



# MEMORIA: LIBRERÍA PRÁCTICA 1

Prácticas de Estructuras de Datos

Lucía Rivas Molina  
Daniel Santo-Tomás López  
Grupo 1201

## ANÁLISIS DEL DISEÑO

En esta primera práctica hemos diseñado y creado una base de datos de una librería cuyo modelo relacional y diagrama entidad-relación están adjuntos en otros pdfs. Se nos pedían ciertos atributos y ciertas tablas, con sus características:

- Libro, Autor y Relacion\_AL:

Un libro puede tener más de un autor, cada libro y cada autor tienen un Id que los identifican, y tuvimos que buscar una solución para no repetir en una misma tabla todas las características del libro. Para poder establecer una relación entre un libro y sus distintos autores, decidimos crear una nueva tabla *Relacion\_AL* la cual contiene dos columnas: Libro y Autor, los cuales hacen referencia a los Ids del libro y del autor respectivamente. La clave primaria de esta tabla es la relación entre ambas puesto que nunca se va a repetir.

```
CREATE TABLE Libro (  
    Id_libro int PRIMARY KEY,  
    Titulo_original text NOT NULL);  
  
CREATE TABLE Autor(  
    Nombre text NOT NULL,  
    Id_autor int PRIMARY KEY);
```

```
CREATE TABLE Relacion_AL(  
    Libro int REFERENCES Libro(Id_libro),  
    Autor int REFERENCES Autor(Id_autor),  
    PRIMARY KEY(Libro,Autor)  
);
```

- Libro y Edición:

Un libro puede tener varias ediciones puesto que puede estar en idiomas distintos, tener distintos tamaños, distintas editoriales... y cada edición cuenta con su ISBN (clave primaria) para diferenciarla de las demás. Para ello, en la tabla Libro añadimos el campo Título\_original que guarda el título en el idioma que se escribió. La relación entre Edición y Libro se realiza a través del Id del libro.

```
CREATE TABLE Edicion(  
    ISBN int PRIMARY KEY,  
    Titulo text NOT NULL,  
    Id int REFERENCES Libro(Id_libro),  
    Tapa text,  
    Idioma text,  
    Tamaño text,  
    Precio float NOT NULL,  
    Editorial text NOT NULL);
```

- Ofertas:

Esta librería ofrece, además, descuentos a sus compradores. Cada oferta se caracteriza por su número de oferta (clave primaria), su descuento, la fecha en la que empieza, la fecha en la que acaba y el Id del libro al cual se le aplica este descuento, es decir, que se aplica a todas las ediciones de ese libro.

```
CREATE TABLE Oferta (
    Num_oferta int PRIMARY KEY,
    Descuento int NOT NULL,
    Fecha_inicio date,
    Fecha_fin date,
    Libro int REFERENCES Libro(Id_libro));
```

- Ventas y usuarios fidelizados:

Una venta se caracteriza por su número de venta (clave primaria), la fecha en la que se vendió, el precio final del libro, la edición que se vendió (pudiendo vender solo un libro por cada venta, en caso de que un mismo usuario compre dos ediciones iguales serían dos ventas distintas), el método de pago, si se le aplicó alguna oferta (siendo NULL en caso de no aplicarse ninguna) y si el comprador era un usuario fidelizado (siendo Null si no era fidelizado). El fidelizado se caracteriza por su Id (clave primaria), su número de tarjeta y los gastos que ha tenido.

```
CREATE TABLE Ventas (
    Num_venta int PRIMARY KEY,
    Modo_pago text NOT NULL,
    Fecha date NOT NULL,
    Edicion int REFERENCES Edicion(ISBN),
    Precio_final float NOT NULL,
    Oferta int REFERENCES Oferta(Num_oferta),
    Fidelizado int REFERENCES Fidelizados(Id_fidelizado)
);
```

```
CREATE TABLE Fidelizados(
    Id_fidelizado int PRIMARY KEY,
    Tarjeta int UNIQUE NOT NULL,
    Gastos float
);
```

El número de tarjeta es UNIQUE NOT NULL puesto que no pueden existir dos tarjetas idénticas.

