

# Ejercicio Oracle Database

José Julián Barbosa Ayala

Michael Hernández Vera

Camilo Vega Ramírez

## Contenido

<b>Punto A</b>	<b>2</b>
Query . . . . .	2
Resultado . . . . .	3
<b>Punto B</b>	<b>3</b>
Query . . . . .	4
Resultado . . . . .	4
<b>Punto C</b>	<b>4</b>
Query . . . . .	4
Resultado . . . . .	5
<b>Punto D</b>	<b>6</b>
Query . . . . .	6
Resultado . . . . .	7
<b>Punto E</b>	<b>7</b>
Query . . . . .	7
Resultado . . . . .	8

## Punto A

*Listar todos los cursos ofrecidos en la carrera de Ingeniería de Sistemas. Para este caso se debe tener en cuenta que un mismo curso puede ser dictado por profesores diferentes, en salones diferentes y en horarios diferentes. Se deben obtener los siguientes datos: el nombre del curso, el nombre completo del profesor que dicta el curso, el salón en que se dicta el curso y la hora en la que se dicta del curso.*

## Query

**NOTA:** Es importante señalar que, aunque en el enunciado no se menciona, se ha decidido incluir la columna DIA en la consulta. Aunque en la actualidad todos los cursos tienen el mismo horario independientemente del día en que se dicten, es posible que en el futuro esto cambie con la implementación de nuevos calendarios de cursos. Al agregar la columna DIA, se prevé esta posible situación y se garantiza la flexibilidad necesaria para adaptarse a los cambios futuros.

```
SELECT DISTINCT
  cursos.nombre AS curso,
  --- Se concatenan nombre completo del profesor, el edificio/salón y la hora de inicio y fin.
  usuarios.nombre || ' ' || apellido AS profesor,
  'Ed. ' || edificios.nombre || ' Salon ' || salones.nombre AS salon,
  calendario_cursos.dia,
  hora_inicio || ' a ' || hora_fin AS horario
FROM calendario_cursos
JOIN cursos
  ON calendario_cursos.id_curso = cursos.id_curso
JOIN cursos_carreras
  ON cursos.id_curso = cursos_carreras.id_curso
JOIN carreras
  ON cursos_carreras.id_carrera = carreras.id_carrera
JOIN usuarios
  ON calendario_cursos.id_profesor = usuarios.id_usuario
JOIN salones
  ON calendario_cursos.id_salon = salones.id_salon
JOIN edificios
  ON salones.id_edificio = edificios.id_edificio
--- Filtrando por la carrera requerida
WHERE carreras.nombre = 'Ingeniería de Sistemas'
--- Ordenando por nombre del curso para facilitar la lectura
ORDER BY cursos.nombre;
```

