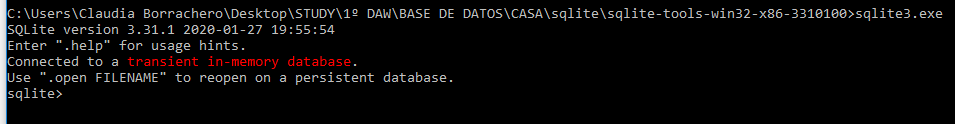
*Claudia Nerea Borrachero Mallofret*

**SQL\_LITE**

Crear un documento que recoja los siguientes apartados:

1. Instalación del servidor SQLite. Incluye las pantallas del proceso



1. Instalación de un cliente SQLite. Puede ser tanto gráfico como por consola. Incluye las pantallas del proceso.
2. Equivalente SQLite a los siguientes comandos SQL:

● show databases; **.databases**

● show tables; **.tables**

● use; **.open**

● create database; **sqlite3 test.db**

● drop database; **There is no DROP DATABASE command in SQLite. If you want to drop a database, all you have to do is to delete the database file.**

1. Comparar los tipos de datos existentes en MySQL con los tipos de datos existentes en SQLite
2. SQLite admite estos tipos de datos: Blob, Integer, Null, Text, Real.

**TEXT**: variable de tipo texto que se almacena en formato UTF, aquí tenemos agrupados los diferentes tipos de datos SQL para las variables de texto: CHARACTER(20), VARCHAR(255), VARYING CHARACTER(255), NCHAR(55), NATIVE CHARACTER(70), NVARCHAR(100), TEXT y CLOB.

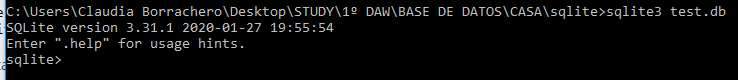
**NUMERIC**: NUMERIC, DECIMAL(10,5), BOOLEAN, DATE y DATETIME

**INTEGER**: es un entero con signo que se almacena con un longitud que va en función del tipo de dato definido: INTEGER, TINYINT, SMALLINT, MEDIUMINT, BIGINT, UNSIGNED BIG INT, INT2 y INT8

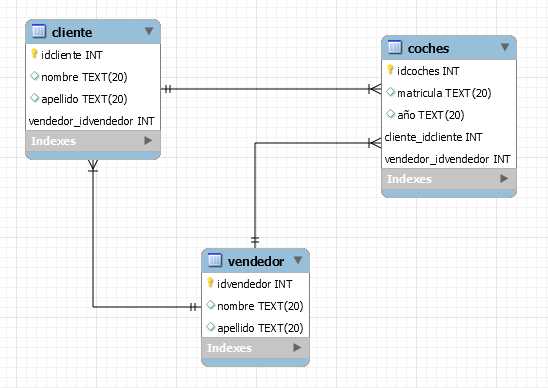
**REAL**: es un dato de tipo float, sus diferentes versiones variarán en la precisión: REAL, DOUBLE, DOUBLE PRECISION y FLOAT

**BLOB**: los datos se almacean en el mismo formato en que se introducen.

1. MySQL admite los tipos de datos mencionados a continuación: Tinyint, Smallint, Mediumint, Int, Bigint, Double, Float, Real, Decimal, Double precision, Numeric, Timestamp, Date, Datetime, Char, Varchar, Year, Tinytext, Tinyblob, Blob, Text, MediumBlob, MediumText, Enum, Set , Longblob, Longtext.
2. Crear una base de datos que tenga al menos 3 tablas (deben estar todas relacionadas) y que esté rellena con al menos 5 filas (las tres tablas). Hacer una copia de seguridad de dicha base de datos y adjuntarla al documento

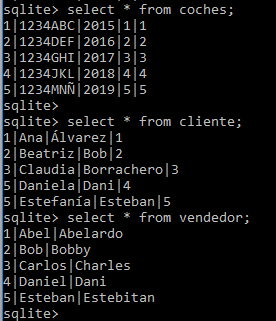




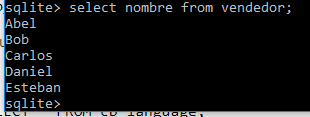


1. Sobre las base de datos anterior realizar los siguientes tipos de consultas:

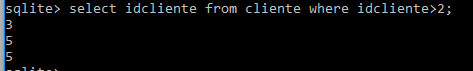
● Un select sobre una única tabla con todos los campos



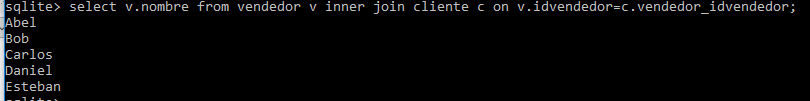
● Un select sobre una única tabla que muestre sólo un campo de la tabla



● Un select sobre una única tabla con una condición.