## ANÁLISIS DE LAS CONSULTAS

Para poder realizar las consultas, agregamos unos cuantos datos a las tablas creadas mediante el programa del que disponemos PGAdminIII, los cuales iremos viendo a medida que vamos analizando las consultas:

### 1. Dado un título, ¿Cuántas ediciones tiene? ¿En cuántos idiomas?

En esta consulta necesitamos las tablas Libro y Edicion.

```
select count(Id) as NumEdiciones, count(distinct(Idioma)) as NumIdiomas from Edicion, Libro
where Libro.Titulo_original = 'Algebra' AND Edicion.Id = Libro.Id_libro;
```

#### LIBRO

	id_libro [PK] integer	titulo_original text
1	1	Algebra
2	2	Harry Potter
3	3	Tipler
4	4	Juego de Tronos
5	5	Java
6	6	The North
*		

En la consulta, entre las comillas “ metemos el título original del libro que queremos buscar, en nuestro caso, metimos ‘Algebra’, el cual tiene 4 ediciones (11, 12, 13, 14) en tres idiomas distintos.

#### EDICIÓN

	isbn [PK] integer	titulo text	id integer	tapa text	idioma text	tamaño text	precio double precis	editorial text
	11	Algeo	1	Dura	Español	Bolsillo	10.5	Salamandra
	12	AlgLineal	1	Blanda	Frances	Normal	16.5	La petit
	13	Algebra	1	Blanca	Español	Normal	15	Salamandra
	14	Maths	1	Dura	Ingles	Bolsillo	7.9	Collins
	21	HP y la pie	2	Dura	Español	Normal	15	Adelpi
	22	HP y la Ord	2	Blanda	Español	Bolsillo	10	Adelpi
	31	Tipler	3	Duro	Ingles	Normal	20	Oxford
	41	GOT	4	Dura	Ingles	Normal	17	Adelpi
	42	GOT	4	Dura	Frances	Normal	17	Adelpi
	51	Java	5	Blanda	Español	Bolsillo	12.5	EPS
	61	King in the	6	Duro	Ingles	Normal	16.4	Westeros

#### CONSULTA EN PGADMINIII

<pre>select count(Id) as NumEdiciones, count(distinct(Idioma)) as NumIdiomas from Edicion, Libro where Libro.Titulo_original = 'Algebra' AND Edicion.Id = Libro.Id_libro;</pre>		
Panel de Salida		
<div>Salida de datos Comentar Mensajes Historial</div>		
	numediciones bigint	numidiomas bigint
1	4	3

## 2. ¿Cuántos libros se han vendido de un autor dado?

```
select count(Num_venta) as LibrosVendidos from Ventas, Edicion, Libro, Relacion_AL, Autor
where Autor.Nombre = ' ' AND Relacion_AL.Autor = Autor.Id_autor

AND Libro.Id_libro = Relacion_AL.Libro

AND Edicion.Id = Libro.Id_libro AND Ventas.Edicion = Edicion.ISBN;
```

**LIBRO**

	id_libro [PK] integer	titulo_original text
1	1	Algebra
2	2	Harry Potter
3	3	Tipler
4	4	Juego de Tronos
5	5	Java
6	6	The North
*		

**RELACION\_AL**

	libro [PK] integer	autor [PK] integer
1	1	1
2	1	2
3	2	4
4	3	5
5	3	6
6	3	7
7	4	3
8	4	7
9	6	7
*		

**AUTOR**

	nombre text	id_autor [PK] integer
1	Pedro Balodis	1
2	Torrea	2
3	George R. R. Martin	3
4	J. K. Rowling	4
5	Almirante	5
6	Alan Turing	6
7	Jon Snow	7
*		

