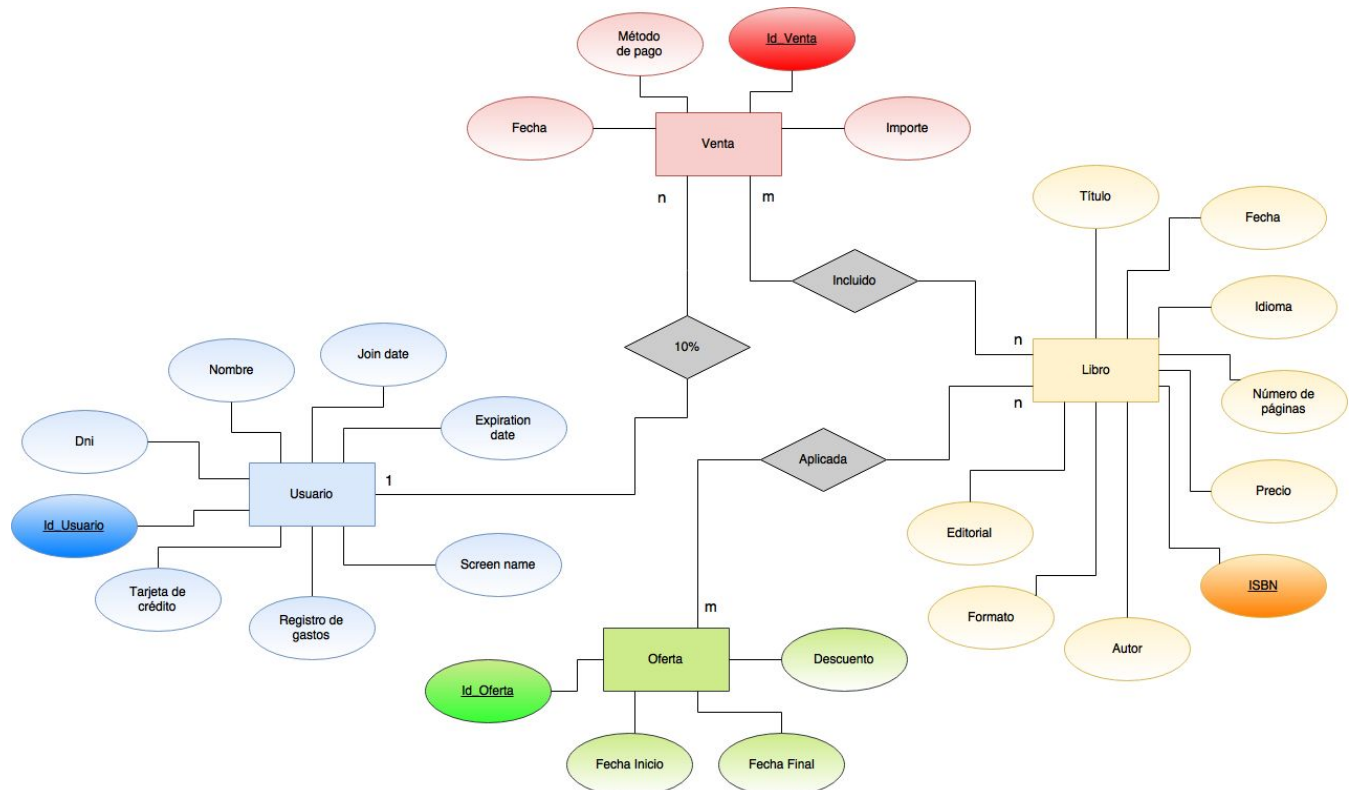


## PRÁCTICA 2 DE ESTRUCTURAS DE DATOS

Realizado por Emilio Cuesta y Adrián Fernández. Grupo 1201.

### Parte I: Refinar el diseño de la base de datos

Revisión del diseño:



*Diseño ER modificado*

#### Cambios realizados:

- Eliminación de las entidades Autor(Nombre, Fecha) y Libro(Título).
- Cambio de nombre de la entidad Edición por Libro.
- Adición de nuevos atributos en Libro y Usuario(Columna “exists”, por ejemplo).
- Modificación de la cardinalidad de la relación “Aplicada” de 1-n a n-m.

#### **Poblar las tablas:**

Nota: Es posible crear la base de datos y añadir los datos ejecutando el código incluido en los archivos .sql, pero antes será necesario reajustar la ruta de los archivos .txt en el código, concretamente en el archivo “3inserciondedatos.sql”.

