

Examen de prácticas de Bases de Datos

Parte 2. Problemas de prácticas

!!!Leer cuidadosamente las normas antes de comenzar el examen!!!

Se le proporcionan 5 enunciados correspondientes a 5 consultas que deberá responder. Las consultas trabajan sobre 4 tablas que almacenan información de una universidad: estudiantes, profesores, asignaturas y matrículas. Las particularidades de cada tabla se recogen en la hoja anexa en dos formatos:

- 1) Script de creación de la base de datos.
- 2) Diagrama de la base de datos.

```
CREATE TABLE estudiantes (  
  est_id int(11) NOT NULL,  
  est_dni varchar(9) NOT NULL,  
  est_apellidos varchar(40) NOT NULL,  
  est_nombre varchar(20) NOT NULL,  
  est_direccion varchar(150) DEFAULT NULL,  
  est_ciudad varchar(50) DEFAULT NULL,  
  est_provincia varchar(50) DEFAULT NULL,  
  PRIMARY KEY (est_id)  
);
```

```
CREATE TABLE profesores (  
  pro_id int(11) NOT NULL,  
  pro_dni varchar(9) NOT NULL,  
  pro_apell varchar(30) NOT NULL,  
  pro_nombre varchar(30) NOT NULL,  
  PRIMARY KEY (pro_id)  
);
```

```
CREATE TABLE asignaturas (  
  asi_id int(11) NOT NULL,  
  asi_nombre varchar(20) DEFAULT NULL,  
  asi_titulacion varchar(50) DEFAULT NULL,  
  asi_semestre char(1) DEFAULT NULL,  
  asi_profesor int(11) NOT NULL,  
  PRIMARY KEY (asi_id),  
  FOREIGN KEY (asi_profesor) REFERENCES profesores (pro_id)  
);
```

```
CREATE TABLE matriculas (  
  mat_id int(11) NOT NULL,  
  mat_estudiante int(11) NOT NULL,  
  mat_profesor int(11) NOT NULL,  
  mat_asignatura int(11) NOT NULL,  
  mat_fecha datetime NOT NULL,  
  PRIMARY KEY (mat_id),  
  FOREIGN KEY (mat_estudiante) REFERENCES estudiantes (est_id),  
  FOREIGN KEY (mat_profesor) REFERENCES profesores (pro_id),  
  FOREIGN KEY (mat_asignatura) REFERENCES asignaturas (asi_id)
```

```

mat_estudiante int(11) NOT NULL,
mat_asignatura int(11) NOT NULL,
mat_curso varchar(6) NOT NULL,
mat_nota_acta float DEFAULT NULL,
PRIMARY KEY (mat_estudiante,mat_asignatura,mat_curso),
FOREIGN KEY (mat_asignatura) REFERENCES asignaturas (asi_id),
FOREIGN KEY (mat_estudiante) REFERENCES estudiantes (est_id)
);

```



1. Listado que muestre los profesores que han dado clase al estudiante con DNI '31442155K'. Muestre todos los atributos de cada profesor.

```
SELECT * FROM profesores
WHERE pro_id IN (
    SELECT asi_profesor FROM asignaturas
    WHERE asi_id IN (
        SELECT mat_asignatura FROM matriculas
        WHERE mat_estudiante IN (
            SELECT est_id FROM estudiantes
            WHERE est_dni = '31442155K'))));
```

2. Listado de estudiantes del curso actual que no están matriculados en la asignatura de Bases de Datos. El listado debe mostrar el dni, nombre y apellidos de los estudiantes, y debe estar ordenados por apellidos, nombre y DNI. El curso actual se almacena como '201718'.

```
SELECT est_dni, est_apellidos, est_nombre
FROM estudiantes, matriculas
WHERE est_id = mat_estudiante
AND mat_curso = '201718'
AND est_id NOT IN (select est_id
FROM estudiantes, matriculas, asignaturas
WHERE est_id = mat_estudiante
AND mat_curso = '201718'
AND asi_id = mat_asignatura
AND asi_nombre = 'Bases de Datos')
ORDER BY est_apellidos, est_nombre, est_dni;
```

3. Se desean obtener un listado resumen de las asignaturas matriculadas por el estudiante con DNI 31442155K. El listado deberá mostrar tres columnas: nombre de asignatura, número de veces que se ha matriculado en esa asignatura y máxima nota en acta para dicha asignatura. El listado deberá ordenarse con respecto al número de matrículas por asignatura (de mayor a menor).

```
SELECT asi_nombre, count(*), max(mat_nota_acta)
FROM asignaturas, matriculas, estudiantes
WHERE est_dni = '31442155K'
AND mat_estudiante = est_id
AND mat_asignatura = asi_id
GROUP BY asi_nombre
ORDER BY count(*) DESC;
```

4. Listado de profesores que imparten clases en asignaturas de titulaciones de cualquier Ingeniería (por ejemplo, Grado en Ingeniería Informática, Grado en Ingeniería Industrial, etc). En el listado deberá aparecer el nombre y apellidos del profesor, la titulación en la que da clase y la asignatura. Un profesor deberá aparecer en listado tantas veces como asignaturas imparta.

```
SELECT pro_nombre, pro_apell, asi_titulacion,
asi_nombre
FROM profesores, asignaturas
WHERE asi_profesor = pro_id
AND asi_titulacion LIKE '%Ingeniería%';
```

5. Se desea obtener un listado que muestre las ciudades en las que hay menos matriculados en las titulaciones que ofrece la universidad. Para ello, se nos pide un listado que recoja para el último curso ('201718') el número de estudiantes que hay para cada titulación en cada ciudad. Debe aparecer las ciudades con menos de 20 estudiantes y el listado se debe ordenar de menor a mayor número de estudiantes. Se deben mostrar tres columnas:

Población	Titulación	Estudiantes
-----	-----	-----
Cádiz	Grado en Ingeniería Informática	9
Cádiz	Grado en Fisioterapia	12
Chiclana	Grado en Fisioterapia	19
....		

```
SELECT est_ciudad, asi_titulacion, count(*)
FROM estudiantes, asignaturas, matriculas
WHERE mat_curso = '201718'
AND mat_asignatura = asi_id
AND est_id = mat_estudiante
GROUP BY est_ciudad, asi_titulacion
HAVING count(*) < 20
ORDER BY count(*) ASC;
```