## Resultado

CURSO	PROFESOR	SALON	DIA	HORARIO
Aprendizaje Automático	Robert Vance	Ed. El Samán Salon 1.1	Martes	04:00 PM a 06:00 PM
Aprendizaje Automático	Robert Vance	Ed. El Samán Salon 1.1	Jueves	04:00 PM a 06:00 PM
Arquitectura del Computador	Grace Stone	Ed. El Lago Salon 2.6	Lunes	04:00 PM a 06:00 PM
Arquitectura del Computador	Grace Stone	Ed. El Lago Salon 2.6	Viernes	04:00 PM a 06:00 PM
Cinemática y Dinámica	Saanvi Bahl	Ed. El Lago Salon 2.2	Lunes	02:00 PM a 04:00 PM
Cinemática y Dinámica	Saanvi Bahl	Ed. El Lago Salon 2.2	Miércoles	02:00 PM a 04:00 PM
Cinemática y Dinámica	Saanvi Bahl	Ed. El Lago Salon 2.2	Viernes	02:00 PM a 04:00 PM
Cálculo Diferencial	Grace Stone	Ed. El Lago Salon 2.5	Martes	08:00 AM a 10:00 AM
Cálculo Diferencial	Grace Stone	Ed. El Lago Salon 2.5	Jueves	08:00 AM a 10:00 AM
Cálculo Integral	Jared Vásquez	Ed. El Lago Salon 3.3	Martes	10:00 AM a 12:00 PM
Cálculo Integral	Jared Vásquez	Ed. El Lago Salon 3.3	Jueves	10:00 AM a 12:00 PM
Cálculo Multivariado	Jared Vásquez	Ed. El Lago Salon 2.1	Martes	02:00 PM a 04:00 PM
Cálculo Multivariado	Jared Vásquez	Ed. El Lago Salon 2.1	Jueves	02:00 PM a 04:00 PM
Desarrollo de Videuegos	Ben Stone	Ed. Palmas Salon 3.2	Martes	02:00 PM a 05:00 PM
Desarrollo de Videuegos	Ben Stone	Ed. Palmas Salon 3.2	Viernes	02:00 PM a 05:00 PM
Electricidad y Magnetismo	Saanvi Bahl	Ed. El Lago Salon 1.5	Lunes	04:00 PM a 06:00 PM
Electricidad y Magnetismo	Saanvi Bahl	Ed. El Lago Salon 1.5	Miércoles	04:00 PM a 06:00 PM
Electricidad y Magnetismo	Saanvi Bahl	Ed. El Lago Salon 1.5	Viernes	04:00 PM a 06:00 PM
Estructuras de Datos	Ben Stone	Ed. Almendros Salon 3.4	Martes	10:00 AM a 12:00 PM
Estructuras de Datos	Ben Stone	Ed. Almendros Salon 3.4	Jueves	10:00 AM a 12:00 PM
Introducción a la Programación	Ben Stone	Ed. El Lago Salon 1.4	Martes	08:00 AM a 10:00 AM
Introducción a la Programación	Ben Stone	Ed. El Lago Salon 1.4	Jueves	08:00 AM a 10:00 AM
Introducción a la Seguridad Informática	Robert Vance	Ed. Palmas Salon 3.5	Miércoles	04:00 PM a 06:00 PM
Introducción a la Seguridad Informática	Robert Vance	Ed. Palmas Salon 3.5	Viernes	04:00 PM a 06:00 PM
Lógica Digital y Lenguaje de Máquina	Grace Stone	Ed. El Lago Salon 1.2	Miércoles	02:00 PM a 04:00 PM
Lógica Digital y Lenguaje de Máquina	Grace Stone	Ed. El Lago Salon 1.2	Viernes	02:00 PM a 04:00 PM
Optimización Matemática	Jared Vásquez	Ed. El Lago Salon 2.2	Lunes	02:00 PM a 04:00 PM
Optimización Matemática	Jared Vásquez	Ed. El Lago Salon 2.2	Miércoles	02:00 PM a 04:00 PM
Optimización Matemática	Ulrich Nielsen	Ed. El Samán Salon 1.3	Miércoles	08:00 AM a 10:00 AM
Optimización Matemática	Ulrich Nielsen	Ed. El Samán Salon 1.3	Viernes	08:00 AM a 10:00 AM
Programación Paralela	Ulrich Nielsen	Ed. Palmas Salon 3.3	Martes	08:00 AM a 10:00 AM
Programación Paralela	Ulrich Nielsen	Ed. Palmas Salon 3.3	Jueves	08:00 AM a 10:00 AM
Práctica Estudiantil	Jared Vásquez	Ed. El Samán Salon 1.6	Viernes	02:00 PM a 04:00 PM
Sistemas Inteligentes	Robert Vance	Ed. Cedro Rosado Salon 2.2	Lunes	10:00 AM a 12:00 PM
Sistemas Inteligentes	Robert Vance	Ed. Cedro Rosado Salon 2.2	Miércoles	10:00 AM a 12:00 PM
Teología	Zeke Landon	Ed. El Samán Salon 1.8	Lunes	10:00 AM a 12:00 PM
Técnicas y Prácticas de Programación	Ben Stone	Ed. Almendros Salon 3.4	Lunes	04:00 PM a 06:00 PM
Técnicas y Prácticas de Programación	Ben Stone	Ed. Almendros Salon 3.4	Miércoles	04:00 PM a 06:00 PM
Álgebra Lineal	Grace Stone	Ed. El Lago Salon 3.1	Martes	10:00 AM a 12:00 PM
Álgebra Lineal	Grace Stone	Ed. El Lago Salon 3.1	Jueves	10:00 AM a 12:00 PM
Ética	Mikaela Stone	Ed. Almendros Salon 3.1	Miércoles	08:00 AM a 10:00 AM
Ética	Jared Vásquez	Ed. Almendros Salon 3.7	Viernes	08:00 AM a 10:00 AM