**VENTAS**

	num_venta [PK] integer	modo_pago text	fecha date	edicion integer	precio_final double precis	oferta integer	fidelizado integer
1	1	Tarjeta	2016-11-27	11	10.5		
2	2	Efectivo	2016-11-27	41	17		1
3	3	Efectivo	2016-11-27	42	17	4	1
4	4	Tarjeta	2016-11-28	41	17	4	2
5	5	Tarjeta	2016-01-28	41	17	4	
6	6	Efectivo	2017-12-20	42	17	6	1
7	7	Tarjeta	2017-01-20	61	16.4		3
8	8	Tarjeta	2017-01-21	51	12.5		2
*							

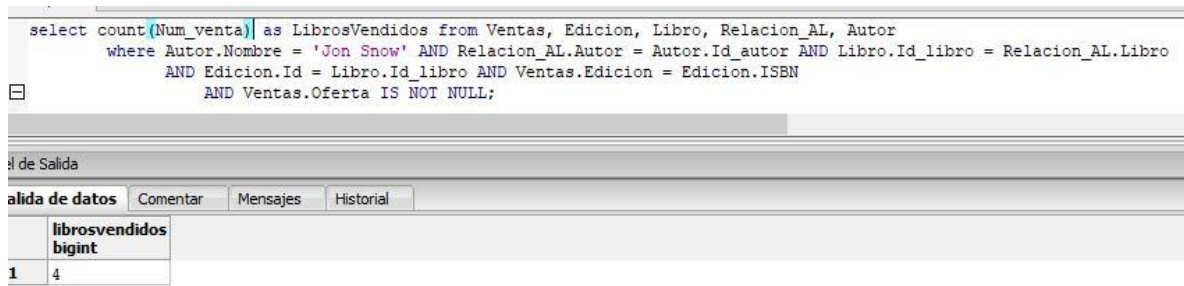
En las comillas metemos el nombre del autor que queremos buscar, en nuestra consulta metemos 'Jon Snow', el cual ha escrito los libros 4 y 6, y se han vendido sus ediciones (41, 42, 61) en seis ventas: las número 2, 3, 4, 5, 6, 7.

<pre>select count(Num_venta) as LibrosVendidos from Ventas, Edicion, Libro, Relacion_AL, Autor where Autor.Nombre = 'Jon Snow' AND Relacion_AL.Autor = Autor.Id_autor AND Libro.Id_libro = Relacion_AL.Libro AND Edicion.Id = Libro.Id_libro AND Ventas.Edicion = Edicion.ISBN;</pre>							
<div> <div>Salida de datos</div> <div>Comentar Mensajes Historial</div> </div>							
	librosvendidos						
	bigint						
1	6						

### 3. ¿Cuántos libros de un autor dado se han vendido en oferta?

```
select count(Num_venta) as LibrosVendidos from Ventas, Edicion, Libro, Relacion_AL, Autor
where Autor.Nombre = ' ' AND Relacion_AL.Autor = Autor.Id_autor
AND Libro.Id_libro = Relacion_AL.Libro
AND Edicion.Id = Libro.Id_libro AND Ventas.Edicion = Edicion.ISBN
```

Igual que en la consulta anterior, comprobamos esta consulta con el autor 'Jon Snow', el cual ha vendido cuatro libros en oferta (ventas 3, 4, 5, 6).