Código utilizado para poblar las tablas:

```
CREATE TABLE librotemporal(autor VARCHAR(200), titulo VARCHAR(300),  
formato VARCHAR(100), numpag VARCHAR(100), editorial VARCHAR(100),  
fecha VARCHAR(100), idioma VARCHAR(100), ISBN VARCHAR(100));
```

```
COPY librotemporal  
FROM '/home/emi/Desktop/EDAT/practica2/Tablas/LIBROS_FINAL.txt' WITH DELIMITER '  
' ENCODING 'ISO-8859-1';
```

```
CREATE TABLE libropreciotemp( ISBN VARCHAR(30), precio FLOAT);
```

```
COPY libropreciotemp  
FROM '/home/emi/Desktop/EDAT/practica2/Tablas/isbns_precios.txt' WITH DELIMITER '  
' ENCODING 'ISO-8859-1';
```

```
CREATE TABLE usuariotemp(uid INTEGER, scrname VARCHAR(50), nombre  
VARCHAR(50),joindate VARCHAR(20), ccard VARCHAR(30), expiration VARCHAR(20));
```

```
COPY usuariotemp  
FROM '/home/emi/Desktop/EDAT/practica2/Tablas/usuarios.txt' WITH DELIMITER '  
' ENCODING 'ISO-8859-1';
```

```
CREATE TABLE ventastemp(idventa INTEGER, uid INTEGER, ISBN VARCHAR(30), fecha  
VARCHAR(20));
```

```
COPY ventastemp  
FROM '/home/emi/Desktop/EDAT/practica2/Tablas/ventas.txt' WITH DELIMITER '  
' ENCODING 'ISO-8859-1';
```

```
INSERT INTO libro(idioma,fecha,isbn,editorial,formato,titulo,autor,numpag)  
(SELECT min(idioma), min(fecha), isbn, min(editorial), min(formato), min(titulo), min(autor),  
min(numpag)  
FROM LibroTemporal  
GROUP BY isbn);
```

```
UPDATE libro  
SET precio=libropreciotemp.precio  
FROM libropreciotemp  
WHERE libro.isbn=libropreciotemp.isbn;
```

```
INSERT INTO usuario(idusuario,nombre,ccard,scrname,joindate,expdate, exists)  
(SELECT uid, min(nombre), min(ccard), min(scrname), min(joindate), min(expiration), '1'  
FROM usuariotemp  
GROUP BY uid) ;
```

```
INSERT INTO venta(idventa,idusuario,fecha)
SELECT idventa, min(uid), min(fecha)
FROM ventastemp
GROUP BY idventa;
```

```
INSERT INTO rincluido(isbn,idventa)
SELECT V.isbn, V.idventa
FROM ventastemp as V;
```

```
DROP TABLE librotemporal;
DROP TABLE libropreciotemp;
DROP TABLE usuariostemp;
DROP TABLE ventastemp;
```

### Ejecutar las consultas:

El código utilizado para ejecutar las consultas se encuentra en el archivo .sql 4consultasp1.sql. Para cada consulta, se adjunta una captura de su resultado.

-Consultas:

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

```
SELECT      COUNT (isbn) as Ediciones
FROM        libro as L
WHERE       L.titulo='Zorro';
```

	ediciones bigint
1	1

#### ¿En cuántos idiomas?

```
SELECT      COUNT(Caux) as Idiomas
FROM        (SELECT      DISTINCT idioma
              FROM libro
              WHERE       titulo='Zorro') as Caux;
```

	idiomas bigint
1	1

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

```
SELECT      COUNT(R.isbn) as LibrosVendidos

FROM libro as L, rincluido as R
WHERE      L.autor='Arturo Perez Reverte' AND L.isbn=R.isbn;
```

	librosvendidos bigint
1	205

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

```
SELECT      COUNT(O.libro) as LibrosVendidos
FROM libro as L, rincluido as R, raplicada as O
WHERE      L.autor='Arturo Perez Reverte' AND L.isbn=R.isbn AND L.isbn=O.libro;
```

	librosvendidos bigint
1	0

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

```
SELECT      SUM(Q.Precios) as Ganancia
FROM      (SELECT      L.precio as Precios
FROM rincluido as R, venta AS V, libro as L
WHERE      L.editorial='Penguin ' AND L.isbn=R.isbn AND
R.idventa=V.idventa) as Q;
```

	ganancia double precision
1	1636.0099999999999

## 5. ¿Cuántos libros han comprado los usuarios fidelizados?

```
SELECT      COUNT(I.isbn) as Libros
FROM      venta as V, rincluido as I
WHERE      V.idusuario IS NOT NULL AND V.idventa=I.idrincluida;
```

