



MYSQL



PROPÓSITO



Comprender que es SQL y el manejo de sentencias en SQL para el modelado de una Base de Datos

TABLA DE SABERES



HACER

- Utilizar sentencias en SQL con el fin de construir y gestionar una base de datos.



SABER

- Sentencias SQL.
- Operadores.



SER

- Analiza la importancia de la gestión de la información en el mundo real.

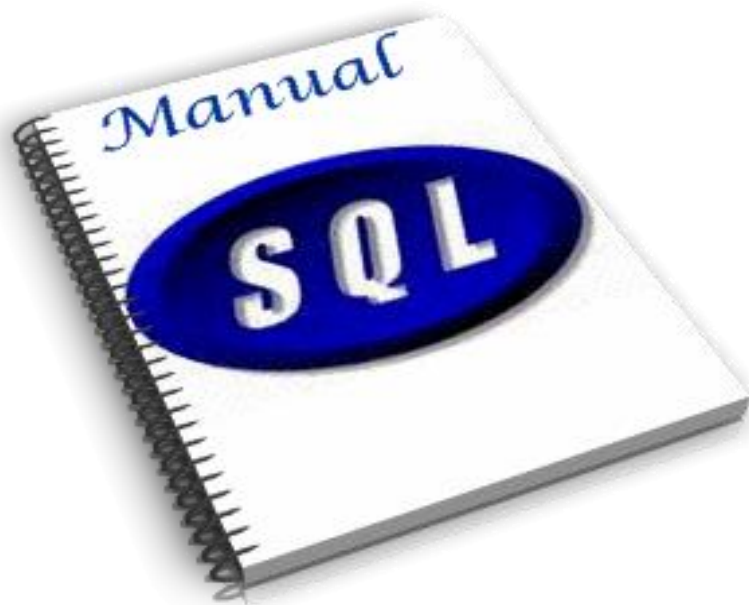
MySQL

- Es una herramienta que te permite diseñar de forma visual las bases de datos facilitando la tarea de trabajar con tablas y vistas
- Es una herramienta de modelado de bases de datos visual multiplataforma desarrollado por My SQL, esta disponible para Windows, linux Mac os.



SQL

- ❑ **SQL** (Structured query language)
- ❑ Creado por IBM
- ❑ Interface entre usuario y el SGBD



CARACTERISTICAS

- Proporciona una representación visual de tablas y vistas, procedimientos y funciones almacenadas y claves foráneas.
- Permite acceso a bases de datos e ingeniería inversa de las mismas para crear las SQL de creación.
- Ofrece sincronización con la base de datos y el modelado.
- Permite generar los scripts SQL a partir del modelo creado.

CARACTERISTICAS

- ❑ Ofrece una arquitectura extensible.
- ❑ Tiene soporte para exportar los datos como scripts SQL CREATE
- ❑ Permite importar de DB Designer
- ❑ Ofrece soporte completo a las características de SQL

PARA QUE SIRVE MYSQL

- Es la herramienta para el diseño visual de esquemas de bases de datos, en la pestaña MySQL model se especifica la estructura física en tablas y vistas con multitud de parámetros definibles en el panel inferior.
- Contiene un editor de diagramas, los elementos pueden arrastrarse desde el catalogo al área de trabajo, usando la caja de herramientas lateral
- Puede exportar el diagrama como imagen o PDF así como generar Scripts SQL CREATE o ALTER

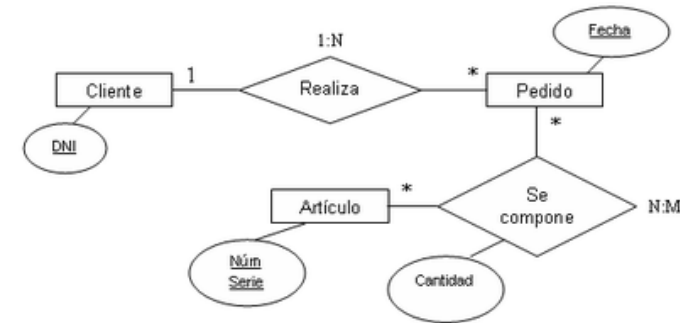
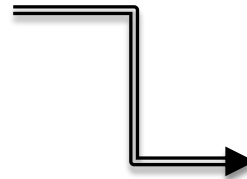
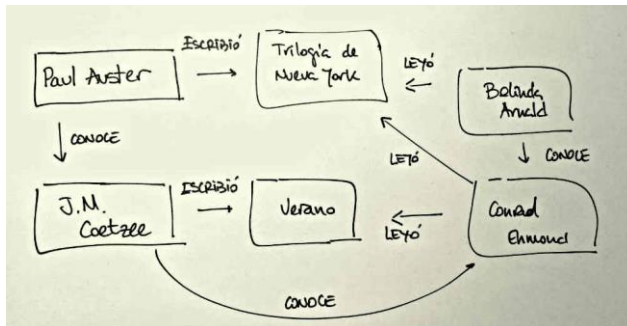
DISEÑO DE BASES DE DATOS