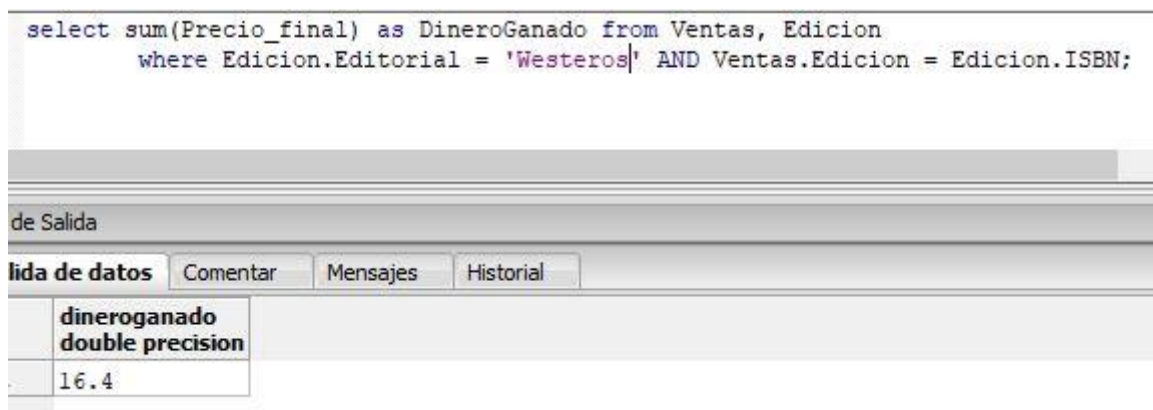
```
select count(Num_venta) as LibrosVendidos from Ventas, Edicion, Libro, Relacion_AL, Autor
where Autor.Nombre = 'Jon Snow' AND Relacion_AL.Autor = Autor.Id_autor AND Libro.Id_libro = Relacion_AL.Libro
AND Edicion.Id = Libro.Id_libro AND Ventas.Edicion = Edicion.ISBN
AND Ventas.Oferta IS NOT NULL;
```

LibrosVendidos	
	bigint
1	4

### 4. ¿Cuánto dinero se ha ganado vendiendo libros de un editor dado?

```
select sum(Precio_final) as DineroGanado from Ventas, Edicion
where Edicion.Editorial = ' ' AND Ventas.Edicion = Edicion.ISBN;
```

Si, por ejemplo, buscamos los libros de la editorial Westeros que hemos vendido en la tabla Ventas, observamos que solamente ha sido uno: la edición número 61, el cual ha costado un total de 16.4 euros.



```
select sum(Precio_final) as DineroGanado from Ventas, Edicion
where Edicion.Editorial = 'Westeros' AND Ventas.Edicion = Edicion.ISBN;
```

DineroGanado	
	double precision
1	16.4

5. ¿Cuántos libros han comprado los usuarios fidelizados?

```
select count(Num_Venta) as NumLibrosVendidos from Ventas
      where Fidelizado IS NOT NULL;
```

En la tabla de Ventas podemos observar que las ventas a usuarios fidelizados han sido 6.

Las ventas número 2, 3, 4, 6, 7, 8; donde el fidelizado no es NULL.

```
select count(Num_Venta) as NumLibrosVendidos from Ventas
      where Fidelizado IS NOT NULL;
```

Panel de Salida	
Salida de datos Comentar Mensajes Historial	
	<b>numlibrosvendidos</b> <b>bigint</b>
1	6

6. ¿Cuántos usuarios fidelizados han comprado libros en inglés?

```
select count(Num_Venta) as LibrosIngles from Ventas, Edicion
      where Edicion.Idioma = 'Ingles' AND Ventas.Edicion = Edicion.ISBN
      AND Fidelizado NOT IN(select Fidelizado from Ventas
                             where Fidelizado = NULL);
```

Las ediciones vendidas en inglés son la 41 y la 61, las cuales tienen un total de cuatro ventas. La idea de esta consulta es buscar todas las ventas en las que no ha comprado un fidelizado e imprimir las demás con un NOT IN.

```
select count(Num_Venta) as LibrosIngles from Ventas, Edicion
      where Edicion.Idioma = 'Ingles' AND Ventas.Edicion = Edicion.ISBN AND
      Fidelizado NOT IN(
          select Fidelizado from Ventas
            where Fidelizado = NULL);
```

Panel de Salida	
Salida de datos Comentar Mensajes Historial	
	<b>librosingles</b> <b>bigint</b>
1	4



