

## Laboratorio #2 – DDL y DML:

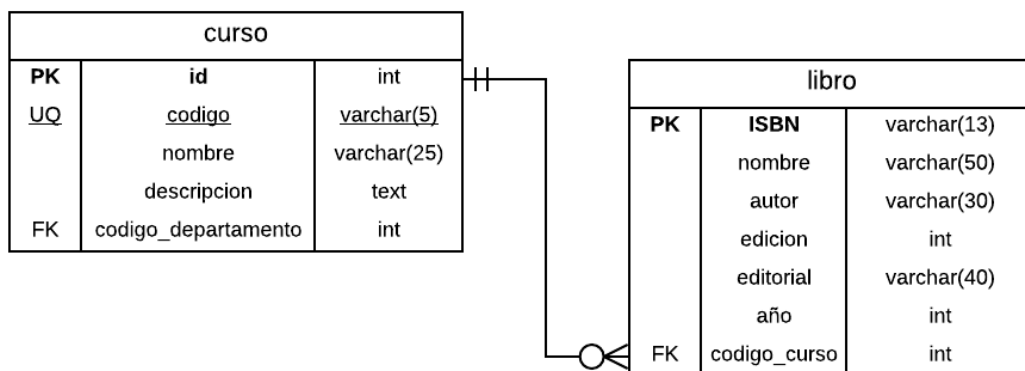
### Parte #1:

Cree el código DDL necesario para las siguientes tablas, tome en cuenta las siguientes restricciones para crear los constraints necesarios para integridad de entidades, integridad de dominio, integridad de referencias.

- El código del curso está formado por dos caracteres alfa numéricos en mayúsculas, un guion medio y dos números los cuales indican el número de semestre.
- El código de departamento es una llave foránea que al eliminar un código de departamento, el curso debe colocar su llave foránea en nulo, y al actualizar el código del departamento debe actualizarse la llave foránea.
- El código de ISBN son 13 números, de los cuales ninguno de los dos primeros pueden ser cero o uno.
- El año del libro no puede ser menor del año 2000 y mayor del año actual.
- La edición del libro debe ser mayor a la 3era y menor a las 6ta.
- Si se elimina un curso o actualiza un curso, se debe realizar la misma acción con el libro.

Posteriormente a esto inserte 47 cursos y 150 libros con data generada, recuerde que no es necesario que los nombres necesariamente tengan sentido, sin embargo asegúrese de que los inserts cumplan con los campos que tiene restricciones asociadas.

Para entregar este ejercicio, deberá mostrar el código SQL del DDL con sus respectivas restricciones y el código de todos los inserts.



## Parte #2:

Corra el siguiente código de MySQL, posteriormente realice los queries que se le solicitan en la parte #3:

```
CREATE DATABASE RecursosHumanos;
```

```
USE RecursosHumanos;
```

```
CREATE TABLE DEPARTAMENTO (  
    ID INT NOT NULL,  
    NOMBRE VARCHAR(50),  
    LOCALIDAD VARCHAR(50),  
    CONSTRAINT DEPARTAMENTOS_PK  
    PRIMARY KEY (ID)  
);
```

```
CREATE TABLE RANGO_SUELDO (  
    GRADO INT PRIMARY KEY ,  
    SUELDO_MIN DECIMAL(9,2),  
    SUELDO_MAX DECIMAL(9,2)  
);
```

```
CREATE TABLE EMPLEADO (  
    ID INT NOT NULL,  
    NOMBRE VARCHAR(30),  
    APELLIDOS VARCHAR(60),  
    TELEFONO VARCHAR(25),  
    EMAIL VARCHAR(50),  
    CARGO VARCHAR(20),  
    JEFE_ID INT,  
    FECHA_ALTA DATE,  
    SUELDO DECIMAL(9,2),  
    COMISION DECIMAL(9,2),  
    DEPARTAMENTO_ID INT NOT NULL,  
    CONSTRAINT EMPLEADOS_PK  
    PRIMARY KEY (ID),  
    CONSTRAINT DEPARTAMENTO_ID_FK  
    FOREIGN KEY (DEPARTAMENTO_ID) REFERENCES DEPARTAMENTO(ID),  
    CONSTRAINT EMAIL_CHK  
    CHECK ( EMAIL REGEXP '^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9-  
    .]+$',),  
    CONSTRAINT TELEFONO_CHK  
    CHECK ( TELEFONO REGEXP '^[+]*[(]{0,1}[0-9]{1,4}[)]{0,1}[-  
    [[:space:]]\./0-9]*$' )  
);
```