- El diseño de una base de datos consiste en definir la estructura de los datos que debe tener una BD.
- En el caso relacional, esta estructura será un conjunto de esquemas de relación con sus atributos, claves primarias y foráneas entre otras



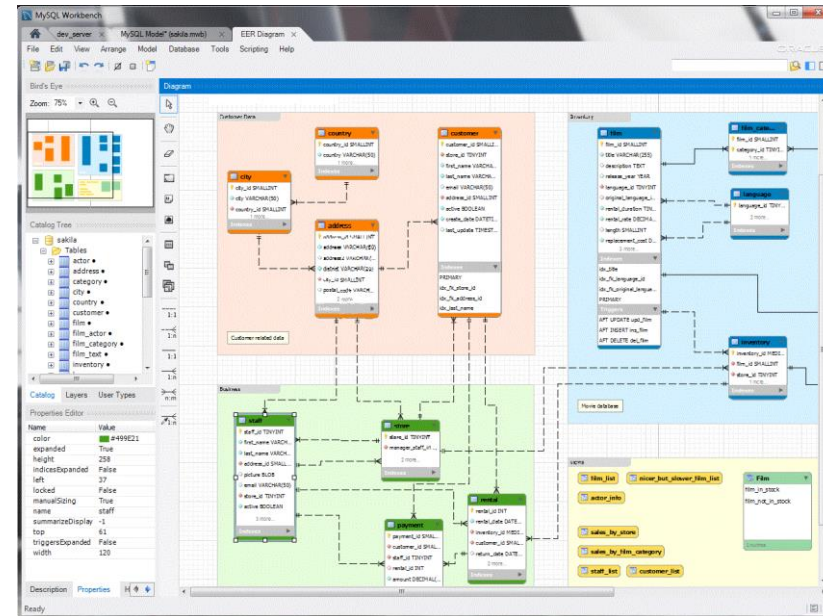
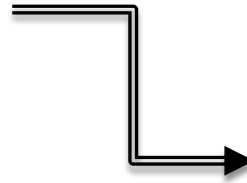
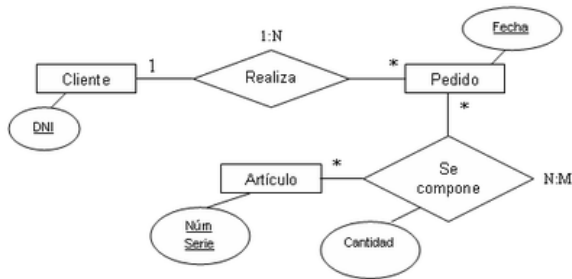
DISEÑO CONCEPTUAL

- En esta etapa se obtiene una estructura de la información de la futura BD independiente de la tecnología a utilizar (MER)



DISEÑO LOGICO

- El objetivo es transformar el esquema conceptual en un modelo de datos determinado para un SGDB, a través de una herramienta informática.