7. ¿Cuánto dinero se ha ganado vendiendo libros en francés?

```
select sum(Precio_final) as GananciasFrances from Ventas, Edicion
where Edicion.Idioma = 'Frances' AND Ventas.Edicion = Edicion.ISBN;
```

El único libro vendido en francés ha sido la edición 42 dos veces. Cada una tiene un precio de 17 euros, un total de 34.

```
select sum(Precio_final) as GananciasFrances from Ventas, Edicion
where Edicion.Idioma = 'Frances' AND Ventas.Edicion = Edicion.ISBN;
```

Salida de datos	
gananciasfrances	double precision
1	34

8. ¿En qué días hubo ofertas de libros de la editorial Adelphi?

```
select distinct Num_oferta as Numero, Fecha_inicio as Desde, Fecha_fin as Hasta
from Oferta, Edicion, Libro
where Edicion.Editorial = 'Adelphi' AND Libro.Id_libro = Edicion.Id
AND Oferta.Libro = Libro.Id_libro
order by Fecha_inicio DESC;
```

Las ediciones de la editorial Adelphi son cuatro: 21, 22, 41, 42; pertenecientes a los libros de Id 2 y 4 respectivamente. Nuestra consulta imprime una tabla en la que pone la fecha de inicio de la oferta y la fecha de fin en orden descendente.

	num_oferta [PK] integer	descuento integer	fecha_inicio date	fecha_fin date	libro integer
1	1	20	1990-04-20	1990-04-26	1
2	2	30	1998-09-04	1998-11-26	2
3	3	50	2000-01-01	2000-02-02	3
4	4	75	2016-11-26	2017-01-20	4
5	5	10	2016-11-26	2017-01-20	5
6	6	15	2017-10-31	2017-12-31	4
*					

```
select distinct Num_oferta as Numero, Fecha_inicio as Desde, Fecha_fin as Hasta from Oferta, Edicion, Libro
where Edicion.Editorial = 'Adelphi' AND Libro.Id_libro = Edicion.Id AND Oferta.Libro = Libro.Id_libro
order by Fecha_inicio DESC;
```

Salida de datos		
numero integer	desde date	hasta date
6	2017-10-31	2017-12-31
4	2016-11-26	2017-01-20
2	1998-09-04	1998-11-26



9. ¿Qué usuarios fidelizados no han comprado nunca libros de bolsillo?

```
select distinct(Id_Fidelizado) from Fidelizados
    where Id_fidelizado NOT IN(
        select Id_fidelizado from Fidelizados, Edicion, Ventas
            where Edicion.Tamaño = 'Bolsillo' AND
            Ventas.Edicion = Edicion.ISBN AND
            Fidelizados.Id_fidelizado = Ventas.Fidelizado)
order by Id_Fidelizado;
```

Hay cuatro usuarios fidelizados:  
FIDELIZADOS

	id_fidelizado [PK] integer	tarjeta integer	gastos integer
1	1	123	500
2	2	223	1000
3	3	323	20
4	4	423	150
*			

De estos cuatro usuarios fidelizados solamente ha comprado un libro de bolsillo el usuario 2 (la edición 51, última venta).

Por tanto, nunca han comprado libros de bolsillo los usuarios 1,3 y 4.

```
select distinct(Id_Fidelizado) from Fidelizados
    where Id_fidelizado NOT IN(
        select Id_fidelizado from Fidelizados, Edicion, Ventas
            where Edicion.Tamaño = 'Bolsillo' AND Ventas.Edicion = Edicion.ISBN AND Fidelizados.Id_fidelizado = Ventas.Fidelizado)
order by Id_Fidelizado;
```

de Salida			
Vista de datos Comentar Mensajes Historial			
id_fidelizado integer			
1			
3			
4			

La idea de esta consulta es buscar los usuarios que nunca lo han comprado y para ello, buscamos todos los usuarios que sí han comprado e imprimimos aquellos que no están en esa lista mediante un NOT IN.