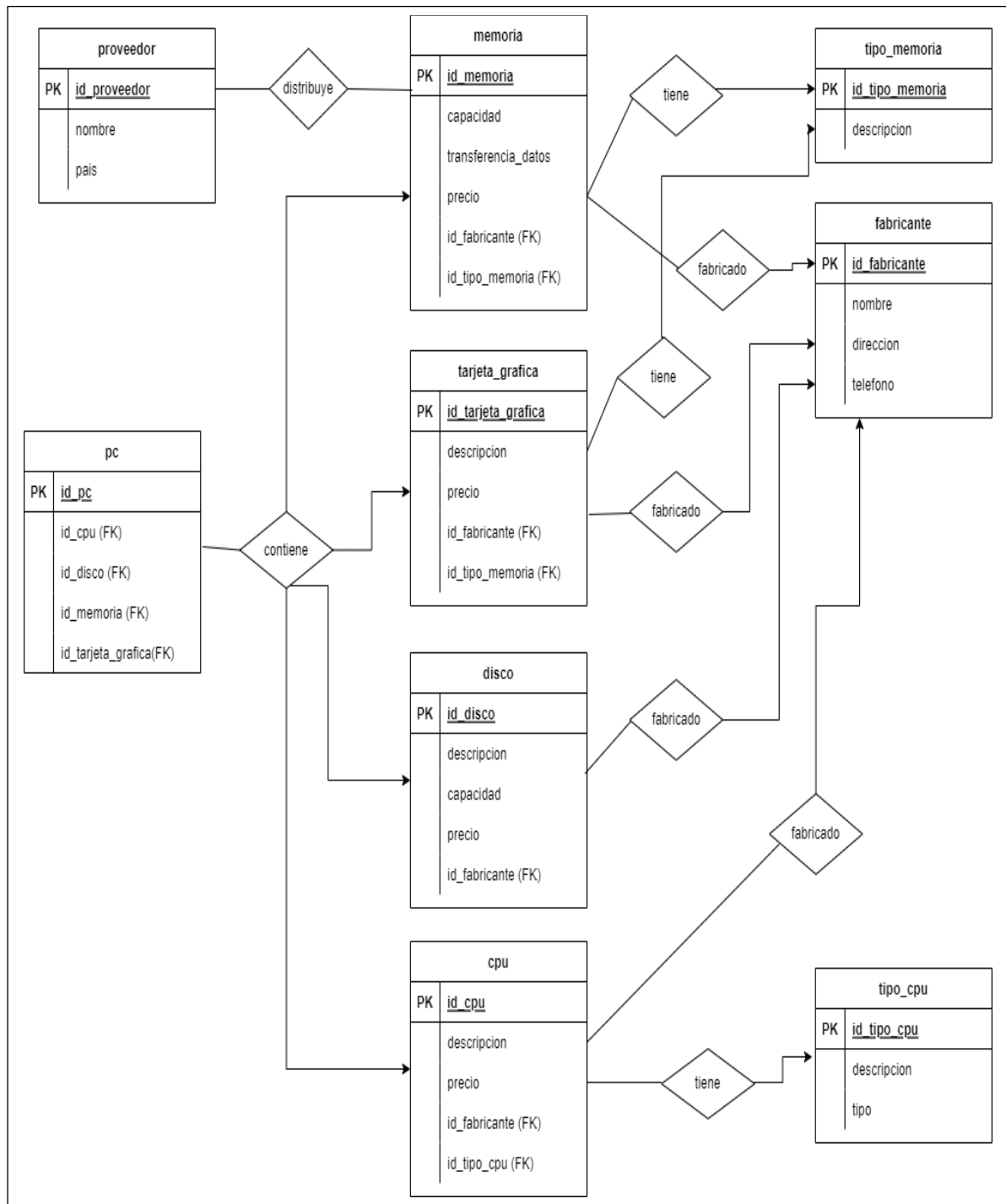




El fichero 'pcs\_esquema.sql' ha sido preparado para la creación de las tablas del modelo relacional correspondiente al modelo entidad-relación de la Figura 1. La popularización de la base de datos denominada 'pcs', se deberá realizar con el fichero 'pcs\_datos.sql' (i '<path al fichero>\\<nombre\_fichero>').



**Figura 1.** Modelo entidad-relación de la base de datos pcs.



En primer lugar, vamos a explicar el concepto de dominio (tipo de dato) que se podrían crear, tal y como se encuentra especificado en la Tabla 1. La información sobre dominios puede ser consultada en <https://www.postgresql.org/docs/9.5/sql-createdomain.html> (última visita 15/04/2022).

