

Creación de la B.D. AULA_GUIRI_SQL que contenga la siguiente información:

- 1. Crear en SQL las tablas con sus claves primarias, no crearéis desde un principio las claves ajenas, se hará posteriormente. El que pueda hacerlo en la creación de las tablas que lo haga.

```
CREATE TABLE IF NOT EXISTS profesores (  
    dni_prof INT AUTO_INCREMENT PRIMARY KEY,  
    nombre_prof VARCHAR(50),  
    ciudad_prof VARCHAR(50),  
    titulo VARCHAR(50),  
    sueldo FLOAT  
);  
  
CREATE TABLE IF NOT EXISTS cursos (  
    cod_curso INT AUTO_INCREMENT PRIMARY KEY,  
    nombre_curso VARCHAR(50),  
    fecha_inicio DATE,  
    fecha_fin DATE,  
    num_horas INT,  
    dni_prof INT,  
  
    CONSTRAINT fk_curpro FOREIGN KEY (dni_prof) REFERENCES  
profesores(dni_prof)  
);  
  
CREATE TABLE IF NOT EXISTS alumnos (  
    dni_alu AUTO_INCREMENT PRIMARY KEY,  
    nombre_alu VARCHAR(50),  
    ciudad_alu VARCHAR(50),  
    fecha_naci DATE  
    cod_curso INT,  
  
    CONSTRAINT fk_alucur FOREIGN KEY (cod_curso) REFERENCES  
cursos(cod_curso)  
);
```

- 2. Modificar las tablas para introducir ahora sus claves ajenas. El que ya lo haya hecho pasará al punto 3.
- 3. Dibujar las tablas junto con las relaciones y, comentar cuál sería el orden a seguir a la hora de introducir los datos en las tablas.
El mejor orden es profesores, cursos, alumnos.
- 4. Se necesita añadir, en la tabla de profesores, una nueva columna para que contenga la edad del profesor, que será de tipo entero.

```
ALTER TABLE profesores
ADD COLUMN edad INT;
```

5. Obtener las siguientes consultas:

- 5.1. Todos los datos de la tabla cursos.

```
SELECT *
FROM cursos;
```

- 5.2. Dni, nombre y ciudad de los alumnos cuyo nombre contenga una 'i' y la ciudad contenga en la tercera letra una 'L'.

```
SELECT dni_alu, nombre_alu, ciudad_alu
FROM alumnos
WHERE nombre_alu LIKE '%i%'
AND ciudad LIKE '__l%';
```

- 5.3. Nombre y título de los profesores cuya edad esté entre 25 y 40 años (ambos inclusive), ordenados por el nombre de forma descendente.

```
SELECT nombre_prof, titulo
FROM profesores
WHERE edad BETWEEN 25 AND 40
ORDER BY DESC;
```

- 5.4. Dni y nombre de los alumnos que estén matriculados en un curso que dure como máximo una semana.

```
SELECT dni_alu, nombre_alu
FROM alumnos, cursos
WHERE fecha_fin - fecha_inicio < 7
AND alumnos.cod_curso = cursos.cod_curso;
```

- 5.5. Nombre del curso junto con el nombre del alumno que empiezan algún curso hoy.

```
SELECT nombre_curso, nombre_alu
FROM alumnos, cursos
WHERE fecha_inicio = CURRENT_DATE()
AND alumnos.cod_curso = cursos.cod_curso;
```

- 5.6. Nombre de los alumnos que están matriculados en el curso de "Japonés" impartido por el profesor "Pepe Chen".

```
SELECT nombre_alu
FROM cursos, alumnos
WHERE nombre_curso = 'Japones'
AND nombre_prof = 'Pepe Chen'
AND alumnos.cod_curso = cursos.cod_curso;
```

- 5.7. Nombre de los profesores (sin duplicados) de los cuales, aún no se conoce la fecha de inicio de los cursos que van a impartir.

```
SELECT DISTINCT nombre_prof
FROM profesores, cursos
WHERE fecha_inicio IS NULL;
AND profesores.dni_prof = cursos.dni_prof;
```

- 5.8. Nombre del profesor más mayor de la academia.

```
SELECT nombre_prof
FROM profesores
WHERE edad = (
    SELECT MAX(edad)
    FROM profesores;
);
```

- 5.9. Total de alumnos de la academia.

```
SELECT COUNT(*)
FROM alumnos;
```

- 5.10. La media del sueldo de los profesores que son de Málaga.

```
SELECT AVG(sueldo)
FROM profesores
WHERE ciudad = 'Málaga';
```

- 5.11. Los dni de los profesores, junto con el total del número de horas de cada curso.

```
SELECT dni_prof, SUM(numhoras)
FROM profesores, cursos
WHERE profesores.dni_prof = cursos.dni_prof;
```

- 5.12. Igual a la anterior, pero cuyo total de número de horas de los cursos supere las 1000 horas.

```
SELECT dni_prof, SUM(numhoras)
FROM profesores, cursos
WHERE numhoras > 1000
AND profesores.dni_prof = cursos.dni_prof;
```

- 5.13. Eliminar todos los cursos anteriores al 2010.

```
DELETE
FROM cursos
WHERE fecha_inicio < '2010-01-01'
```

- 5.14. Suponiendo que hemos exigido integridad referencial entre la relaciones profesores y cursos, sin establecer borrado en cascada. Elimina al profesor con dni 12.

```
DELETE
FROM profesor
WHERE dni_prof = 12;

DELETE
FROM profesor
WHERE dni_prof = 12;
```

- 5.15. La academia ha sido traspasada a otra, ya no necesita ninguna información sobre el alumnado, pues ella tiene los suyos y con otros datos, por lo tanto pasamos eliminar toda la información de los alumnos.

```
DROP TABLE alumnos;
```