

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	VARCHAR(10)
APELLIDO1	VARCHAR(15)
APELLIDO2	VARCHAR(15)
DIRECC1	VARCHAR(25)
DIRECC2	VARCHAR(20)
CIUDAD	VARCHAR(20)
PROVINCIA	VARCHAR(20)
COD_POSTAL	VARCHAR(5)
SEXO	VARCHAR(1)
FECHA_NAC	DATE

DEPARTAMENTOS

Column Name	Data Type
-----	-----
DPTO_COD	NUMBER(5)
NOMBRE_DPTO	VARCHAR(30)
DPTO_PADRE	NUMBER(5)
PRESUPUESTO	NUMBER
PRES_ACTUAL	NUMBER

ESTUDIOS

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

HISTORIAL_LABORAL

Column Name	Data Type
-----	-----
EMPLEADO_DNI	NUMBER(8)
TRABAJO_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	VARCHAR(25)
CIUDAD	VARCHAR(20)
MUNICIPIO	VARCHAR(2)
COD_POSTAL	VARCHAR(5)

HISTORIAL_SALARIAL

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

TRABAJOS

Column Name	Data Type
-----	-----
TRABAJO_COD	NUMBER(5)
NOMBRE_TRAB	VARCHAR(20)
SALARIO_MIN	NUMBER(2)
SALARIO_MAX	NUMBER(2)

- Los campos marcados en amarillo son las claves primarias de cada tabla
- Los campos marcados con flecha roja son las claves foráneas.

- 1-Crea una nueva base de datos llamada universidad
- 2-Genara el script de creación de esta base de datos
- 3-Una vez creada utilizando MySQL Workbench obtén el diagrama entidad relación.

Ejercicio 2 Gestión pubs

Disponemos de la siguiente Base de Datos para gestionar la información de los pubs de una determinada provincia.

PUB	TITULAR	EMPLEADO
#COD_PUB	#DNI_TITULAR	#DNI_EMPLEADO
NOMBRE	NOMBRE	NOMBRE
LICENCIA_FISCAL	DOMICILIO	DOMICILIO
DOMICILIO	COD_PUB	
FECHA_APERTURA		
HORARIO		
COD_LOCALIDAD		

EXISTENCIAS	LOCALIDAD	PUB_EMPLEADO
#COD_ARTICULO	#COD_LOCALIDAD	#COD_PUB
NOMBRE	NOMBRE	#DNI_EMPLEADO
CANTIDAD		#FUNCION
PRECIO		
COD_PUB		

Se pide escribir los comandos SQL que permitan la creación de las tablas anteriores teniendo en cuenta las siguientes restricciones:

- Todos los valores son de tipo carácter excepto los campos FECHA_APERTURA (fecha) , CANTIDAD, PRECIO y COD_LOCALIDAD (numéricos).
- Los únicos campos que no son obligatorios son los campos DOMICILIO.
- Los valores del campo horario sólo pueden ser HOR1, HOR2 y HOR3.
- No es posible dar de alta EXISTENCIAS a precio 0.
- El campo función de la tabla PUB_EMPLEADO sólo puede tener los valores CAMARERO, SEGURIDAD, LIMPIEZA.
- Se ha de mantener la integridad referencial entre las tablas.
- Las claves primarias vienen marcadas con el símbolo #.

Ejercicio 3 Venta de entradas

La siguiente base de datos está pensada para almacenar la información necesaria para gestionar la venta automática de entradas para diferentes espectáculos desde múltiples puntos de venta, como pueden ser oficinas bancarias, terminales tipo Servicaixa, o las mismas taquillas de teatros u otros recintos.

- ESPECTACULOS (COD_ESPECTACULO, NOMBRE, TIPO, FECHA_INICIAL, FECHA_FINAL, INTERPRETE, COD_RECINTO)
- PRECIOS_ESPECTACULOS (COD_ESPECTACULO, COD_RECINTO, ZONA, PRECIO)
- RECINTOS (COD_RECINTO, NOMBRE, DIRECCION, CIUDAD, TELEFONO, HORARIO)
- ZONAS_RECINTOS (COD_RECINTO, ZONA, CAPACIDAD)
- ASIENTOS (COD_RECINTO, ZONA, FILA, NUMERO)
- REPRESENTACIONES (COD_ESPECTACULO, FECHA, HORA)
- ENTRADAS (COD_ESPECTACULO, FECHA, HORA, COD_RECINTO, FILA, NUMERO, ZONA, DNI_CLIENTE)
- ESPECTADORES (DNI_CLIENTE, NOMBRE, DIRECCION, TELEFONO, CIUDAD, NTARJETA)

Se pide:

1. Establecer las claves primarias de cada una de las tablas y las restricciones de integridad referencial existentes entre las mismas.
2. Crear las sentencias SQL que nos permiten crear las tablas anteriores y sus restricciones.