Tabla 1. Dominio.

| DOMINIO   |  |
|---|--|
| Es un tipo de dato al que se le puede atribuir restricciones opcionales sobre el conjunto de valores que se le podrían asignar. | <p><b>Sintaxis:</b></p> <p><b>Create domain</b> nombre [as] tipo_dato<br/>[collate collation]<br/>[default expresion]<br/>[constraint [...]]</p> <p>donde <b>constraint</b> representa:</p> <p>[<b>constraint</b> constraint_nombre]<br/>{ <b>not null</b>   <b>null</b>   <b>check</b> (expresion) }</p> <p><b>Ejemplos:</b></p> <p><b>1. Email:</b><br/><b>create domain</b> domain_email <b>as</b> text<br/><b>check</b> (<br/>  <b>value</b> ~ '^\\w+@[a-zA-Z_]+?\\.[a-zA-Z]{2,3}\$'<br/>);</p> <p>\\w es un metacaracter usado para encontrar un carácter perteneciente a a-z, A-Z, 0-9, incluyendo el caracter '_'.</p> <p><b>2. Postcode:</b><br/><b>create domain</b> domain_post_code <b>as</b> text<br/><b>check</b> (<br/>  <b>value</b> ~ '^\\d{5}\$'<br/>);</p> |



A continuación, se presenta una breve explicación sobre los tipos de restricciones que podrían ser impuestas a los datos que vayan a ser almacenados en la base de datos (Tabla 2).

**Tabla 2. Restricciones.**

|                     |  |
|---------------------|--|
| <p><b>check</b></p> | <p>1) <b>Restricciones de columna:</b> un valor en una cierta columna debe satisfacer una expresión, y se especifica después del tipo de dato. Por ejemplo:</p> <pre>create table productos( ....     precio numeric <b>check</b> (precio &gt; 0));</pre> <p>A las restricciones de columna se les puede asignar un nombre, como por ejemplo:</p> <pre>create table productos(...     precio numeric <b>constraint</b> <b>precio_positivo check</b> (precio &gt; 0));</pre> <p>2) <b>Restricciones de tabla:</b> englobaría más de una columna, como por ejemplo:</p> <pre>create table productos ( ....     precio numeric <b>check</b> (precio &gt; 0), -&gt; <b>restricción de columna</b>     descuento_precio numeric <b>check</b> (descuento_precio &gt; 0),                                      -&gt; <b>restricción de columna</b>     <b>check</b> (precio &gt; descuento_precio)); -&gt; <b>restricción de tabla</b></pre> <p>A la restricción de tabla también se le puede asignar un nombre, tal como:</p> <pre>create table productos ( ....     precio numeric <b>check</b> (precio &gt; 0),     descuento_precio numeric,     <b>check</b> (descuento_precio &gt; 0),     <b>constraint</b> <b>descuento_valido check</b> (precio &gt; descuento_precio));</pre> <p>Otras formas de expresar lo mismo serían las siguientes:</p> <ul style="list-style-type: none"> <li>• <b>create table</b> productos ( ....             precio numeric,             <b>check</b> (precio &gt; 0),             descuento_precio numeric,             <b>check</b> (descuento_precio &gt; 0),             <b>check</b> (precio &gt; descuento_precio));</li> <li>• <b>create table</b> productos ( ....             precio numeric <b>check</b> (precio &gt; 0),             descuento_precio numeric,             <b>check</b> (descuento_precio &gt; 0 and precio &gt; descuento_precio));</li> </ul> <p>✓ <b>El chequeo de una restricción es válida if la expresión en el check es true o el valor null.</b></p> <p>✓ <b>Cuidado! Porque si cualquier operando es null, entonces la expresión se evalúa a null.</b></p> |
|---------------------|--|



