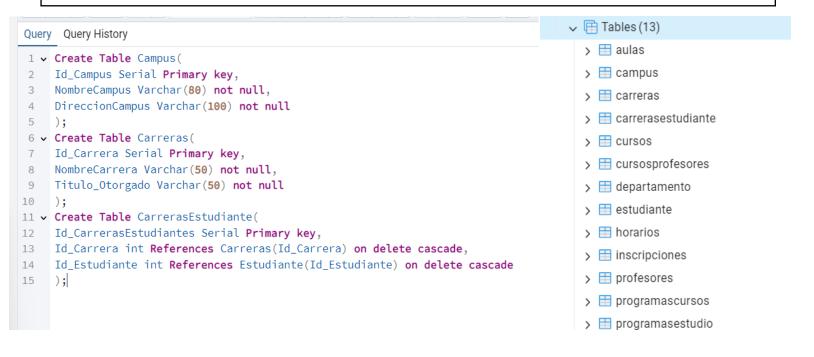
Imagina que estás trabajando en el equipo de desarrollo de software para una universidad. La universidad necesita un sistema para gestionar la información de sus estudiantes, los cursos que ofrecen, las inscripciones de los estudiantes a los cursos, los profesores que imparten los cursos y los departamentos a los que pertenecen los profesores. Tu tarea es diseñar y trabajar con la base de datos que almacenará toda esta información.

Objetivo General: Diseñar una base de datos relacional y realizar diversas operaciones para gestionar la información de la universidad. Esto incluye la creación y modificación de la estructura de las tablas, la manipulación de los datos y la realización de consultas complejas para obtener información específica.

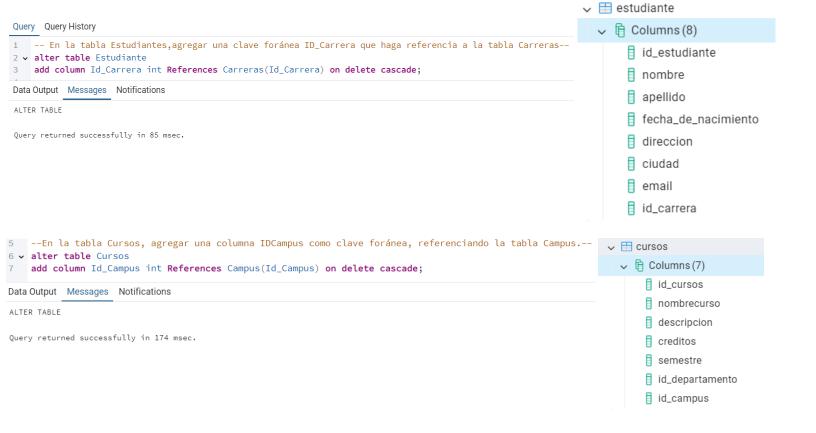
Creación de Tablas

```
Query Query History
 1 - Create Table Estudiante (
     Id_Estudiante Serial primary key,
     Nombre Varchar(50) not null,
     Apellido Varchar(50) not null,
    Fecha De Nacimiento DATE not null.
    Direccion Varchar(100) not null,
     Ciudad Varchar(50) not null,
 8 Email Varchar(50) unique not null
 9
10 Create Table Departamento (
11 Id_Departamento Serial primary key,
     NombreDepartamento Varchar(50) not null,
13 Edificio VarChar(30) not null
14 );
15
16 v Create Table Cursos (
17
    Id_Cursos Serial primary key,
18 NombreCurso Varchar(50) not null,
19 Descripcion Varchar(80) not null,
20
    Creditos int not null,
21 Semestre int not null,
22
    Id Departamento int References Departamento(Id Departamento) on delete cascade
23
24
25 Create Table Inscripciones (
26 Id_Inscripcion Serial primary key
    Id_Estudiante int References Estudiante(Id_Estudiante) on delete cascade,
    Id_Cursos int References Cursos(Id_Cursos) on delete cascade,
28
29 FechaInscripcion Date not null,
    Calificacion Decimal(10,2) not null );
Query Query History
32 Create Table Profesores(
    Id_Profesor Serial Primary Key,
34 NombreProfesor VarChar(50) not null,
35 Apellido Varchar(50)not null,
36
    Titulo VarChar(50) not null,
37 Id_Departamento int References Departamento(Id_Departamento) on delete cascade
38 );
39 v Create Table Aulas(
40 Id_Aulas Serial primary key
41
    NombreAula Varchar(50) not null,
   Capacidad int not null,
    Ubicacion Varchar(50) not null
43
44
45 Create Table Horarios(
    Id_Horario Serial primary key ,
   Id_Cursos int References Cursos(Id_Cursos) on delete cascade,
    Id_Aulas int References Aulas(Id_Aulas) on delete cascade,
48
49
    FechaInicio DATE NOT NULL.
    FechaFin DATE NOT NULL,
    HoraInicio TIME NOT NULL,
52
    HoraFin TIME NOT NULL
54
55 ∨ Create Table CursosProfesores(
    Id_CursoProfesor Serial Primary key,
    Id Cursos int References Cursos(Id Curso
                                            s) on delete cascade.
58 Id_Profesor int References Profesores(Id_Profesor) on delete cascade
59 );
Create table ProgramasEstudio(
Id_Programa Serial Primary key,
NombrePrograma Varchar(50) not null,
DescripcionPrograma Varchar(80) not null
);
Create table ProgramasCursos(
Id_ProgramaCurso Serial Primary key,
Id_Programa int References ProgramasEstudio(Id_Programa) on delete cascade,
Id_Cursos int References Cursos(Id_Cursos) on delete cascade
);
```

