

Bases de Datos II
Serie Ejercicios Prácticos 4
Bases de Datos Objeto Relacionales

Para estos ejercicios, se debe de especificar aquellas sentencias en Sql3, necesarias para cada caso, como siempre les indicamos, las resoluciones que se adjuntan son las sugeridas, pero pueden variar de acuerdo al manejador de bases de datos con el cual uno opere, pudiendo incorporar alternativas que crean necesarias.

1)

a)

```
Create table Libros (
    Cod_libro integer generated always as identity (
        start with 1
        increment by 1
        minvalue 1
        maxvalue 5000
        no cycle),
    Titulo  varchar(40),
    Autor   varchar(35),
    Año     char(4),
    Precio  decimal(10,2)
);
```

b)

```
Create table Editorial (
    Cod_libro      integer
    Titulo         varchar(50),
    Descripcion   clob(40k),
    Autor          varchar(40),
    Foto_portada  blob(2m),
    Videopresen   blob(1G)
);
```

2)

```
Create type persona as (
    Dni           char(8),
    Apeynom      varchar2(50),
    Direccion    varchar2(40),
    Ciudad        char(25),
    Fechanac     date) not final;
```

```
Create type empresa as (
    Empleado      persona,
    Mail          varchar2(50),
    Codpostal    integer) not final;
```

(Suponemos que hay un nivel de jerarquía inferior a Empresa, por ejemplo Departamentos – Sección)

```
Create type jefe as (
    Director      persona,
    Mail          varchar2(50),
    Area          char(2),
    Sueldo        decimal(8,2)) final;
```

```
Create table empleados of empresa;
```

3)

a)

```
Create table Universidad (
    Nombre        varchar2(100),
    Rector        varchar2(50),
    Direccion    varchar(40),
    Facultades   varchar(15) array[9],
    Codpostal    integer,
    Sitio_web    varchar2(80));
```

Variante: para definir facultad de tipo multiconjunto.

```
Facultades      varchar(15)      multiset
```

b)

```
Insert into    universidad (nombre, rector, direccion, facultades, codpostal, sitio_web)
Values        ('Universidad Nacional del Nordeste',
'Maria Veirave',
'25 de Mayo 868',
['Medicina', 'Humanidades', 'Exactas', 'Veterinaria', 'Económicas',
'Agrarias', 'Arquitectura', 'Derecho', 'Odontología'],
3400, 'www.unne.edu.ar');
```

Variante: para incorporar valores en el atributo facultad de tipo multiconjunto:

```
Insert into    universidad (nombre, rector, direccion, facultades, codpostal, sitio_web)
Values        ('Universidad Nacional del Nordeste',
'Maria Veirave',
'25 de Mayo 868',
'Multiset['Medicina', 'Humanidades', 'Exactas', 'Veterinaria', 'Económicas',
'Agrarias', 'Arquitectura', 'Derecho', 'Odontología'],
3400, 'www.unne.edu.ar');
```

4)

```
Create table Banco (
  Identificación row (
    Bcra      integer,
    Cuit      char(13)),
  Razon_social row (
    Nombre_comer      varchar2(60),
    Tipo_emp          char(2),
    Cond_Afip         char(4)),
  Presidente row (
    Nombre            varchar2(40),
    Apellido          varchar2(30),
    Dni               char(8),
    Mail              varchar2(50)),
  Dirección row (
    Calle             varchar2(40),
    Altura            integer,
    Ciudad            varchar(30),
    Cod_postal        integer),
  Telefono row (
    Prefijo           integer,
    Num_te            integer )
)
```

5)

```
Create type empleado as (
  Dni      char(8),
  Apeynom  varchar(50),
  method Antiguedad_laboral() returns integer,
  Direccion varchar(40),
  Cargo    varchar(15),
  method Sueldo() returns decimal (10,2);
```

```
Create method Antiguedad_laboral(...lista de argumentos...) for empleado
```

```
Begin
  Return(año_actual - año_ingreso);
End;
```

```
Create method sueldo(...lista de argumentos...) for empleado
```

```
Begin
  Return((básico + titulo + escolaridad) - aporte_jubilatorio);
End;
```

```
)
```

6)

```
Create type producto as
(
  Codproducto      char(12),
  Denominacion    varchar(70),
```

```

Stock_actual      integer,
Stock_minimo      integer,
Precio_fabrica    decimal (8,2)
method Precio_consumidor() returns  decimal(11,2)

);

Create method Precio_consumidor(...lista de argumentos...) for producto
Begin
    Return(Precio_fabrica + (Precio_fabrica * 0,15));
End;

```

Otra forma es variar la forma de calcular el 15% sobre el precio de fábrica, podría ser:

```

Create method Precio_consumidor(...lista de argumentos...) for producto
Begin
    Return(Precio_fabrica * 1,15);
End;

```

En cuanto a los métodos, queda indicado el cálculo que debería de devolver el mismo, puede variar de acuerdo a la forma de implementar de cada alumno, no tenemos un criterio determinado de solución al respecto.

7)

```

Create type direccion_postal as
(
    calle          varchar(50),
    numero         varchar(70),
    provincia      varchar(20),
    cp              integer
);

Create table personas (
    dni             char(8) primary key,
    nombres         varchar2(40),
    apellidos       varchar2(40),
    fecha_nacimiento date,
    telefonos       text[],
    direccion       direccion_postal
);

```

Así, a modo de ejemplo podemos considerar una inserción en PostgreSQL:

```

Insert into personas
    values ('35765091', 'Estela', 'Maciel', '1991-08-21', '{"3794117490", "3624701534"}',
            row('Pellegrini', '1591', 'Corrientes', 3400));

```

Y para consultar sería como sigue:

```
Select direccion.cp from personas where nombre = 'Estela' and apellidos= 'Maciel';
```