|          |  |
|----------|--|
|          |  |
| default  | <ul style="list-style-type: none"><li>✓ Si no se especifica ningún valor, entonces el atributo se inicializa a null.</li><li>✓ El valor por defecto puede ser una expresión la cual es evaluada cuando se produzca la inserción del valor.</li></ul> <pre>create table productos (...<br/>    precio numeric default 34.99<br/>);</pre>  |
| not null | <ul style="list-style-type: none"><li>✓ Se usa para especificar que el valor de una columna no puede ser null.</li></ul> <pre>create table productos( ....<br/>    nombre text not null,<br/>    precio numeric not null check (precio &gt; 0));</pre> <p>Por lo tanto, es equivalente a crear una restricción con check, tal como:</p> <pre>check (nombre_columna is not null)</pre> <p>En este caso a la restricción no se da un nombre.</p> <p><b>Consejo:</b> En la mayoría de los diseños de bases de datos, las columnas deberían de ser marcadas como not null.</p>   |
| null     | <ul style="list-style-type: none"><li>✓ Inicializar el valor de la columna con null por defecto.</li></ul> <pre>create table productos(...<br/>    precio numeric (default) null;</pre>  |
| Unique   | <ul style="list-style-type: none"><li>✓ Los valores de una columna específica o grupo de columnas específicas no se pueden repetir, y por lo tanto están catalogadas como unique.</li></ul> <pre>create table productos(...<br/>    id_producto integer unique, ...);</pre> <ul style="list-style-type: none"><li>✓ También se puede especificar como una restricción:</li></ul> <pre>create table productos (...<br/>    id_producto integer, ...<br/>    unique (id_producto));</pre> <ul style="list-style-type: none"><li>✓ Se le puede dar un nombre a la restricción:</li></ul> <pre>create table productos ( ...<br/>    id_producto integer constraint id_producto_constraint unique, ...);</pre> <ul style="list-style-type: none"><li>✓ También se puede definir una restricción única sobre un grupo de columnas.</li></ul> |



|             |   |
|-------------|---|
|             | <p><b>create table</b> ejemplo (<br/>    a integer,<br/>    b integer,<br/>    c integer,<br/>    <b>unique</b> (a, c)); -&gt; <b>la combinación de a y c debe ser unique.</b></p>  |
| Primary key | <p>✓ Una columna o grupo de columnas se puede usar como identificador único para las filas (unique y not null).</p> <p><b>create table</b> productos (<br/>    id_producto <b>integer unique not null, ...</b>);</p> <p>lo cual es equivalente a:</p> <p><b>create table</b> productos (<br/>    id_producto <b>integer primary key, ...</b>);</p> <p>✓ Si hay más de una clave primaria. Por ejemplo,</p> <p><b>create table</b> productos (<br/>    id_producto <b>integer primary key, ...</b>);</p> <p><b>create table</b> ordenes (<br/>    id_orden <b>integer primary key, ...</b>);</p> <p><b>create table</b> orden_items (<br/>    id_producto <b>integer references</b> productos,<br/>    id_orden <b>integer references</b> ordenes, ...<br/><br/>    <b>primary key</b> (id_producto, id_orden));</p> |
| Foreign key | <p>✓ El valor en una columna (o grupo de columnas) debe casar con los valores de una fila en otra tabla. Esta característica se denomina “integridad referencial” entre dos tablas.</p> <p><b>create table</b> ordenes ( -&gt; <b>tabla que referencia</b><br/>    id_orden <b>integer primary key,</b><br/>    id_producto <b>integer references</b> productos (id_producto), ....);</p> <p>productos -&gt; <b>tabla referenciada</b></p> <p>✓ También puede ser un grupo de columnas</p> <p><b>create table</b> ejemplo (<br/>    a <b>integer primary key,</b><br/>    b <b>integer,</b><br/>    c <b>integer,</b><br/>    <b>foreign key</b> (b, c) <b>references</b> otra_tabla (c1, c2));</p>   |



|  |   |
|--|---|
| <p><b>on update<br/>cascade</b></p> <p>/</p> <p><b>on delete<br/>cascade</b></p> | <p>✓ Imaginemos que existe una relación muchos a muchos entre tablas, como por ejemplo:</p> <pre>create table productos (   id_producto integer primary key, ...);</pre> <pre>create table ordenes (   id_orden integer primary key, ...);</pre> <pre>create table orden_items (   id_producto integer references productos,   id_orden integer references ordenes, ...);</pre> <p><b>Pero, que ocurriría si un producto es borrado después de que una orden fuese creada:</b><br/>¿Se debería de borrar la orden también? o ¿se eliminaría la referencia a producto?</p> <p>En el caso de la segunda opción, es necesario 'on delete cascade' tal y como sigue:</p> <pre>create table orden_items (   id_producto integer references productos on delete cascade,   id_orden integer references ordenes on delete cascade, ...);</pre> <p>Por lo tanto, si 'productos' y 'ordenes' son borrados, entonces se debería de eliminar la tupla correspondiente en la tabla 'orden_items'.</p> <p>Además, si una tupla es actualizada en 'productos' y/o en 'ordenes', entonces debe de ser actualizado el valor correspondiente en 'orden_items', si se añade 'on update cascade'.</p> <pre>create table orden_items (   id_producto integer references productos     on delete cascade     on update cascade,   id_orden integer references ordenes     on delete cascade     on update cascade, ...);</pre> |
|--|---|

