**Se trata de:**

* Realizar borrados de registros sobre tablas de una base de datos.

**Objetivos:**

* Analizar el diseño físico y lógico de una base de datos.
* Comprobar los efectos de las modificaciones del valor de una PRIMARY KEY en una tabla principal sobre una tabla que la referencia por clave ajena.
* Comprobar los efectos de borrado de una fila en una tabla principal sobre una tabla que la referencia por clave ajena.
* Realizar eliminaciones con DELETE conforme a las especificaciones dadas.
* Usar combinaciones de tablas en DELETE.
* Utilizar subconsultas en DELETE.

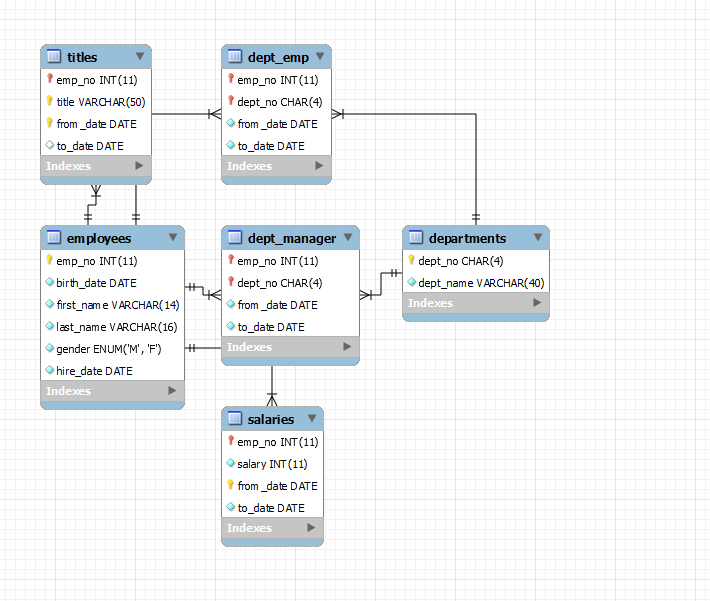
**Ejercicios**

En esta actividad vas a trabajar con una nueva base de datos. Se trata de la base de datos **employees** que tiene información sobre los empleados de una gran empresa.

Dado que queremos conservar los datos de esta base de datos, es aconsejable que realices las tareas dentro de una o varias transacciones que podrás cancelar al terminar el ejercicio 10.

En esta actividad, además de realizar fundamentalmente borrados de filas o registros, realizarás otras tareas para analizar el comportamiento de borrados y modificaciones de datos.

1.- Realiza la importación de la base de datos **employees.** Obtén el grafo relacional de esta base de datos e interpreta:



* Que información contiene cada tabla.
* Que columna o columnas forman la PRIMARY KEY de cada tabla.
* Que relaciones de clave ajena están establecidas.
* Que restricciones de borrado y modificación hay establecidas para las relaciones de clave ajena.

2.- Obtén cuantos salarios tiene registrados **Kyoichi Maliniak.**

select count(salary) from salaries inner join employees on employees.emp\_no=salaries.emp\_no where first\_name="Kyoichi" and last\_name="Maliniak";

Elimina todos los salarios de **Kyoichi Maliniak.**

delete salaries from salaries inner join employees on employees.emp\_no=salaries.emp\_no where first\_name="Kyoichi" and last\_name="Maliniak";

3.- Obtén los salarios del empleado más joven del departamento de código **d005.**

Select \* from salaries where emp\_no=(select emp\_no from employees inner join dept\_emp using(emp\_no) where to\_date="9999-01-01" and dept\_no="d005" order by birth\_date desc limit 1);

Elimina los salarios del empleado más joven del departamento **de código d005.**

**delete from salaries where emp\_no=(select emp\_no from employees inner join dept\_emp using(emp\_no) where to\_date="9999-01-01" and dept\_no="d005" order by birth\_date desc limit 1);**

4.- Modifica el valor de la clave primaria del departamento de **Development** para que contenga **d015.**

update departments set dept\_no="d015" where dept\_name="Development";

¿Qué ocurre y por qué?

Modifica el nombre del departamento de código **d005** para que contenga **Desarrollo.**

update departments set dept\_name="Desarrollo" where dept\_no="d005";

5.- Elimina todos los empleados que en la fecha actual pertenecen al departamento **Quality Management.**

delete employees from employees where emp\_no in(Select emp\_no from dept\_emp inner join departments using(dept\_no) where dept\_name="Quality Management" and to\_date="9999-01-01");

6.- Elimina los salarios de cada empleado que actualmente trabaja en la categoría **Technique Leader**(tabla **Titles)**

**delete salaries from salaries where emp\_no in (Select emp\_no from titles where title="Technique Leader" and to\_date="9999-01-01");**

7.- Elimina todos los salarios cuya fecha final del periodo sea anterior a hace 37 años desde la fecha actual.

delete salaries from salaries where to\_date < subdate(curdate(),Interval 37 year);

8.- Elimina los salarios del empleado que era director del departamento **d003** el 1 de enero de 1990.

delete salaries from salaries where emp\_no=(select emp\_no from dept\_manager where dept\_no="d003" and from\_date<"1990-01-01" and to\_date>"1990-01-01");

9.- Elimina los empleados del departamento **d003** cuyo salario actual sea superior a la media de los salarios actuales en ese departamento.

delete employees from employees inner join salaries using(emp\_no) inner join dept\_emp using(emp\_no) where dept\_no="d003" and salaries.to\_date="9999-01-01" and

salary > (select avg(salary) from salaries inner join dept\_emp using(emp\_no) where dept\_no="d003" and salaries.to\_date="9999-01-01");

10.- Elimina los empleados nacidos antes de 1980 y que pertenezcan al mismo departamento que el de **Urs Lung.**

**delete employees from employees inner join dept\_emp using(emp\_no) where birth\_date<"1980-01-01" and**

**dept\_no=(select dept\_no from dept\_emp inner join employees using(emp\_no) where first\_name="Urs" and last\_name="Lung");**

11.- Establece que las fechas de nacimiento (birth\_date) y de alta en la empresa (hire\_date), de todos los empleados, sean 20 años superiores a las que tienen actualmente.

12.- En la tabla **dept\_emp** establece que todas las fechas, excepto las que contienen año 9999, sean 20 años superiores a las actuales.

13.- En la tabla **dept\_manager** establece que en todas las fechas haya fechas 20 años superiores a las actuales

14.- En la tabla **salaries** establece que en todas las fechas haya fechas 20 años superiores a las actuales, excepto para las que contengan 9999 en el año.

15.- En la tabla **titles** establece que todas las fechas haya una fecha 20 años superiores a las actuales, excepto las que contengan 9999 en el año.