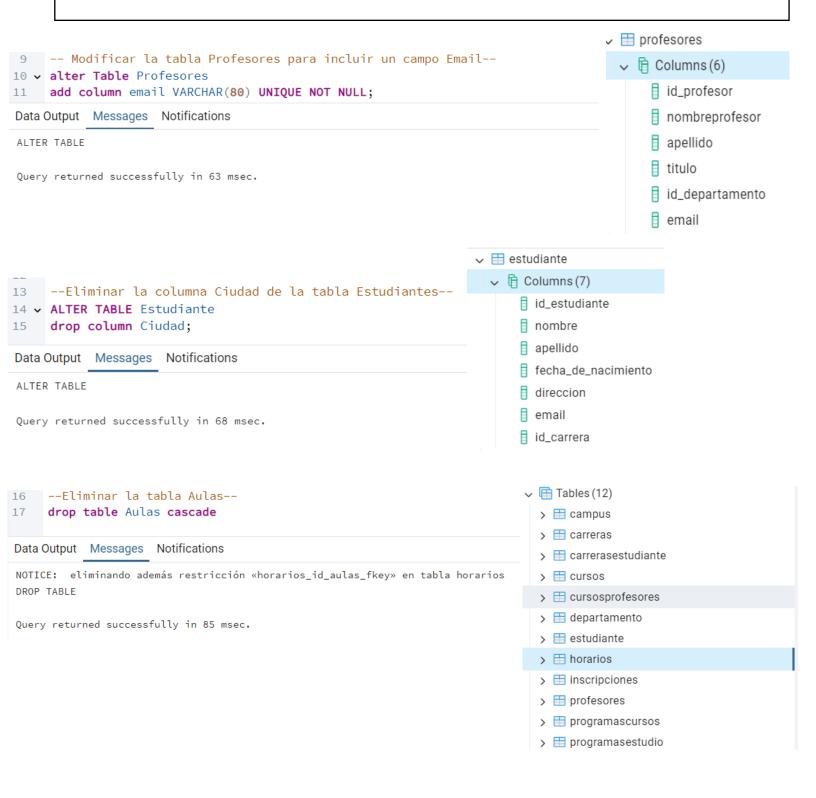
Agregar Tablas



Modificar Tablas



Modificar Tablas



Insert de Datos

Query Query History

```
--INSERT DE DATOS EN TABLAS--
    -- Tabla: departamento--
3 - INSERT INTO departamento (id_departamento, nombredepartamento, edificio) VALUES
     (1, 'Ingeniería en Sistemas', 'Edificio A1'),
     (2, 'Ingeniería Industrial', 'Edificio B1'),
    (3, 'Ciencias Básicas', 'Edificio C1'),
    (4, 'Administración', 'Edificio D1'),
    (5, 'Electromecánica', 'Edificio E1');
10 -- Tabla: campus--
11 • INSERT INTO campus (id_campus, nombrecampus, direccioncampus) VALUES
12 (1, 'Instituto Tecnológico de Mexicali', 'Av. Instituto Politécnico Nacional S/N, Col. Plutarco Elías Calles');
13
14
    -- Tabla: carreras--
15 • INSERT INTO carreras (id_carrera, nombrecarrera, titulo_otorgado) VALUES
    (1, 'Ingeniería en Sistemas Computacionales', 'Ingeniero en Sistemas Computacionales'),
     (2, 'Ingeniería Industrial', 'Ingeniero Industrial'),
     (3, 'Licenciatura en Administración', 'Licenciado en Administración'),
   (4, 'Ingeniería Electromecánica', 'Ingeniero Electromecánico'),
   (5, 'Contador Público', 'Licenciado en Contaduría');
     -- Tabla: estudiante--
23 v INSERT INTO estudiante (id_estudiante, nombre, apellido, fecha_de_nacimiento, direccion, email, id_carrera) VALUES
24 (1, 'Alejandro', 'Gómez', '2000-01-15', 'Col. Pueblo Nuevo #123', 'alejandro.gomez@itmexicali.edu.mx', 1),
     (2, 'María', 'Luna', '1999-12-05', 'Col. Independencia #456', 'maria.luna@itmexicali.edu.mx', 2),
   (3, 'Luis', 'Hernández', '2001-05-21', 'Col. Robledo #789', 'luis.hernandez@itmexicali.edu.mx', 3), (4, 'Carmen', 'Salazar', '2000-07-03', 'Col. Libertad #321', 'carmen.salazar@itmexicali.edu.mx', 4),