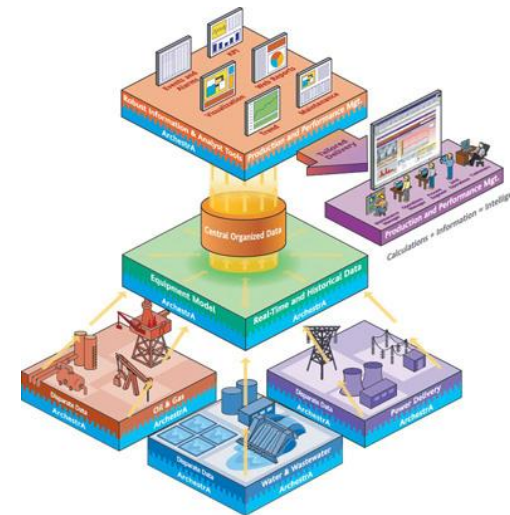
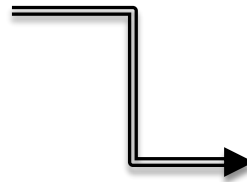
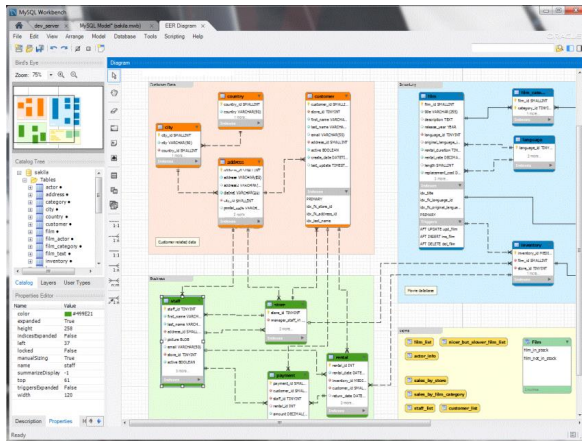
□ LENGUAJE DE DATOS DE CONTROL (DCL)

Un **Lenguaje de Control de Datos (DCL** por sus siglas en inglés: **Data Control Language**) es un lenguaje proporcionado por el Sistema de Gestión de Base de Datos que incluye una serie de comandos SQL que permiten al administrador controlar el acceso a los datos contenidos en la Base de Datos.

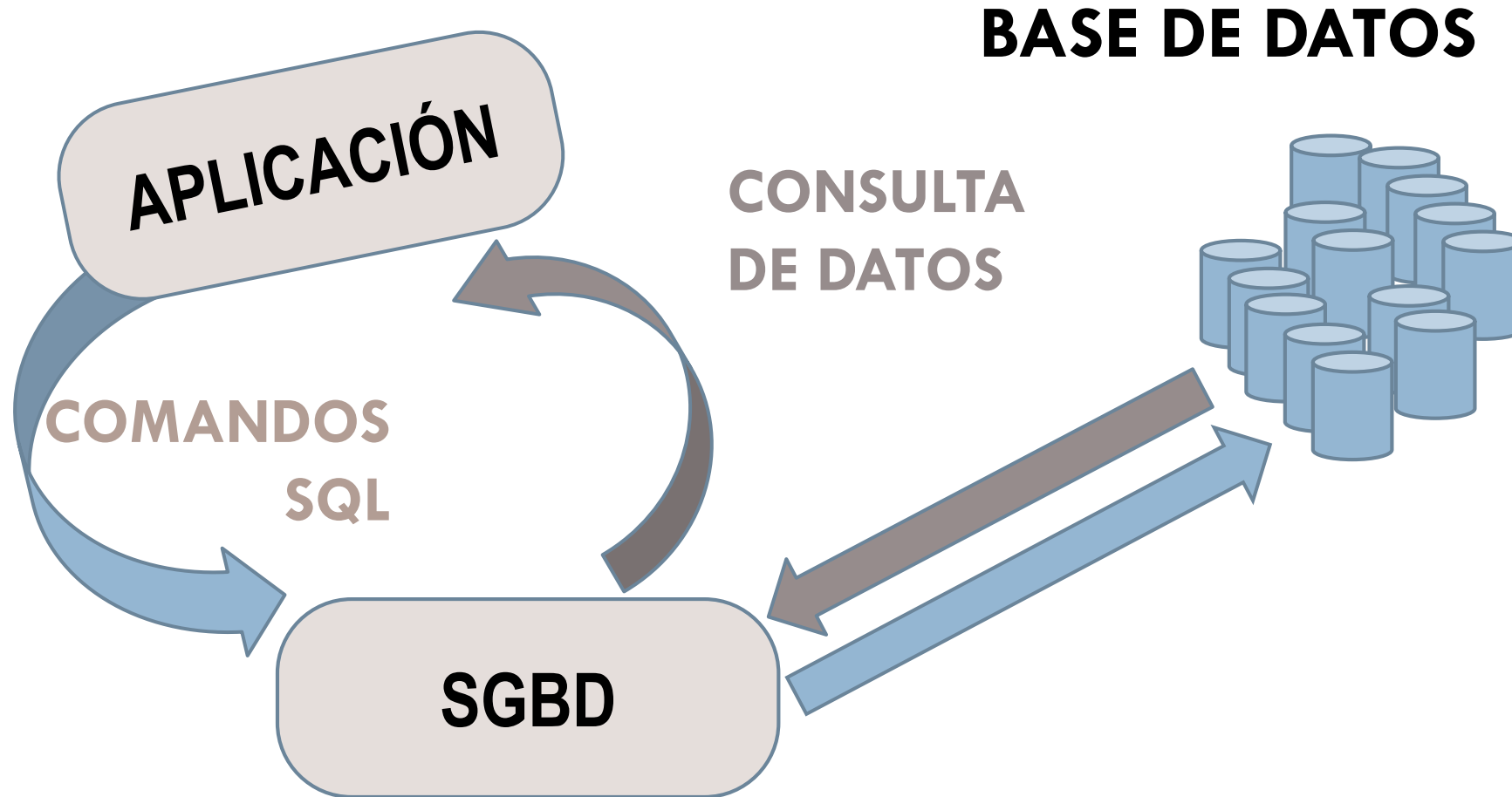
SELECT	Selección de datos
INSERT UPDATE DELETE MERGE	DML: Lenguaje de manipulación de datos
CREATE ALTER DROP RENAME TRUNCATE	DDL: Lenguaje de definición de datos
COMMIT ROLLBACK SAVEPOINT	Control de transacciones
GRANT REVOKE	DCL: Lenguaje de control de datos

