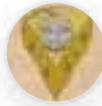


WUOLAH



QuesoViejo_
www.wuolah.com/student/QuesoViejo_

★ 128969

Examen-2.pdf

Ejercicios Resueltos SQL



2º Ficheros y Bases de Datos



Grado en Ingeniería Informática



Escuela Politécnica Superior
Universidad Carlos III de Madrid

ENCENDER TU LLAMA
CUESTA MUY POCO

BURN.COM

BURN
ENERGY DRINK

#StudyOnFire



Examen de prácticas de Bases de Datos

Parte 2. Problemas de prácticas

!!!Leer cuidadosamente antes de comenzar las consultas!!!

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) Código de creación de la base de datos.
- 2) Diagrama de la base de datos.

Consultas:

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



¿Harto de chapar algo que **no te renta?**

Olvida tus apuntes este verano
y ponte a programar 🧑💻

Si no encuentras tu crush, por lo menos
dedícate a algo que te guste.



<http://bit.ly/necesitouncambio>



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

```
SELECT p.*
```

```
FROM profesores P, estudiantes e,  
asignaturas a, matriculas m
```

```
WHERE m.mat_estudiante = e.est_id  
AND m.mat_asignatura = a.asi_id  
AND a.asi_profesor = p.pro_id  
AND e.est_dni = "31442155K"
```

```
GROUP BY p.pro_id ; //Este group by para que  
// no muestre repetidos.  
// Se agrupa por la clave  
// primaria.
```

* Ejercicio 3, otra manera

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

```
SELECT DISTINCT p.* //para que no muestre repetidos
FROM profesores p, estudiantes e,
asignaturas a, matriculas m
WHERE m.mat_estudiante = e.est_id
AND m.mat_asignatura = a.asi_id
AND a.asi_profesor = p.pro_id
AND e.est_dni = "31442155K";
```


UNDERWATER

31 DE ENERO
EN CINES

¿Exámenes?
Hay cosas más agobiantes

★NOTA: Equivalencias

IN \Leftrightarrow = ANY

NOT IN \Leftrightarrow != ALL

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

```
SELECT DISTINCT p.*
```

```
FROM profesores p
```

```
WHERE p.pro_id = ANY // IN
```

```
( SELECT a.asi_profesor  
FROM asignaturas a
```

```
where a.asi_id = ANY // IN
```

```
( SELECT m.mat_asignatura
```

```
FROM matriculas m
```

```
WHERE m.mat_estudiante = ANY // IN
```

```
( SELECT e.est_id
```

```
FROM estudiante e
```

```
WHERE e.est_dni = "31442155K" ) ) ) ;
```



QuesoViejo_ WUOLAH

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 e.est-dni, e.est.nombre, e.est.apellidos  
FROM estudiantes e  
WHERE e.est-id NOT IN // != ALL
```

```
(SELECT e2.est-id  
FROM estudiantes e2, matriculas m, asignaturas a  
WHERE e2.est-id = m.mat-estudiantes  
AND a.asi-id = m.mat-asignatura  
AND m.mat-curso = "201718"  
AND a.asi-nombre = "Bases de Datos"  
)  
ORDER BY e.est-apellidos, e.est.nombre, e.est-dni ;
```


Otra forma, con consultas anidadas en vez de producto natural

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 e.est_dni, e.est_nombre, e.est_apellidos
FROM estudiantes e
WHERE e.est_id NOT IN // != ALL
( SELECT e2.est_id
  FROM estudiantes e2
  WHERE e2.est_id IN // = ANY
    ( SELECT m.mat_estudiante
      FROM matriculas m
      WHERE m.mat_curso = "201718"
        AND m.mat_asignatura IN // = ANY
          ( SELECT a.asi_id
            FROM asignaturas a
            WHERE a.asi_nombre = "Bases de Datos"
          )
        )
    )
ORDER BY e.est_apellidos, e.est_nombre, e.est_dni;
```

QuesoViejo_ WUOLAH

¡Sube una stories con tus apuntes y menciónanos! @Wuolah_apuntes

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 a.asi_nombre, COUNT(*) "nveces",  
       MAX(m.mat-nota-acta) "MaxNota"
```

```
FROM asignaturas a, matriculas m  
WHERE a.asi_id = m.mat-asignatura  
      AND m.mat-estudiante =  
      (SELECT e.est-id  
       FROM estudiantes e  
       WHERE e.est-dni = "33442155K")
```