   (5, 'Jorge', 'Ruiz', '2000-03-30', 'Col. Jardines del Lago #654', 'jorge.ruiz@itmexicali.edu.mx', 5);
29
Query Query History
31 v INSERT INTO profesores (id_profesor, nombreprofesor, apellido, titulo, id_departamento, email) VALUES
     (1, 'Víctor', 'Medina', 'Doctor en Ciencias de la Computación', 1, 'victor.medina@itmexicali.edu.mx'),
     (2, 'Silvia', 'Ramírez', 'Maestra en Ingeniería Industrial', 2, 'silvia.ramirez@itmexicali.edu.mx'),
     (3, 'Arturo', 'Meza', 'Licenciado en Matemáticas', 3, 'arturo.meza@itmexicali.edu.mx'),
     (4, 'Rosa', 'Hernández', 'Maestra en Administración', 4, 'rosa.hernandez@itmexicali.edu.mx'),
35
36
     (5, 'Luis', 'Escobar', 'Doctor en Ingeniería Eléctrica', 5, 'luis.escobar@itmexicali.edu.mx');
     -- Tabla: programasestudio--
39 • INSERT INTO programasestudio (id_programa, nombreprograma, descripcionprograma) VALUES
     (1, 'Plan 2015 ISC', 'Programa educativo de Ingeniería en Sistemas 2015'),
     (2, 'Plan 2018 II', 'Programa educativo de Ingeniería Industrial 2018'),
     (3, 'Plan 2016 ADM', 'Programa educativo de Administración 2016'),
     (4, 'Plan 2017 ELEC', 'Programa educativo de Electromecánica 2017'),
43
     (5, 'Plan 2019 CP', 'Programa educativo de Contaduría Pública 2019');
44
     -- Tabla: cursos--
47 v INSERT INTO cursos (id_cursos, nombrecurso, descripcion, creditos, semestre, id_departamento, id_campus) VALUES
     (1, 'Estructura de Datos', 'Manejo de estructuras lineales y no lineales', 5, 4, 1, 1),
     (2, 'Estadística', 'Probabilidad y estadística aplicada', 4, 3, 2, 1),
     (3, 'Contabilidad Financiera', 'Fundamentos de contabilidad básica', 4, 2, 4, 1),
     (4, 'Circuitos Eléctricos', 'Análisis de circuitos eléctricos de CD y CA', 5, 3, 5, 1),
```