	libros bigint
1	5000

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

```
SELECT      COUNT(Q.Usuarios)
FROM        (SELECT      DISTINCT V.idusuario as Usuarios
              FROM venta as V, rincluido as I, libro as L
              WHERE       V.idusuario IS NOT NULL AND V.idventa=I.idventa AND
                          I.isbn=L.isbn AND L.idioma='Inglés' ) as Q;
```

	count bigint
1	97

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

```
SELECT      SUM(Q.Ganancia)
FROM        (SELECT      L.precio as Ganancia
              FROM venta as V, rincluido as I, libro as L
              WHERE       V.idventa=I.idventa AND I.isbn=L.isbn AND
                          L.idioma='Francés') as Q;
```

	sum double precision
1	26150.00999999999

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

```
SELECT      O.fechai AND O.fechaf
FROM libro as L, oferta as O, replicada as R
WHERE       L.editorial ='Adelphi' AND L.isbn=R.libro AND R.idoferta=O.idoferta;
```

	fechai character varying(15)	fechaf character varying(15)
--	---------------------------------	---------------------------------

*No existe ninguna oferta de momento. Es posible añadirlas utilizando el programa “compra” del fichero compra.c. En el backup se incluye una base de datos con una oferta, pero utilizando el código sql para poblar las tablas no se introducirá ninguna.*

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

```
(SELECT      DISTINCT U.scrname
FROM        usuario as U)
EXCEPT
(SELECT      DISTINCT Us.scrname as Excluir
FROM venta as V, libro as L, rincluido as R, usuario as Us
WHERE       L.formato='De bolsillo' AND L.isbn=R.isbn AND R.idventa=V.idventa
AND V.idusuario IS NOT NULL AND V.idusuario=Us.idusuario)
```

	scrname character varying(50)
1	ericboehs
2	cityofsound
3	JulioAlonso
4	BBCNews
5	JoeF
6	faaborg
7	danrubin
8	landay
9	eoghanmccabe
10	mathias
11	donpark
12	copyblogger
13	semanticvoid
14	fredwilson
15	WIRED
16	pvponline
17	wilshipley
18	hmason
19	jkottke
20	andycroll

*Sólo se muestran los primeros 20 resultados.*

## **Parte II: Programas en C**

Todos los ficheros .c necesarios para ejecutar este programa así como un Makefile están disponibles en el fichero subido a Moodle. Los ficheros usuario.c, oferta.c, compra.c y best\_seller.c están debidamente comentados.

Al ejecutar el código con valgrind hemos obtenido pérdidas de memoria al llamar a funciones de odbc.h, sin embargo esas pérdidas no son debidas a nuestro código.

## 1. Descripción de usuario +:

Consulta si existe algún usuario con el screen name proporcionado, después encuentra el id de usuario más alto y finalmente crea una fila más en la tabla usuario con los datos proporcionados, el id obtenido anteriormente incrementado en una unidad y la fecha actual.

Nota: El programa no permite crear un usuario con el mismo nombre que otro existente incluso si éste ha sido “borrado”, ya que consideramos que el screen name es único para cada usuario de la base de datos.

-Ejemplo de ejecución:

```
alumno@coolm:~/Desktop/Definitivo/Código$ ./usuario + usuario "ejemplo"
Practica numero 2
Realizada por: Emilio Cuesta y Adrian Fernandez
Grupo: 1201
```

Usuario añadido correctamente!

*Ejecución en la consola*

104	4215927		ejemplo		usuario	2016-11-22	10000-1-1	1
-----	---------	--	---------	--	---------	------------	-----------	---

*Modificación de la tabla usuario*

## 2. Descripción de usuario -:

Se asegura de que el usuario que se quiere borrar exista y luego cambia su campo “exists” a 0.

-Ejemplo de ejecución:

```
alumno@coolm:~/Desktop/Definitivo/Código$ ./usuario - usuario
Practica numero 2
Realizada por: Emilio Cuesta y Adrian Fernandez
Grupo: 1201
```

Usuario eliminado correctamente!

*Ejecución en la consola*

104	4215927		ejemplo		usuario	2016-11-22	10000-1-1	0
-----	---------	--	---------	--	---------	------------	-----------	---

*Modificación de la tabla usuario*

### 3. Descripción de oferta:

En el control de argumentos, comprueba que la fecha de inicio sea anterior a la fecha de final y que el descuento sea positivo. **Es muy importante el formato de las fechas (aaaa-mm-dd).**

Luego se asegura de que los isbn introducidos existen en la base de datos, después encuentra el valor máximo de id en las ofertas y crea una oferta con los datos proporcionados y con la id obtenida incrementada en uno. Finalmente añade filas a la tabla de la relación libro-oferta con el id insertado anteriormente y cada uno de los isbn introducidos. La clave primaria de esta relación se incrementa de forma automática debido a que su generación se ha vinculado con una secuencia.

