

Grupo: 1202

Andrés Salas Peña y Rodrigo Alonso de Pool Alcántara

Funcionalidades:

Funcionalidad	Implementado (SI / NO)
Tipos de datos	
DBL	SI
LNG	SI
Módulos	
table.c	SI
record.c	SI
Operaciones	
COUNT	SI
UNION	SI
LIMIT	SI
OFFSET	SI

Lista de libros comprados por "jack":

ESECUELE	SQL
usuarios SEQUENTIAL 1 STR Jack C_COLEQCTE SELECT INT 0 P_COL 1 PROJECT ventas SEQUENTIAL PRODUCT 0 2 C_COLEQCOL SELECT INT 1 P_COL 1 PROJECT edicion SEQUENTIAL PRODUCT 0 1 C_COLEQCOL SELECT INT 0 P_COL STR 2 P_COL 2 PROJECT	Select edicion_id, titulo From Edicion NATURAL JOIN Ventas NATURAL JOIN Usuarios Where nombre = 'Jack'

Números de libros comprados por "jack":

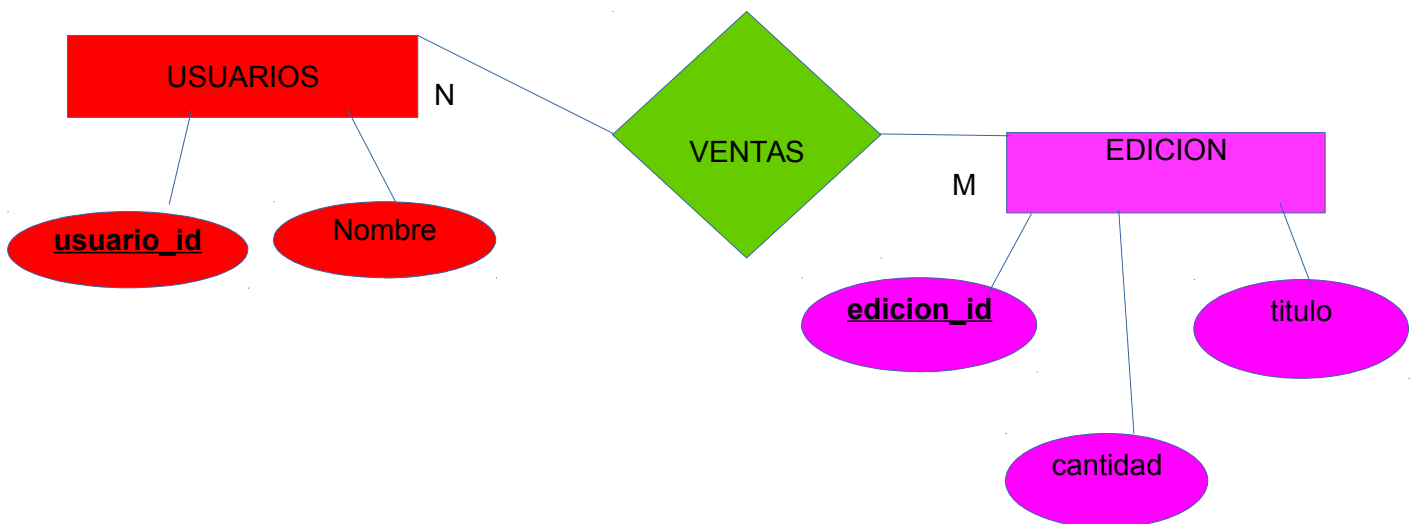
ESECUELE	SQL
usuarios SEQUENTIAL 1 STR Jack C_COLEQCTE SELECT INT 0 P_COL 1 PROJECT ventas SEQUENTIAL PRODUCT 0 2 C_COLEQCOL SELECT INT 1 P_COL 1 PROJECT edicion SEQUENTIAL PRODUCT 0 1 C_COLEQCOL SELECT INT 0 P_COL STR 2 P_COL 2 PROJECT COUNT	Select count(*) From Ventas NATURAL JOIN Usuarios Where nombre = 'Jack'

ER de las tablas *edición*, *ventas* y *usuarios*:

EDICION		
edicion_id	titulo	cantidad

VENTAS	
edicion_id	usuario_id

USUARIOS	
usuario_id	nombre



Ejemplos del funcionamiento:

Se debe entregar un ejemplo de funcionamiento de cada una de las funciones implementadas para la base de datos "*libros_db*". Para ello se puede reutilizar el ".bash" que se entrega con la práctica.