## Punto B

*Obtener la lista de profesores que dictan cursos pertenecientes a la facultad de Humanidades. Se deben obtener los siguientes datos: el nombre completo del profesor y el nombre del curso que dicta.*

## Query

```
SELECT DISTINCT
--- Se concatenan nombre completo del profesor.
    usuarios.nombre || ' ' || apellido AS profesor,
    cursos.nombre AS curso
FROM calendario_cursos
JOIN cursos
    ON calendario_cursos.id_curso = cursos.id_curso
JOIN facultades
    ON cursos.id_facultad = facultades.id_facultad
JOIN usuarios
    ON calendario_cursos.id_profesor = usuarios.id_usuario
--- Filtrando por la facultad requerida
WHERE facultades.nombre = 'Humanidades y Ciencias Sociales'
--- Ordenando por profesor para facilitar la lectura
ORDER BY profesor;
```

## Resultado

PROFESOR	CURSO
Jared Vásquez	Ética
Mikaela Stone	Ética
Zeke Landon	Teología
Zeke Landon	Filosofía Antigua
Zeke Landon	Antropología Filosófica
Zeke Landon	Filosofía de la Ciencia

## Punto C

Obtener la lista de profesores que dictan cursos en **dos** carreras diferentes. Se deben obtener los siguientes datos: el nombre completo del profesor, el nombre del curso que dicta y el nombre de la carrera a la que pertenece el curso. Pista: averigua cómo funciona la sentencia **WITH**.

## Query

```
WITH p2c AS(
--- El subquery p2c contiene aquellos profesores que dictan en 2 carreras
SELECT DISTINCT
    usuarios.id_usuario,
    usuarios.nombre,
    apellido
FROM calendario_cursos
```

```

JOIN cursos
  ON calendario_cursos.id_curso = cursos.id_curso
JOIN cursos_carreras
  ON calendario_cursos.id_curso = cursos_carreras.id_curso
JOIN carreras
  ON cursos_carreras.id_carrera = carreras.id_carrera
JOIN usuarios
  ON calendario_cursos.id_profesor = usuarios.id_usuario
GROUP BY usuarios.id_usuario, usuarios.nombre, apellido
  HAVING COUNT(DISTINCT carreras.id_carrera) = 2
)
--- Una vez con los profesores identificados se adiciona curso y carreras
SELECT DISTINCT
--- Se concatenan nombre completo del profesor.
  p2c.nombre || ' ' || p2c.apellido AS profesor,
  cursos.nombre AS curso,
  carreras.nombre AS carrera
FROM p2c
JOIN calendario_cursos
  ON p2c.id_usuario = calendario_cursos.id_profesor
JOIN cursos
  ON calendario_cursos.id_curso = cursos.id_curso
JOIN cursos_carreras
  ON cursos.id_curso = cursos_carreras.id_curso
JOIN carreras
  ON cursos_carreras.id_carrera = carreras.id_carrera
--- Ordenando por profesor y carrera para facilitar la lectura
ORDER by profesor, carrera;

```

## Resultado

PROFESOR	CURSO	CARRERA
Ben Stone	Introducción a la Programación	Ingeniería de Sistemas
Ben Stone	Estructuras de Datos	Ingeniería de Sistemas
Ben Stone	Técnicas y Prácticas de Programación	Ingeniería de Sistemas
Ben Stone	Desarrollo de Videouegos	Ingeniería de Sistemas
Ben Stone	Estructuras de Datos	Matemáticas Aplicadas
Claudia Tiedemann	Química y Ciencia de Materiales	Ingeniería Mecánica
Claudia Tiedemann	Propiedades de los Materiales	Ingeniería Mecánica
Claudia Tiedemann	Diseño Mecánico	Ingeniería Mecánica
Claudia Tiedemann	Máquinas Térmicas e Hidráulicas	Ingeniería Mecánica
Claudia Tiedemann	Álgebra Moderna	Matemáticas Aplicadas
Ulrich Nielsen	Programación Paralela	Ingeniería de Sistemas
Ulrich Nielsen	Optimización Matemática	Ingeniería de Sistemas
Ulrich Nielsen	Optimización Matemática	Matemáticas Aplicadas