DISEÑO FISICO

- Recibe como entrada el esquema lógico, que es una descripción de la implementación de una base de datos, describe las estructuras, de almacenamiento, métodos utilizados para tener acceso a los datos, consultas, relaciones entre otros

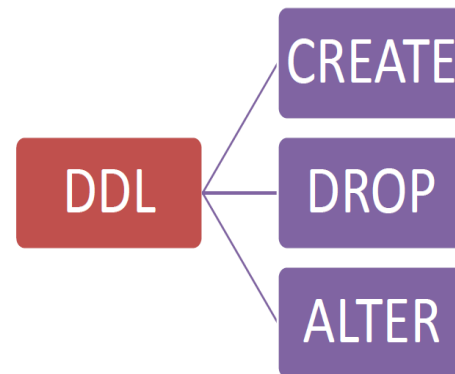


Rendimiento



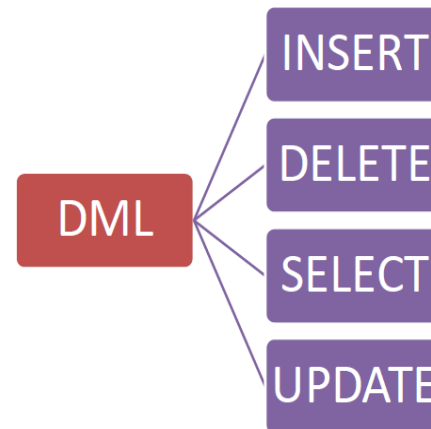
LENGUAJE DE DEFINICIÓN DE DATOS (DDL)

Un **lenguaje de definición de datos** (**Data Definition Language**, DDL por sus siglas en inglés) es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios de la misma llevar a cabo las tareas de definición de las estructuras que almacenarán los datos así como de los procedimientos o funciones que permitan consultarlos



▣ LENGUAJE DE MANIPULACIÓN DE DATOS (DML)

Lenguaje de Manipulación de Datos (**Data Manipulation Language, DML**) es un lenguaje proporcionado por los sistemas gestores de bases de datos que permite a los usuarios de la misma llevar a cabo las tareas de consulta o modificación de los datos contenidos en las Bases de Datos del Sistema Gestor de Bases de Datos.



INTEGER (INT)

El tipo INT se usa para representar un número entero con signo, de -2147483648 a 2147483647.

INT(ancho a mostrar)

INT(3);

Ejm: 1 , 56 , -120 , 899

TinyInt

- es un número entero con o sin signo. Con signo el rango de valores válidos va desde -128 a 127. Sin signo, el rango de valores es de 0 a 255

TinyInt (*ancho a mostrar*)

TinyInt (123);

Tinyint(1) 1/0

FLOAT - DOUBLE

El tipo FLOAT representa un número con coma flotante y *precisión* de 0 a 23 y; el DOUBLE, un número con coma flotante y *precisión* de 24 a 53.