## DML. Inserción de datos:

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, EMAIL, TELEFONO, CARGO, JEFE_ID,  
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)  
VALUES (6, 'INES', 'FERNANDEZ SANCHEZ', 'fsanchez!@gmail.com', '+50255552612a',  
'PRESIDENTE', NULL, '1991-11-17', 5000, NULL, 1);
```

```

INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (4, 'ESTHER', 'PEÑA MATA', '+34937377761', 'epena@gmail.com', 'DIRECTOR',
6, str_to_date('01/05/1991', '%d/%m/%Y'), 2850, NULL, 3);

INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (5, 'DANIEL', 'LOPEZ LEON', '+34937377780', 'dlopez@gmail.com', 'DIRECTOR',
6, str_to_date('09/06/1991', '%d/%m/%Y'), 2450, NULL, 1);

INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (1, 'CARLOS', 'ALONSO GARCIA',
'34937377770', 'carlos.alonso@gmail.com', 'COMERCIAL', 4,
str_to_date('20/02/1991', '%d/%m/%Y'), 1600, 30, 3);

INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (2, 'ANDRES', 'GOMEZ SEGUNDO', '+ (34) 937-376-525',
'andres.gomez@gmail.com', 'COMERCIAL', 4, str_to_date('22/02/1991', '%d/%m/%Y'),
1250, 50, 3);

INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (3, 'ANA', 'HERRANZ CIFUENTES', '+34937376523', 'herranz.ana@gmail.com',
'COMERCIAL', 4, str_to_date('28/09/1991', '%d/%m/%Y'), 1250, 140, 3);

INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (7, 'TOMAS', 'CALLEJA ARCAZ', '+34937376541', 'tomas.calleja@gmail.com',
'COMERCIAL', 4, str_to_date('08/09/1991', '%d/%m/%Y'), 1500, 0, 3);

COMMIT ;

```

```

INSERT INTO DEPARTAMENTO (ID, NOMBRE, LOCALIDAD)
VALUES (1, 'INFORMATICA', 'BILBAO');
INSERT INTO DEPARTAMENTO (ID, NOMBRE, LOCALIDAD)
VALUES (2, 'INVESTIGACION', 'MADRID');
INSERT INTO DEPARTAMENTO (ID, NOMBRE, LOCALIDAD)
VALUES (3, 'VENTAS', 'BARCELONA');
INSERT INTO DEPARTAMENTO (ID, NOMBRE, LOCALIDAD)
VALUES (4, 'PRODUCCION', 'VALENCIA');
COMMIT ;

```

```

INSERT INTO RANGO_SUELDO (GRADO, SUELDO_MIN, SUELDO_MAX)
VALUES (1, 700, 1200);
INSERT INTO RANGO_SUELDO (GRADO, SUELDO_MIN, SUELDO_MAX)
VALUES (2, 1201, 1400);
INSERT INTO RANGO_SUELDO (GRADO, SUELDO_MIN, SUELDO_MAX)
VALUES (3, 1401, 2000);
INSERT INTO RANGO_SUELDO (GRADO, SUELDO_MIN, SUELDO_MAX)

```

```
VALUES (4, 2001, 3000);
INSERT INTO RANGO_SUELDO (GRADO, SUELDO_MIN, SUELDO_MAX)
VALUES (5, 3001, 9999);
COMMIT ;
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, EMAIL, TELEFONO, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (6, 'INES', 'FERNANDEZ SANCHEZ', 'fsanchez!@gmail.com', '+50255552612a',
'PRESIDENTE', NULL, '1991-11-17', 5000, NULL, 1);
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (4, 'ESTHER', 'PEÑA MATA', '+34937377761', 'epena@gmail.com', 'DIRECTOR',
6, str_to_date('01/05/1991', '%d/%m/%Y'), 2850, NULL, 3);
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (5, 'DANIEL', 'LOPEZ LEON', '+34937377780', 'dlopez@gmail.com', 'DIRECTOR',
6, str_to_date('09/06/1991', '%d/%m/%Y'), 2450, NULL, 1);
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (1, 'CARLOS', 'ALONSO GARCIA',
'34937377770', 'carlos.alonso@gmail.com', 'COMERCIAL', 4,
str_to_date('20/02/1991', '%d/%m/%Y'), 1600, 30, 3);
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (2, 'ANDRES', 'GOMEZ SEGUNDO', '+ (34) 937-376-525',
'andres.gomez@gmail.com', 'COMERCIAL', 4, str_to_date('22/02/1991', '%d/%m/%Y'),
1250, 50, 3);
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (3, 'ANA', 'HERRANZ CIFUENTES', '+34937376523', 'herranz.ana@gmail.com',
'COMERCIAL', 4, str_to_date('28/09/1991', '%d/%m/%Y'), 1250, 140, 3);
```