● Un select que utilice un Join. Comparar dichas consultas con sus equivalentes en MySQL.



En MySQL se realizan las mismas consultas.

**BASE DE DATOS**

CREATE TABLE vendedor (

`idvendedor` INTEGER NOT NULL,

`nombre` TEXT NOT NULL,

`apellido` TEXT NOT NULL,

PRIMARY KEY (`idvendedor`));

CREATE TABLE cliente (

`idcliente` INTEGER NOT NULL,

`nombre` TEXT NOT NULL,

`apellido` TEXT NOT NULL,

`vendedor\_idvendedor` INTEGER NOT NULL,

PRIMARY KEY (`idcliente`, `vendedor\_idvendedor`),

CONSTRAINT `fk\_cliente\_vendedor1`,

FOREIGN KEY (`vendedor\_idvendedor`)

REFERENCES `vendedor` (`idvendedor`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

CREATE TABLE coches (

`idcoches` INTEGER NOT NULL,

`matricula` TEXT NOT NULL,

`año` TEXT NOT NULL,

`cliente\_idcliente` INTEGER NOT NULL,

`vendedor\_idvendedor` INTEGER NOT NULL,

PRIMARY KEY (`idcoches`, `cliente\_idcliente`, `vendedor\_idvendedor`),

CONSTRAINT `fk\_coches\_cliente`

FOREIGN KEY (`cliente\_idcliente`)

REFERENCES `cliente` (`idcliente`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_coches\_vendedor1`

FOREIGN KEY (`vendedor\_idvendedor`)

REFERENCES `vendedor` (`idvendedor`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

INSERT INTO `vendedor` (`idvendedor`, `nombre`, `apellido`) VALUES (1, 'Abel', 'Abelardo');

INSERT INTO `vendedor` (`idvendedor`, `nombre`, `apellido`) VALUES (2, 'Bob', 'Bobby');

INSERT INTO `vendedor` (`idvendedor`, `nombre`, `apellido`) VALUES (3, 'Carlos', 'Charles');

INSERT INTO `vendedor` (`idvendedor`, `nombre`, `apellido`) VALUES (4, 'Daniel', 'Dani');

INSERT INTO `vendedor` (`idvendedor`, `nombre`, `apellido`) VALUES (5, 'Esteban', 'Estebitan');

INSERT INTO `cliente` (`idcliente`, `nombre`, `apellido`, `vendedor\_idvendedor`) VALUES (1, 'Ana', 'Álvarez', 1);

INSERT INTO `cliente` (`idcliente`, `nombre`, `apellido`, `vendedor\_idvendedor`) VALUES (2, 'Beatriz', 'Bob', 2);

INSERT INTO `cliente` (`idcliente`, `nombre`, `apellido`, `vendedor\_idvendedor`) VALUES (3, 'Claudia', 'Borrachero', 3);

INSERT INTO `cliente` (`idcliente`, `nombre`, `apellido`, `vendedor\_idvendedor`) VALUES (5, 'Daniela', 'Dani', 4);

INSERT INTO `cliente` (`idcliente`, `nombre`, `apellido`, `vendedor\_idvendedor`) VALUES (5, 'Estefanía', 'Esteban', 5);

INSERT INTO `coches` (`idcoches`, `matricula`, `año`, `cliente\_idcliente`, `vendedor\_idvendedor`) VALUES (1, '1234ABC', '2015', 1, 1);

INSERT INTO `coches` (`idcoches`, `matricula`, `año`, `cliente\_idcliente`, `vendedor\_idvendedor`) VALUES (2, '1234DEF', '2016', 2, 2);

INSERT INTO `coches` (`idcoches`, `matricula`, `año`, `cliente\_idcliente`, `vendedor\_idvendedor`) VALUES (3, '1234GHI', '2017', 3, 3);

INSERT INTO `coches` (`idcoches`, `matricula`, `año`, `cliente\_idcliente`, `vendedor\_idvendedor`) VALUES (4, '1234JKL', '2018', 4, 4);

INSERT INTO `coches` (`idcoches`, `matricula`, `año`, `cliente\_idcliente`, `vendedor\_idvendedor`) VALUES (5, '1234MNÑ', '2019', 5, 5);