

**Se tienen las siguientes relaciones de una base de datos de una Universidad.**

1. Persona (**DNI**, Nombre, Apellido, Ciudad, Dirección) Clave primaria (DNI)
2. Alumno (**IdAlumno**, DNI)  
Clave primaria (IdAlumno)  
DNI clave foránea de Persona.
3. Profesor (**IdProfesor**, DNI)  
Clave primaria (IdProfesor)  
DNI clave foránea de Persona.
4. Asignatura (**IdAsignatura**, Nombre, Créditos, semestre, coste, idProfesor, idTitulacion)  
Clave primaria (IdAsignatura)  
IdProfesor es clave foránea de Profesor  
IdTitulacion es clave foránea de Titulacion
5. AlumnoAsignatura (**IdAlumno, IdAsignatura**, NumeroMatricula, nota)  
Clave primaria (IdAlumno, IdAsignatura)  
IdAlumno clave foránea de Alumno.  
IdAsignatura clave foránea de Asignatura
6. Titulacion(**idTitulacion**, nombre)  
Clave primaria (IdTitulación)

USANDO SQL RESUELVA LAS SIGUIENTES PREGUNTAS

1. (0.5) Para cada profesor seleccione el nombre de la asignatura que tengan más créditos entre las que este dicta.

```
SELECT idProfesor, nombre, MAX(CREDITOS)
FROM ASIGNATURA
GROUP BY idProfesor)
```

2. (0.5) Hallar el nombre y los créditos de las asignaturas no han sido matriculadas por ningún alumno.

```
SELECT nombre, credits FROM Asignatura WHERE idAsignatura NOT IN (SELECT DISTINCT
idAsignatura FROM AlumnoAsignatura)
```

3. (0.5) Explicar en máximo cuatro líneas cuál es el resultado de la siguiente operación:

```
SELECT DISTINCT idprofesor
FROM Asignatura AS ext
WHERE NOT EXISTS
    (SELECT *
     FROM titulacion AS T
     WHERE NOT EXISTS
        (SELECT idprofesor
         FROM asignatura
         WHERE idprofesor = ext.idprofesor and
              Idtitulacion = T.Idtitulacion ) )
```

RTA:

Selecciona los identificadores de los profesores que tengan asignaturas de tal forma que todo su conjunto de asignaturas abarquen todas las titulaciones que se encuentren en la tabla Titulaciones

4. (0.5) Hallar las asignaturas que su coste sea menor o igual al 10% del valor total de su titulación.  
Explicación: El valor total de la titulación es la suma del costo de todas las asignaturas pertenecientes a esa titulación.

RTA:

```
SELECT * FROM Asignatura A1 WHERE coste <= (SELECT sum(coste) * 0.1 FROM Asignatura A2
WHERE A1.IdTitulacion = A2.IdTitulacion)
```

5. (0.5) Hallar las personas que viven en la ciudad en donde viven más personas.

RTA:

```
SELECT DNI, Nombre, Apellido
FROM PERSONA
WHERE ciudad IN
    (SELECT ciudad
     FROM PERSONA
     GROUP BY Ciudad
     HAVING COUNT(*) >= ALL
     SELECT COUNT(*)
     FROM PERSONA
     GROUP BY ciudad)
```

6. (0.5) Mostrar el id del alumno (idalumno) y el costo que cada estudiante paga en promedio por semestre.  
Explicación. Cada asignatura tiene su costo, así cuando un estudiante se matricula, podemos saber cuanto paga cada estudiante por semestre sumando el costo de las asignaturas vistas en ese semestre.

Por ejemplo:

Alumno	semestre	costo del semestre
1021	2005-1	300.000
1021	2005-2	380.000
1021	2006-1	450.000
1022	2005-1	150.000
1022	2005-2	150.000
1022	2006-1	100.000

Las primera tupla quiere decir que el alumno 1021 en el semestre 2005-1 tomó varias materias que en total le costaron 300.000  
Igualmente, la segunda tupla quiere decir que el mismo alumno 1021 en el semestre 2005-2 tomó varias materias que en total le costaron 380.000

Se pide mostrar cuanto paga en promedio cada estudiante, es decir:

Alumno	costo promedio
1021	376.667
1022	133.333

Las primera tupla quiere decir que el alumno 1021 paga en promedio por un semestre 376.667, es decir  $(300.000 + 380.000 + 450.000) / 3 = 376.667$   
Las segunda tupla quiere decir que el alumno 1022 paga en promedio por un semestre 133.333, es decir  $(150.000 + 150.000 + 100.000) / 3 = 133.333$

```
SELECT idAlumno, AVG(coste_semestre) as coste_promedio
FROM [ (SELECT idAlumno, semestre, sum(coste) as coste_semestre
      FROM (SELECT AA.idAlumno, A.idAsignatura, A.coste, A.semestre
            FROM AlumnoAsignatura AA, asignatura A
            WHERE AA.idAsignatura=A.idAsignatura)
      GROUP BY semestre) ]
GROUP BY idAlumno
```

Taller No. 3 - SQL

Presentado por: Diana María Madrid Madrid

---

7. (1.0) Para cada asignatura hallar los estudiantes con una nota mayor al promedio obtenido por todos los estudiantes de esa asignatura. Cada asignatura está asociada a un solo semestre. Mostrar el Id de la asignatura (IdAsignatura), el id del alumno (IdAlumno), y la nota. Es decir, si el promedio de la nota de los estudiantes que cursaron “Algebra” es 4.2, entonces debo mostrar todos los estudiantes que en su nota de “Algebra” sacaron más de 4.2. Idem para las otras asignaturas.

Un ejemplo del resultado esperado es:

IdAlumno	IdAsignatura	nota
1020	2020	4.5
1021	2020	4.4
1021	2021	4.0
1021	2022	4.1
...	.....	

Las dos primeras tuplas quieren decir que los alumnos 1020 y 1021 tienen una nota en la materia 2020 superior al promedio de la nota de todos aquellos alumnos que cursaron la materia 2020. La tercer tupla quiere decir que el alumno 1021 es uno de los que en la materia 2021 sacó una nota superior al promedio de todos aquellos estudiantes que cursaron la materia 2021.

```
SELECT AA.idAlumno, AA.idAsignatura, AA.nota
FROM AlumnoAsignatura AA, asignatura A
WHERE AA.idAsignatura=A.idAsignatura
and AA.nota> ( SELECT AA.idAlumno, AA.idAsignatura, AVG(AA.nota)
FROM AlumnoAsignatura AA, asignatura A
WHERE AA.idAsignatura=A.idAsignatura
GROUP BY AA.nota)
```

8. (0.5) Mostrar aquellas personas que han sido profesores y alumnos en esta universidad.

```
SELECT pf.dni, pe.nombre, pe.apellido FROM Profesor Pf INNER JOIN Persona Pe ON Pf.dni = pe.dni
WHERE pf.dni IN (SELECT dni FROM Alumno)
```

9. (0.5) Explicar qué pasa con la siguiente instrucción. Es decir como la procesa el sistema de gestión de bases de datos. Asuma que no hay errores de sintaxis.

```
DELETE Asignatura
WHERE NOT EXISTS ( SELECT *
FROM AlumnoAsignatura
WHERE Asignatura.IdAsignatura = AlumnoAsignatura.IdAsignatura)
```

**Esta consulta busca primero en la tabla AlumnoAsignatura si existe algún alumno matriculado en una asignatura en especial. Si no hay ningún alumno matriculado en dicha asignatura entonces el motor de base de datos procede a borrar dicha asignatura de la tabla Asignatura**