(5, 'Álgebra Lineal', 'Espacios vectoriales, matrices, determinantes', 4, 1, 3, 1);

Insert de Datos

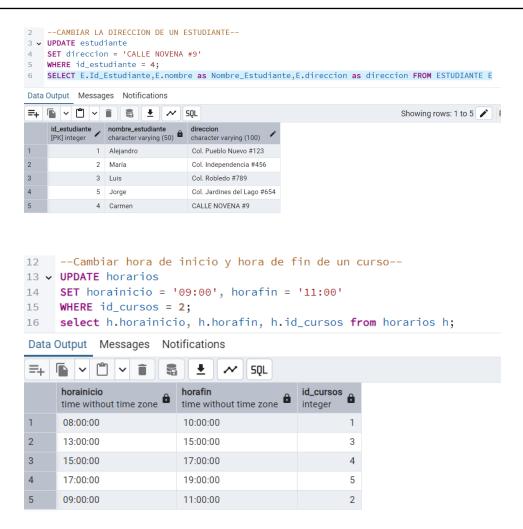
```
Query Query History
54 -- Tabla: programascursos--
55 v INSERT INTO programascursos (id_programacurso, id_programa, id_cursos) VALUES
56 (1, 1, 1),
57 (2, 2, 2),
58 (3, 3, 3),
59 (4, 4, 4),
60 (5, 1, 5);
61
    -- Tabla: cursosprofesores--
63 ▼ INSERT INTO cursosprofesores (id_cursoProfesor, id_cursos, id_profesor) VALUES
64 (1, 1, 1),
     (2, 2, 2),
66 (3, 3, 4),
67 (4, 4, 5),
     (5, 5, 3);
69
70 -- Tabla: horarios (sin aula)--
71 - INSERT INTO horarios (id_horario, id_cursos, id_aulas, fechainicio, fechafin, horarinicio, horafin) VALUES
     (1, 1, NULL, '2025-01-22', '2025-06-10', '08:00', '10:00'),
73 (2, 2, NULL, '2025-01-22', '2025-06-10', '10:00', '12:00'),
74 (3, 3, NULL, '2025-01-22', '2025-06-10', '13:00', '15:00'),
75 (4, 4, NULL, '2025-01-22', '2025-06-10', '15:00', '17:00'),
76 (5, 5, NULL, '2025-01-22', '2025-06-10', '17:00', '19:00');
78 -- Tabla: inscripciones--
79 v INSERT INTO Inscripciones (id_inscripcion, id_estudiante, id_cursos, FechaInscripcion, calificacion) VALUES
80 (1, 1, 1, '2025-01-10', 95.0),
81 (2, 2, 2, '2025-01-11', 89.5),
82 (3, 3, 3, '2025-01-12', 92.0),
     (4, 4, 4, '2025-01-13', 85.0),
(5, 5, 5, '2025-01-14', 90.5);
83
84
85
     -- Tabla: carrerasestudiante--
86
87 	imes 	extbf{INSERT INTO} carrerasestudiante (id_carrerasestudiantes, id_carrera, id_estudiante) \textbf{VALUES}
88 (1, 1, 1),
89
     (2, 2, 2),
90 (3, 3, 3),
91 (4, 4, 4),
92 (5, 5, 5);
```

Data Output Messages Notifications

INSERT 0 5

Query returned successfully in 51 msec.