FLOAT/DOUBLE(ancho a mostrar, números de dígitos a almacenar a continuación del punto decimal)

FLOAT (3,2);

DOUBLE (3,2);

Ejm: 1.23 , 240.67 , 8.23

Bit ó Bool

- un número entero que puede ser 0 ó 1

SmallInt

- número entero con o sin signo. Con signo el rango de valores va desde -32768 a 32767. Sin signo, el rango de valores es de 0 a 65535.

MediumInt

- número entero con o sin signo. Con signo el rango de valores va desde -8.388.608 a 8.388.607. Sin signo el rango va desde 0 a 16

BigInt

- número entero con o sin signo. Con signo el rango de valores va desde -9.223.372.036.854.775.808 a 9.223.372.036.854.775.807. Sin signo el rango va desde 0 a 18.446.744.073.709.551.615.

Float

- número pequeño en coma flotante de precisión simple. Los valores válidos van desde $-3.402823466E+38$ a $-1.175494351E-38$, 0 y desde $1.175494351E-38$ a $3.402823466E+38$.

xReal, Double

- número en coma flotante de precisión doble. Los valores permitidos van desde $-1.7976931348623157E+308$ a $-2.2250738585072014E-308$, 0 y desde $2.2250738585072014E-308$ a $1.7976931348623157E+308$

Decimal, Dec, Numeric

- Número en coma flotante desempaquetado. El número se almacena como una cadena

$M+2$ bytes sí $D > 0$, $M+1$ bytes sí $D = 0$

VARCHAR

El tipo VARCHAR almacena una cadena de longitud variable. En MySQL 5.0, la longitud puede especificarse de 0 a 255 antes de MySQL 5.0.3, y de 0 a 65535 en 5.0.3 y versiones posteriores.

VARCHAR(longitud de la cadena)

VARCHAR(16);

Ejm: 'oso', 'preventivo', 'rosa'

Char(n)

- almacena una cadena de longitud fija. La cadena podrá contener desde 0 a 255 caracteres.

Tipos de cadena

- **TinyText y TinyBlob:** Columna con una longitud máxima de 255 caracteres.

Blob y Text: un texto con un máximo de 65535 caracteres.

Tipos de cadena

- **MediumBlob y MediumText:** un texto con un máximo de 16.777.215 caracteres.

LongBlob y LongText: un texto con un máximo de caracteres 4.294.967.295. Hay que tener en cuenta que debido a los protocolos de comunicación los paquetes pueden tener un máximo de 16 Mb.

Tipos de cadena

- **Enum:** campo que puede tener un único valor de una lista que se especifica. El tipo Enum acepta hasta 65535 valores distintos

Set: un campo que puede contener ninguno, uno ó varios valores de una lista. La lista puede tener un máximo de 64 valores.

Tipos fecha

- A la hora de almacenar fechas, hay que tener en cuenta que Mysql no comprueba de una manera estricta si una fecha es válida o no. Simplemente comprueba que el mes esta comprendido entre 0 y 12 y que el día esta comprendido entre 0 y 31.

DATETIME

El tipo DATETIME se usa cuando se necesitan valores que contienen información de fecha y hora. El tipo DATE se usa cuando se necesita sólo un valor de fecha, sin una parte de hora.

`DATETIME(AAAA-MM-DD HH:MM:SS)`

`DATE(AAAA-MM-DD)`

El rango soportado es de '1000-01-01 00:00:00' a '9999-12-31 23:59:59'.

Date

- tipo fecha, almacena una fecha. El rango de valores va desde el 1 de enero del 1001 al 31 de diciembre de 9999. El formato de almacenamiento es de año-mes-día

Time

- almacena una hora. El rango de horas va desde -838 horas, 59 minutos y 59 segundos a 838, 59 minutos y 59 segundos. El formato de almacenamiento es de 'HH:MM:SS'

Year

- almacena un año. El rango de valores permitidos va desde el año 1901 al año 2155. El campo puede tener tamaño dos o tamaño 4 dependiendo de si queremos almacenar el año con dos o cuatro dígitos.

DATETIME - DATE

'1998-12-31 11:30:45'

ES IGUAL A:

'98-12-31 11:30:45'

'98.12.31 11+30+45'

'98/12/31 11*30*45'

'98@12@31 11^30^45'

Tipos de cadena

- **Char(n):** almacena una cadena de longitud fija. La cadena podrá contener desde 0 a 255 caracteres.