```
GROUP BY m.mat-estudiante, m.mat-asignatura  
ORDER BY nveces DESC;
```

***NOTA** m.mat-asignatura es clave foránea a asignaturas, así que es como si agrupáramos por a.asi-id pudiendo mostrar en el select a.asi_nombre.

UNDERWATER

**31 DE ENERO
EN CINES**

¿Exámenes?

Hay cosas más agobiantes

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.

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. Deben aparecer las ciudades con menos de 20 estudiantes y el listado se debe ordenar de menor a mayor número de estudiantes. Un ejemplo de resultado que debe generar su consulta sería el siguiente:

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



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 p.pro-nombre, p.pro-apell, a.asi-titulacion,  
       a.asi-nombre
```

```
FROM profesores p, asignaturas a  
WHERE p.pro-id = a.asi-profesor  
      AND a.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. Deben aparecer las ciudades con menos de 20 estudiantes y el listado se debe ordenar de menor a mayor número de estudiantes. Un ejemplo de resultado que debe generar su consulta sería el siguiente:

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

```
SELECT e.est-ciudad "Población",  
       a.asi-titulacion "Titulación",  
       COUNT(DISTINCT est_id) "Nestudiantes"
```

```
FROM estudiantes e, asignaturas a, matriculas m  
WHERE e.est_id = m.mat-estudiante  
      AND a.asi_id = m.mat-asignatura  
      AND m.mat-curso = "201718"
```

```
GROUP BY e.est-ciudad a.asi-titulacion  
HAVING Nestudiantes < 20  
ORDER BY Nestudiantes ASC;
```

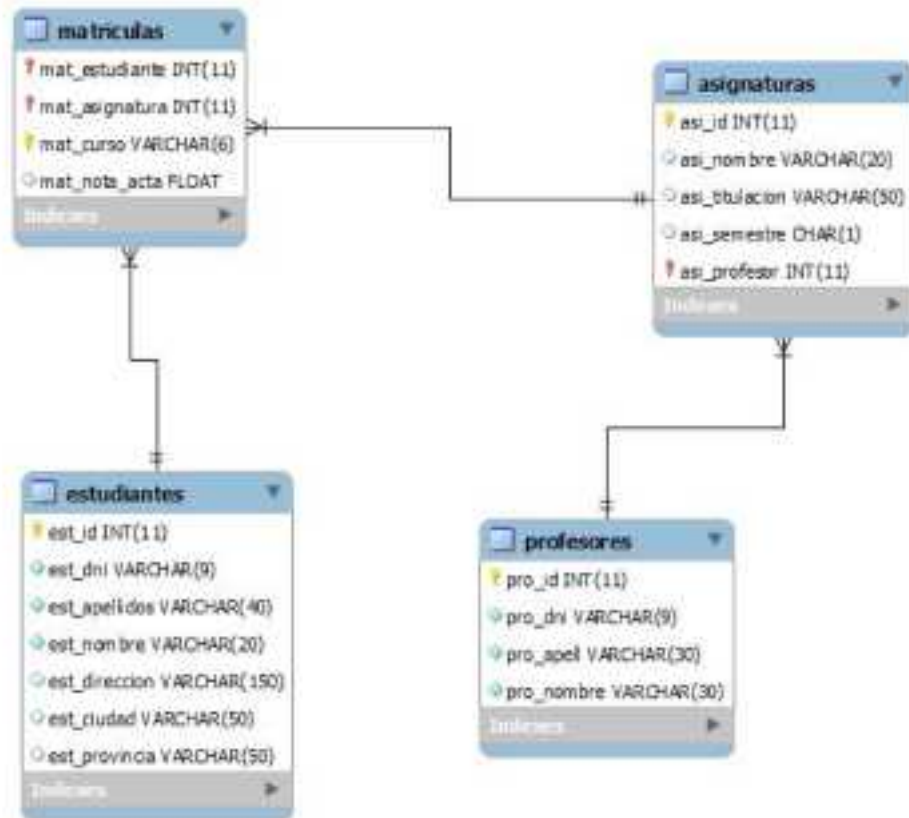

Anexo con código de creación de tablas y diagrama correspondiente

```
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_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)  
);
```

UNDERWATER

31 DE ENERO
EN CINES

¿Exámenes?
Hay cosas más agobiantes



La arista de unión entre dos tablas, muestra que hay una relación entre dichas tablas, pero para saber con certeza qué campos son los que están relacionados (clave primaria - clave foránea), observe el código SQL de creación de la tabla.



QuesoViejo_ WUOLAH