Para observar el correcto funcionamiento de nuestras operaciones se adjunta el bank.bash con las nuevas consultas realizadas, de todas formas se puede observar cuales son las consultas realizadas para probar las operaciones aquí abajo:

```
# Probando la operacion COUNT
# - Count the number of clients
$COMMAND query bank_db << EOF
clients SEQUENTIAL COUNT
EOF
```

```
# Probando la operacion UNION
# - Do the union of the same table accounts
$COMMAND query bank_db << EOF
accounts SEQUENTIAL accounts SEQUENTIAL UNION
EOF
```

```
# Probando la operacion LIMIT
# -Show the first five rows of table accounts
$COMMAND query bank_db << EOF
accounts SEQUENTIAL 5 LIMIT
EOF
```

```
# Probando la operacion OFFSET
## -Show the rows from position 5 of table accounts
$COMMAND query bank_db << EOF
accounts SEQUENTIAL 5 OFFSET
EOF
```

```
# Probando todas las operaciones
## -Count the number of first five accounts union the rows from position 5 of table
accounts.
$COMMAND query bank_db << EOF
accounts SEQUENTIAL 5 OFFSET accounts SEQUENTIAL 5 LIMIT UNION COUNT
EOF
```

Bank.bash

```
andressp05@andressp05-X556UJ:~/EDAT/PRACTICA3/eseecuele/install$ bash bank.bash
```

```
Creating database bank_db
```

```
d> Creating table clients
```

```
d> Creating table accounts
```

```
d> Creating table accounts_clients accounts SEQUENTIAL 5 OFFSET accounts SEQUENTIAL 5 LIMIT UNION COUNT
```

```
d>
```

```
i> copy ./example_files/clients.txt
```

```
(10 rows inserted)
```

```
i> copy ./example_files/accounts.txt
```

```
(15 rows inserted)
```

```
i> copy ./example_files/accounts_clients.txt
```

```
(18 rows inserted)
```

```
i>
```

```
q> 1 456
```

```
3 2411
```

```
10 6545
```

```
14 741
```

```
(4 rows retrieved)
```

```
q>
```

```
q> 2 1422 Simmons
```

```
4 245 Brandon
```

```
4 245 Lee
```

```
5 778 Lee
```

```
13 12 Lee
```

```
15 471 Jones
```

```
(6 rows retrieved)
```

```
q>
```

```
q> 10
```

```
(1 rows retrieved)
```

```
q>
```

```
q> 1 Brooklyn 456
```

```
2 Queens 1422
```

```
3 Brooklyn 2411
```

```
4 Queens 245
```

```
5 Queens 778
```

```
6 Bronx 123
```

```
7 Bronx 1000
```

```
8 Manhattan 555
```

```
9 Manhattan 1877
```

```
10 Brooklyn 6545
```

```
10 Brooklyn 6545
```

```
11 Manhattan 888
```

```
12 Manhattan 1211
```

```
13 Queens 12
```

```
14 Brooklyn 741
```

```
15 Queens 471
```

```
1 Brooklyn 456
```

```
2 Queens 1422
```

```
3 Brooklyn 2411
```

```
4 Queens 245
```

```
5 Queens 778
```

```
6 Bronx 123
```

```
7 Bronx 1000
```

```
8 Manhattan 555
```

```
9 Manhattan 1877
```

```
10 Brooklyn 6545
```

```
11 Manhattan 888
```

```
12 Manhattan 1211
```

```
13 Queens 12
```

```
14 Brooklyn 741
```

```
15 Queens 471
```

```
(30 rows retrieved)
```

```
q>
```

```
q> 1 Brooklyn 456
```

```
2 Queens 1422
```

```
3 Brooklyn 2411
```

```
4 Queens 245
```

```
5 Queens 778
```

```
(5 rows retrieved)
```

```
q>
```

```
q> 6 Bronx 123
```

```
7 Bronx 1000
```

```
8 Manhattan 555
```

```
9 Manhattan 1877
```

```
10 Brooklyn 6545
```

```
11 Manhattan 888
```

```
12 Manhattan 1211
```

```
13 Queens 12
```

```
14 Brooklyn 741
```

```
15 Queens 471
```

```
(10 rows retrieved)
```

```
q>
```

```
q> 15
```

```
(1 rows retrieved)
```

```
q>
```

```
andressp05@andressp05-X556UJ:~/EDAT/PRACTICA3/eseecuele/install$
```

NOTA:

Nuestros scripts son directamente creados tras el comando make esecuele, en la carpeta install. Los scripts que se encontrará son:

- bank.bash y un fichero .bash por cada subacción ejecutada. De este modo puede probar cada acción por separado.

Libros.bash

```
andressp05@andressp05-X556UJ:~/EDAT/PRACTICA3/eseccuele/install$ bash libros.bash
Creating database libros_db
d> Creating table edicion
d> Creating table usuarios
d> Creating table ventas
d>
i> copy ./libros_files/edicion.txt
(15 rows inserted)
i> copy ./libros_files/usuarios.txt
(10 rows inserted)
i> copy ./libros_files/ventas.txt
(20 rows inserted)
i>
q> 1      El universo en una cascara de nuez
2      La caida de los gigantes
3      El resplandor
4      La sombra del viento
7      El secreto del camino
10     El Aleph
(6 rows retrieved)
q>
q> 6
(1 rows retrieved)
q>
andressp05@andressp05-X556UJ:~/EDAT/PRACTICA3/eseccuele/install$
```

NOTA:

Nuestros scripts son directamente creados tras el comando make esecuele, en la carpeta install. Los scripts que se encontrará son:

- bank.bash y un fichero .bash por cada subacción ejecutada. De este modo puede probar cada acción por separado.
- libros.bash y ,al igual que en bank.bash, un fichero .bash por cada subacción ejecutada.

Para ejecutar el main de prueba de nuestras tables y records se debe ejecutar el makefile que se encuentra en la carpeta development/database/ y tras hacer el make ejecutar el fichero test que se genera.