## Punto D

*Para un estudiante en particular, obtener el listado de cursos que puede matricular. Para este caso particular se debe tener en cuenta que los cursos se dictan en semestres diferentes, pertenecen a carreras diferentes y un estudiante no puede matricular un curso que ya se encuentre matriculado. Se deben obtener los siguientes datos: el nombre del curso, el nombre de la carrera a la que pertenece y el semestre en el que se ubica. Pista: averigua el uso de la sentencia **NOT EXISTS**.*

### Query

**NOTA:** Este query asume que un estudiante solo puede matricular cursos de su carrera que sean de un semestre igual o anterior al que actualmente está cursando. La lista muestra únicamente los cursos disponibles en los que el estudiante seleccionado no se encuentra ya matriculado. Para encontrar los cursos disponibles para un estudiante en particular, simplemente sustituir su nombre y apellido en la cláusula **WHERE** al final del query.

```
--- En esta primera parte se identifican todos los cursos a los cuales pueden acceder
--- Los estudiantes de acuerdo al semestre en el que se encuentran
SELECT DISTINCT
    cursos.nombre as curso,
    carreras.nombre carrera,
    cursos_carreras.semestre
FROM usuarios
JOIN carreras_estudiantes
    ON usuarios.id_usuario = carreras_estudiantes.id_usuario
JOIN carreras
    ON carreras_estudiantes.id_carrera = carreras.id_carrera
JOIN cursos_carreras
    ON carreras.id_carrera = cursos_carreras.id_carrera
    AND usuarios.semestre >= cursos_carreras.semestre
JOIN cursos
    ON cursos_carreras.id_curso = cursos.id_curso
--- Usando NOT EXISTS quitamos de la lista de cursos que puede matricular los
--- estudiantes aquellos cursos en los que ya se encuentran matriculados
WHERE NOT EXISTS (
    SELECT null
    FROM cursos_estudiantes
    JOIN calendario_cursos
        ON cursos_estudiantes.id_calendario = calendario_cursos.id_calendario
    WHERE cursos_estudiantes.id_usuario = usuarios.id_usuario
        AND calendario_cursos.id_curso = cursos_carreras.id_curso
)
--- Filtramos por nombre y apellido del estudiante del que deseamos ver qué cursos tiene
--- tiene disponibles para matricular
    AND usuarios.nombre = 'Olive'
    AND usuarios.apellido = 'Stone'
--- Ordenando por curso y semestre para facilitar la lectura.
ORDER BY cursos_carreras.semestre;
```

## Resultado

CURSO	CARRERA	SEMESTRE
Fundamentos de Matemáticas	Matemáticas Aplicadas	1
Geometría Plana y Analítica	Matemáticas Aplicadas	1
Estructuras de Datos	Matemáticas Aplicadas	2
Matemáticas Discretas	Matemáticas Aplicadas	2
Humanidades	Matemáticas Aplicadas	2
Lógica para Ciencias de la Computación	Matemáticas Aplicadas	3
Ecuaciones Diferenciales	Matemáticas Aplicadas	5
Estadística Aplicada	Matemáticas Aplicadas	5

## Punto E

*Listar los estudiantes que se han inscrito a un curso determinado. Se deben obtener los siguientes datos: el nombre completo del estudiante y el nombre del curso.*

## Query

**NOTA:** Es posible sustituir el nombre del curso deseado en la cláusula **WHERE** para encontrar a los estudiantes que se hayan matriculado en él.

```
SELECT DISTINCT
--- Se concatena nombre completo del estudiante.
    usuarios.nombre || ' ' || apellido AS estudiante,
    cursos.nombre AS curso
FROM cursos_estudiantes
JOIN calendario_cursos
    ON cursos_estudiantes.id_calendario = calendario_cursos.id_calendario
JOIN cursos
    ON calendario_cursos.id_curso = cursos.id_curso
JOIN usuarios
    ON cursos_estudiantes.id_usuario = usuarios.id_usuario
--- Se filtra por el curso del cual queremos ver la lista de estudiantes
WHERE cursos.nombre = 'Cálculo Diferencial'
--- Ordenando por estudiante para facilitar la lectura.
ORDER BY estudiante;
```

## Resultado

ESTUDIANTE	CURSO
Bartosz Tiedemann	Cálculo Diferencial
Franziska Doppler	Cálculo Diferencial
Jonas Kahnwald	Cálculo Diferencial
Mads Nielsen	Cálculo Diferencial
Magnus Nielsen	Cálculo Diferencial
Martha Nielsen	Cálculo Diferencial
Olive Stone	Cálculo Diferencial