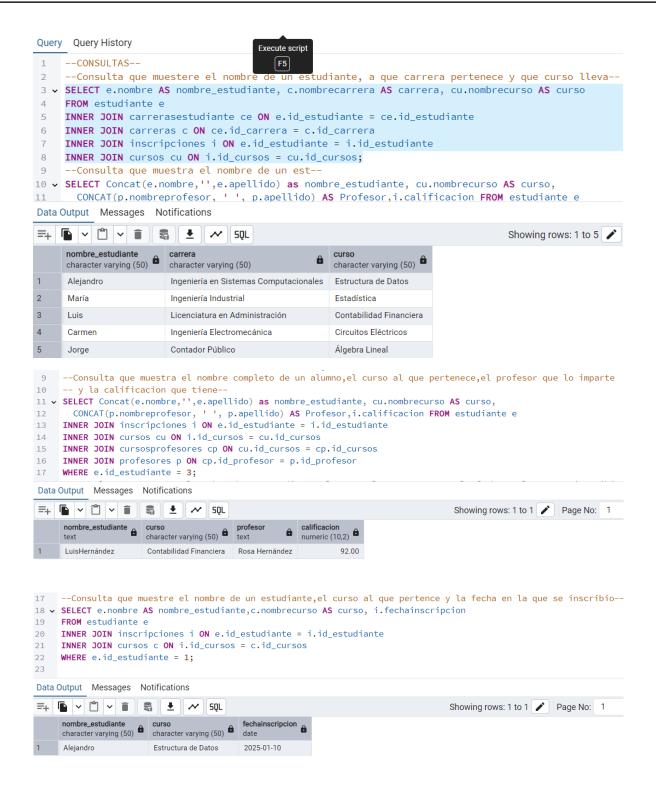
Actualizando Datos

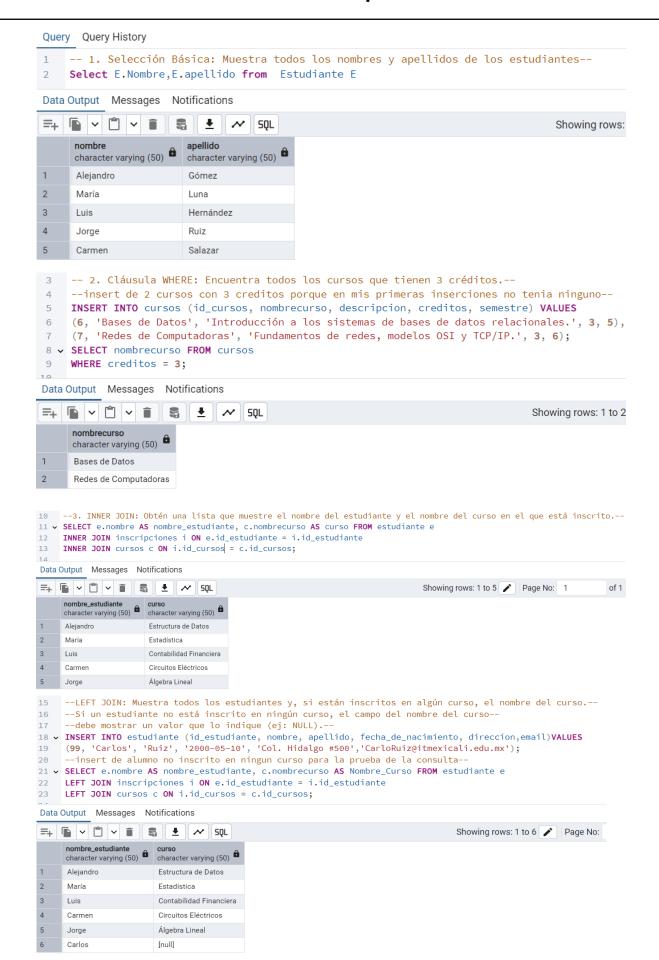


Eliminación de Datos

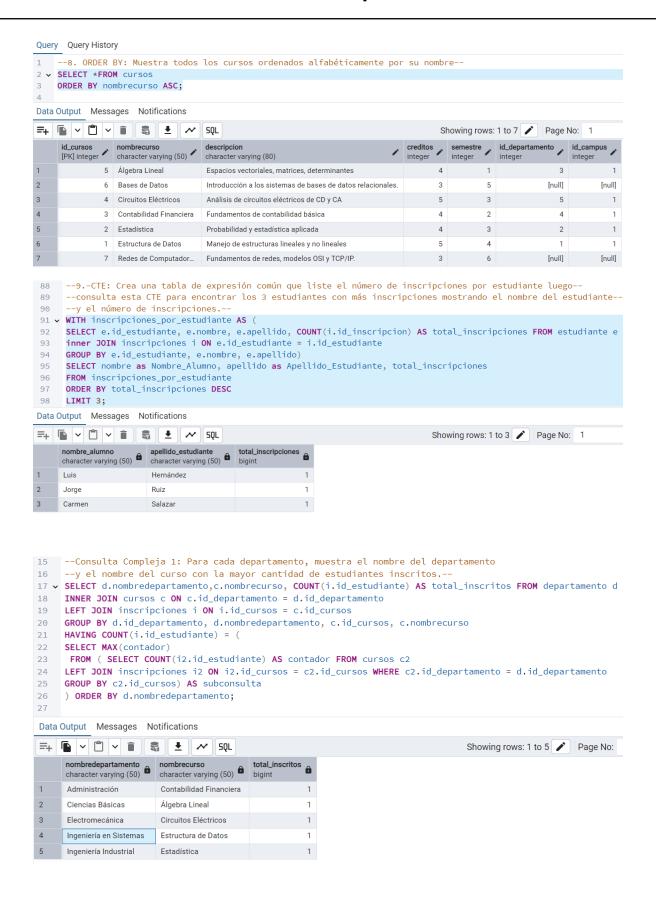


Realización de Consultas









```
28 --. Consulta Compleja 2: Encuentra a los profesores que imparten más de dos
29 --cursos, mostrando su nombre, apellido y la cantidad de cursos que imparten.--
30 -- INSERT PARA QUE FUNCIONE LA CONSULTA--
31 v INSERT INTO cursosprofesores (id_cursoprofesor,id_profesor, id_cursos) VALUES
32 (6,1, 2),
     (7,1, 3),
33
     (8,2,5),
34
35
     (9,2, 1),
36
     (10,3,4),
    SELECT p.nombreprofesor, p.apellido, COUNT (cp.id_cursos) AS cantidad_cursos FROM profesores p
37
38    JOIN cursosprofesores cp ON p.id_profesor = cp.id_profesor
39 GROUP BY p.id_profesor, p.nombreprofesor, p.apellido
40 HAVING COUNT(cp.id_cursos) > 2
41 ORDER BY cantidad_cursos DESC;
Data Output Messages Notifications
=+ | • | • | • | • |
                         $ 1

✓ | 5QL

                                                                                                Showing rows: 1 to 3
                                                cantidad_cursos
                           character varying (50)
1
      Silvia
                           Ramírez
                                                              3
2
      Víctor
                           Medina
                                                              3
3
      Arturo
                           Meza
                                                              2
42 --Consulta Compleja 3: Lista los nombres de los programas de estudio y, para cada programa--
