

## Ejercicio resumen Definición de datos en SQL

Se desea tener una base de datos que almacene la **información sobre los empleados** de una empresa, los **departamentos en los que trabajan** y los **estudios de que disponen**. **Guardaremos** el **historial laboral y salarial de todos los empleados**. Para ello contamos con las siguientes tablas:

### EMPLEADOS

| Column Name | Data Type    |
|-------------|--------------|
| -----       | -----        |
| DNI         | NUMBER(8)    |
| NOMBRE      | VARCHAR2(10) |
| APELLIDO1   | VARCHAR2(15) |
| APELLIDO2   | VARCHAR2(15) |
| DIRECC1     | VARCHAR2(25) |
| DIRECC2     | VARCHAR2(20) |
| CIUDAD      | VARCHAR2(20) |
| MUNICIPIO   | VARCHAR2(20) |
| COD_POSTAL  | VARCHAR2(5)  |
| SEXO        | CHAR(1)      |
| FECHA_NAC   | DATE         |

### DEPARTAMENTOS

| Column Name | Data Type    |
|-------------|--------------|
| -----       | -----        |
| DPTO_COD    | NUMBER(5)    |
| NOMBRE_DPTO | VARCHAR2(30) |
| JEFE        | NUMBER(5)    |
| PRESUPUESTO | INTEGER      |
| PRES_ACTUAL | INTEGER      |

### ESTUDIOS

| Column Name  | Data Type    |
|--------------|--------------|
| -----        | -----        |
| EMPLEADO_DNI | NUMBER(8)    |
| UNIVERSIDAD  | NUMBER(5)    |
| AÑO          | SMALLINT     |
| GRADO        | VARCHAR2(3)  |
| ESPECIALIDAD | VARCHAR2(20) |

### HISTORIAL\_LABORAL

| Column Name    | Data Type |
|----------------|-----------|
| -----          | -----     |
| EMPLEADO_DNI   | NUMBER(8) |
| TRAB_COD       | NUMBER(5) |
| FECHA_INICIO   | DATE      |
| FECHA_FIN      | DATE      |
| DPTO_COD       | NUMBER(5) |
| SUPERVISOR_DNI | NUMBER(8) |

### UNIVERSIDADES

| Column Name | Data Type    |
|-------------|--------------|
| -----       | -----        |
| UNIV_COD    | NUMBER(5)    |
| NOMBRE_UNIV | VARCHAR2(25) |
| CIUDAD      | VARCHAR2(20) |
| MUNICIPIO   | VARCHAR2(2)  |
| COD_POSTAL  | VARCHAR2(5)  |

### HISTORIAL\_SALARIAL

| Column Name    | Data Type |
|----------------|-----------|
| -----          | -----     |
| EMPLEADO_DNI   | NUMBER(8) |
| SALARIO        | INTEGER   |
| FECHA_COMIENZO | DATE      |
| FECHA_FIN      | DATE      |

### TRABAJOS

| Column Name | Data Type    |
|-------------|--------------|
| -----       | -----        |
| TRABAJO_COD | NUMBER(5)    |
| NOMBRE TRAB | VARCHAR2(20) |
| SALARIO_MIN | NUMBER(2)    |
| SALARIO_MAX | NUMBER(2)    |

## Controlar las siguientes restricciones:

- Los siguientes **atributos son obligatorios**: Nombre (en todas las tablas), Apellido1 de cada Empleado, Presupuesto en Departamentos, Salario de un Empleado y Salario Mínimo y Máximo de cada tipo de trabajo.
- El **atributo Sexo** en **empleados sólo puede tomar los valores H para hombre y M para mujer**.
- Dos departamentos no se llaman igual. Dos trabajos tampoco**.
- Las **fechas de comienzo** y de fin en los dos **historiales** deben **respetar el orden cronológico**, teniendo en cuenta que **la fecha de finalización puede ser NULL**, indicando que **ese es el valor actual de su salario**.
- Cada **empleado tiene un solo salario en cada momento**. También, cada **empleado tendrá asignado un solo trabajo en cada momento**.
- Se ha de **mantener la regla de integridad de referencia** y **pensar una clave primaria para cada tabla**.

## Realizar las siguientes operaciones:

1. Añada una **restricción** que **obligue** a que las personas de **sexo masculino** hayan de tener el campo **Fecha de Nacimiento NOT NULL**.

```
ALTER TABLE empleados add( CONSTRAINT sexo_ck CHECK( sexo='V' and DEFAULT 5, CHECK(valoracion BETWEEN 1 AND 10));
```

```
ALTER TABLE empleados add
(
constraint sexo_ck check ((sexo = 'V' and fecha_nac IS NOT NULL) or sexo = 'M')
);
```

2. Añada un nuevo atributo **Valoración** en la **tabla de Empleados** que indica de **1 a 10** la valoración que obtuvo el empleado en su entrevista de trabajo al iniciar su andadura en la empresa. Ponga el **valor por defecto 5** para ese campo.

```
ALTER TABLE empleados add(valoracion NUMBER(2) DEFAULT 5, CHECK(valoracion BETWEEN 1 AND 10));
```

```
ALTER TABLE empleados add
(
valoracion number(2) default 5,
check (valoracion between 1 and 10)
);
```

3. Elimine la restricción de que el atributo **Nombre** de la **tabla empleado no puede ser nulo**. Otra forma de decir que puede ser nulo.

```
ALTER TABLE empleados MODIFY(nombre VARCHAR2(10));
```

```
ALTER TABLE empleados modify
(
nombre varchar2(10)
);
```

4. Modificar el tipo de datos de **DIRECC1** de la **tabla Empleados** a **cadena de caracteres de 40** como máximo. ¿Podría modificar el tipo de datos del atributo **Código** de empleado a **VARCHAR(40)**? ¿Y la **fecha de nacimiento convertirla a tipo cadena**?

```
ALTER TABLE empleados MODIFY(DIRECC1 CHAR(40));
```

```
ALTER TABLE empleados modify
(
direcc1 char(40)
);
```

5. Cambiar la **clave primaria** de Empleados al nombre y los dos apellidos.

```
ALTER TABLE empleados drop constraint SYS_C007070 cascade;
```

AL HACER ESTO PERDEMOS TODAS LAS REFERENCES QUE HACÍAN REFERENCIA .

Cambiamos la clave primaria ahora:

```
ALTER TABLE empleados add  
(  
constraint pk_emple PRIMARY KEY (nombre, apellido1,apellido2)  
);
```

6. Borrar todas las tablas. ¿Hay que tener en cuenta las claves foráneas a la hora de borrar las tablas?

SI, HAY QUE **BORRARLAS EN EL MISMO ORDEN DE CREACIÓN**, ES DECIR, **DESDE LA ULTIMA TABLA QUE CREAMOS HASTA LA PRIMERA.**

```
DROP TABLE empleados;  
drop table historial_laboral;  
drop table historial_salarial;  
drop table trabajos;  
drop table universidades;  
drop table estudios;  
drop table departamentos;
```

NOTAS sobre lo revisado:

## SQL Server | 06 - Alterar o modificar una tabla Alter Table

Sobre el Alter Table desde el [ORAU.440](#)

Sobre alter table [comandos](#)

Caso clave primaria y alter [ELIMINAR](#) PRIMERO