A continuación, se realizarán las siguientes tareas:

1. Crear los dominios email y postcode en la base de datos, denominados dominio\_email y dominio\_postcode respectivamente, de acuerdo a las especificaciones proporcionadas.
2. Añadir dos columnas a la tabla 'fabricante' denominadas email y postcode cuyos tipos de datos serán dominio\_email y dominio\_postcode respectivamente.
3. Actualiza la información en la tabla 'fabricacion' de acuerdo a los datos de la siguiente tabla.



| fabricante | email                        | postcode |
|------------|------------------------------|----------|
| Kingston   | customerservice@kington.eu   | 92200    |
| Digi-key   | spainsupport@digkey.com      | 44640    |
| Intel      | intelsupport@intel.com       | 55232    |
| AMD        | amdcustomer@amd.com          | 35467    |
| Tesla      | teslasupport@tesla.com       | 39472    |
| Logitech   | logitechservice@logitech.com | 22456    |

4. Crea un nuevo dominio que represente el número de años de garantía para un pc denominado `dominio_anios_garantia`. El valor deberá de ser mayor que 0 y menor o igual que 3. Luego, añade una nueva columna a la tabla 'pc' denominada 'garantia' de tipo 'dominio\_anios\_garantia' con un valor por defecto de 2.
5. Presentar el nombre y el país de aquellos proveedores que suministren todas las memorias. (división)

```
nombre | pais
-----+-----
Romanesco | Italia
(1 fila)
```

6. Presentar la información de aquellas memorias que sean suministradas por todos los fabricantes. (división)

```
id_memoria | capacidad | transferencia_datos | precio | id_fabricante | id_tipo_memoria
-----+-----+-----+-----+-----+-----
300 | 64GB | PC4-2666 | 48.00 | 100 | 200
(1 fila)
```

7. Realizar la consulta para los siguientes apartados y transformarla en función junto con la sentencia que la invoque.

- 7.1. Presenta el identificador del pc y la descripción del tipo de cpu que contiene, además de aquellas cpu's que no han sido instaladas en ningún pc.

```
getinfocpu
-----
(6,"dual core")
(2,"dual core")
(1,"dual core")
(7,"dual core")
(3,"dual core")
(5,"quad core")
(4,"quad core")
(,"ocho core")
(,"seis core")
(9 filas)
```



7.2. Presenta aquellos tipos de cpu, tanto descripción como tipo, que no han sido instaladas en ningún pc.

```
getinfotipocpu
-----
("ocho core",8)
("seis core",6)
(2 filas)
```

7.3. Presenta aquellos discos, tanto la descripción como la capacidad, que han sido fabricados por Intel o Kingston (haz la comparación en letras minúsculas).

```
getinfodiscos
-----
("SSDNOW A2000",500GB)
("SSD A400",120Gb)
("SSD Pro 5450S",256GB)
("SSD 660P",2TB)
(4 filas)
```

7.4. Presenta la descripción y el precio de aquellos discos con un precio mayor o igual que la media de todos los precios.

```
getinfopreciodiscos
-----
(120GB,140.00)
(2TB,268.00)
(2 filas)
```

7.5. Presenta el nombre de todos aquellos fabricantes que realicen más de un tipo de disco.

```
getinfofabricante
-----
Intel
Kingston
(2 filas)
```

7.6. Presenta toda la información de aquellos pc's que tengan la tarjeta gráfica más barata.

```
getinfopcs
-----
(2,500,602,302,701)
(6,500,601,304,701)
(2 filas)
```





8. Crea un trigger denominado 'presenta\_info\_pc', donde cada vez que se realice una operación de insert ('INSERT'), delete ('DELETE') o update ('UPDATE') en la tabla 'pc', se presente un mensaje. Realiza una operación de cada uno de los tipos mencionados en la tabla 'pc', con el fin de verificar que el mensaje se realiza.

**Objetivos de aprendizaje:**

- ✓ **Crear un dominio.**
- ✓ **Repaso general:**
  - **Join**
  - **Agregación.**
  - **Function**
  - **Trigger**