    --el nombre del curso con el promedio de calificación más alto.-
44 v SELECT p.nombreprograma,c.nombrecurso,ROUND(AVG(i.calificacion), 2) AS promedio FROM programasestudio p
    INNER JOIN programascursos pc ON p.id_programa = pc.id_programa
46 INNER JOIN cursos c ON pc.id_cursos = c.id_cursos
    JOIN inscripciones i ON c.id cursos = i.id cursos
47
48 GROUP BY p.id_programa, p.nombreprograma, c.id_cursos, c.nombrecurso
    HAVING AVG(i.calificacion) = (SELECT MAX(prom) FROM (SELECT AVG(i2.calificacion) AS prom FROM programascursos pc2
    INNER JOIN cursos c2 ON pc2.id_cursos = c2.id_cursos
    INNER JOIN inscripciones i2 ON c2.id_cursos = i2.id_cursos
     WHERE pc2.id_programa = p.id_programa
    GROUP BY c2.id_cursos
53
54
    ) AS sub
55
    ORDER BY p.nombreprograma:
Data Output Messages Notifications
=+ 🖺 ∨ 🖺 ∨ 🛊 👼 👲 🚜 SQL
                                                                              Showing rows: 1 to 4 Page No: 1
                                                                                                                    of 1
     nombreprograma character varying (50) a nombrecurso character varying (50) character varying (50)
    Plan 2015 ISC
                      Estructura de Datos
2
     Plan 2016 ADM
                      Contabilidad Financiera
                                           92.00
     Plan 2017 ELEC
3
                      Circuitos Eléctricos
                                           85.00
     Plan 2018 II
                      Estadística
                                           89.50
```