Nota: Este programa no acepta más de 100 isbns a la vez como argumentos de entrada.

-Ejemplo de ejecución:

```
emi@EmilioPC:~/Desktop/EDAT/practica2/Codigo$ ./oferta 20 2016-01-01 2016-12-31
0099453959
Practica numero 2
Realizada por: Emilio Cuesta y Adrian Fernandez
Grupo: 1201
Oferta añadida correctamente
```

*Ejecución en la consola*

	idoferta [PK] integer	descuento integer	fechai character varying(15)	fechaf character varying(15)
1	1	20	2016-01-01	2016-12-31

*Modificación de la tabla oferta*

	oferta integer	libro character varying(200)	idraplicada [PK] integer
1	1	0099453959	1

*Modificación de la tabla de la relación libro-oferta*

Nota: La oferta mostrada en la imagen ha sido incluida en la base de datos del backup. Su uso está disponible para pruebas si fuera necesario.



#### 4. Descripción de compra:

En primer lugar, confirma que los libros que se quieren comprar existen en la base de datos, luego obtiene el mayor id de ventas para introducir una nueva con este valor incrementado en 1. Después, encuentra el id del usuario introducido (si no existe ese usuario, devuelve error). Tras esto, para cada libro introducido, comprueba si tiene ofertas. El programa distingue entre tres casos:

- El libro no tiene ofertas: Entonces al coste de ese libro se le asigna su precio original.
- El libro tiene ofertas pero ninguna en fecha: En este caso también se le asigna el precio que figura en la base de datos.
- El libro tiene una o más ofertas en fecha: En este caso siempre se aplicará aquella que tenga un máximo descuento sobre el precio original del libro.

Sumando los costes de cada libro, se calcula el importe total. Posteriormente, se introduce una nueva tupla en la tabla “ventas” con el id calculado previamente, el importe total, la fecha, el id del usuario que ha realizado la compra y el método de pago.

Justo después, se rellena la tabla de la relación venta-libro con tantas tuplas como libros hayan sido vendidos en esa venta (número de isbnns pasados como argumento). Todas ellas tendrán en el campo idventa el mismo valor, pero en el campo isbn tendrán el propio de cada libro. La clave primaria de las tuplas de esta relación también se asigna automáticamente utilizando una secuencia.

Nota: Este programa no acepta más de 100 isbnns a la vez como argumentos de entrada.

##### -Ejemplo de ejecución:

```
alumno@coolm:~/Desktop/Definitivo/Código$ ./compra jack 0099453959
Practica numero 2
Realizada por: Emilio Cuesta y Adrian Fernandez
Grupo: 1201
```

```
Idusuario es: 12
Idventa es: 5108
El precio es: 11.040000
El descuento es: 20%
El coste del libro con isbn 0099453959 es: 8.83 €
El coste total de la compra es de 8.83 €
Compra añadida correctamente
```

—  
*Ejecución en la consola*

5010	8.832	5109	12	ccard	2016-11-22
------	-------	------	----	-------	------------

*Modificación de la tabla venta*

24853	5109	0099453959	24854
-------	------	------------	-------

*Modificación de la tabla de la relación libro-venta*

## 5. Descripción de best\_seller:

Primero genera una tabla con los isbn y las veces que han sido comprados ordenada de mayor número de compras a mayor, luego muestra los n (número introducido por el usuario) primeros títulos asociados a los isbn junto al número de ejemplares vendidos.

Nota: Este programa no acepta más de '100' como argumento de entrada.

### -Ejemplo de ejecución:

```
alumno@coolm:~/Desktop/Definitivo/Código$ ./best_seller 5
Practica numero 2
Realizada por: Emilio Cuesta y Adrian Fernandez
Grupo: 1201

Best seller #1:
    Titulo: El Coloquio de los Perros
    Numero de ventas: 55
Best seller #2:
    Titulo: Literatura Hispanica De Facil Lectura: Don Quijote 1 + CD I
    Numero de ventas: 55
Best seller #3:
    Titulo: Comprendre le pouvoir : Tome 2
    Numero de ventas: 54
Best seller #4:
    Titulo: Destra e sinistra radicali in America e in Francia: Un saggio su Nozick, Chomsky, Horkheimer, S
    orel, de Benoist
    Numero de ventas: 53
Best seller #5:
    Titulo: El ingenioso hidalgo Don Quijote de la Mancha / The Ingenious Nobleman Don Quixote de la Mancha
    : 2
    Numero de ventas: 53
Operacion realizada correctamente
```

*Ejecución en la consola*