```
INSERT INTO EMPLEADO (ID, NOMBRE, APELLIDOS, TELEFONO, EMAIL, CARGO, JEFE_ID,
FECHA_ALTA, SUELDO, COMISION, DEPARTAMENTO_ID)
VALUES (7, 'TOMAS', 'CALLEJA ARCAZ', '+34937376541', 'tomas.calleja@gmail.com',
'COMERCIAL', 4, str_to_date('08/09/1991', '%d/%m/%Y'), 1500, 0, 3);
```

```
COMMIT ;
```

### **Parte #3:**

**Genere el código SQL para resolver las preguntas que se le plantean, en su entrega debe incluir un archivo .SQL colocando como comentario la pregunta que está resolviendo con el código:**

A continuación, escriba las siguientes consultas, de ser posible de más de una forma.

1. Considerando las reglas de normalización, ¿Qué mejoras podría realizar en las tablas?
2. ¿Qué registro de empleado falla y por qué? ¿Cómo se corrige?
3. Obtener los apellidos del empleado
4. Obtener los apellidos del empleado sin repetición
5. Obtener todos los datos de los empleados apellidados López
6. Obtener todos los datos del empleado con apellido López o Peña
7. Obtener los datos del empleado que trabaja en el departamento de informática.
8. Obtener los datos del empleado cuyo nombre comience por l
9. Obtener el nombre del departamento y el número de empleados que tiene.
10. Consultar los diferentes cargos desempeñados en la empresa.
11. Listar los empleados como sigue: LOPEZ PEREZ, CARLOS: es PRESIDENTE
12. Mostrar el nombre, apellidos, sueldo y comisión del empleado que tiene jefe, y su comisión es superior a 150 €.
13. Obtener los empleados que no ejercen de comercial y cuyos nombres contienen la letra N.
14. De los empleados que tienen comisión, obtener el nombre, sueldo y comisión, ordenados por sueldo de mayor a menor.
15. Ordenar a los empleados por sus apellidos y nombres.
16. Obtener el nombre, sueldo anual y fecha de alta de los empleados que no tienen comisión, ordenados por fecha de alta en la empresa.
17. Mostrar el nombre y sueldo de los empleados incrementados en un 10% y ordenados por nombre y apellidos.
18. Mostrar el nombre y fecha de alta de cada empleado, ordenado por fecha de entrada como sigue: Fecha de entrada (como título) Ejemplo: 'Carlos Alonso García empezó a trabajar el 20 de febrero de 1991'
19. Obtener el nombre de los comerciales, la comisión, el sueldo mensual y el sueldo incrementado como sigue: si la comisión es menor que 50 €, aumentar un 25%; y si es igual o superior, aumentar un 12%.
20. Mostrar la localidad, nombre y sueldo del empleado cuyo grado es 2 o 3.
21. Mostrar el nombre de los departamentos cuyos empleados tienen comisión o su sueldo anual es superior a 18.000 €.
22. Calcular el número de empleados del departamento de VENTAS.
23. Calcular la comisión media de los empleados, excluyendo al presidente y suponiendo que todos los empleados cobran al menos una comisión de 0 €.
24. Calcular el sueldo máximo de los empleados de cada departamento siempre que el mínimo sueldo del departamento sea superior a 780 €.
25. Mostrar el nombre y fecha de entrada de todos los empleados que trabajan en el mismo departamento que ESTHER.
26. Calcular el número de empleados que están en BILBAO