

# Base de Datos I

---

Introducción a SQL - DDL

# Herramientas

- MySQL

- Es un Sistema de Gestión de Bases de Datos (SGBD)
- Tiene una versión gratuita y otra paga
- Hay que descargar el [Server](#) y el [Workbench \(IDE\)](#)



- SQLFiddle (opcional)

- Es una web que genera una Base de datos para que podamos trabajar directamente
- Se puede acceder en [sqlfiddle.com](http://sqlfiddle.com)

# Conceptos

- SQL (Structured Query Language)

- DDL (Data Definition Language)

- Se utiliza para crear y modificar la estructura de la Base de datos
    - Algunas palabras reservadas son: CREATE, ALTER, DROP, TRUNCATE

- DML (Data Manipulation Language)

- Son sentencias utilizadas para la manipulación (crear, eliminar, modificar, consultar) de los datos de una base de datos.
    - Algunas palabras reservadas: SELECT, INSERT, UPDATE, DELETE,

# Tipos de datos

---

# Tipos de datos

- INT

- Ocupa 4 bytes
- Puede alojar un número del -2147483648 al 2147483647
- Opción UNSIGNED
- Otras variantes: TINYINT, SMALLINT, MEDIUMINT, BIGINT

# Variantes Integer

MySQL 8.0 Reference Manual / ... / Integer Types (Exact Value) - INTEGER, INT, SMALLINT, TINYINT, MEDIUMINT, BIGINT

version 8.0 ▼

## 11.1.2 Integer Types (Exact Value) - INTEGER, INT, SMALLINT, TINYINT, MEDIUMINT, BIGINT

MySQL supports the SQL standard integer types `INTEGER` (or `INT`) and `SMALLINT`. As an extension to the standard, MySQL also supports the integer types `TINYINT`, `MEDIUMINT`, and `BIGINT`. The following table shows the required storage and range for each integer type.

**Table 11.1 Required Storage and Range for Integer Types Supported by MySQL**

Type	Storage (Bytes)	Minimum Value Signed	Minimum Value Unsigned	Maximum Value Signed	Maximum Value Unsigned
TINYINT	1	-128	0	127	255
SMALLINT	2	-32768	0	32767	65535
MEDIUMINT	3	-8388608	0	8388607	16777215
INT	4	-2147483648	0	2147483647	4294967295
BIGINT	8	$-2^{63}$	0	$2^{63}-1$	$2^{64}-1$

# Tipos de datos

- **DECIMAL(M,D)**

- Longitud no requerida. M=Longitud total, incluyendo decimales (Default 10), D=Decimales (Default 0).
- Es equivalente a NUMERIC
- Otras variantes: FLOAT, DOUBLE

# Tipos de datos

- BOOL, BOOLEAN
  - Es equivalente a TINYINT



# Tipos de datos

- DATE

- Formato Default: 'yyyy-mm-dd'
- La base de datos almacena la fecha en un formato desconocido por nosotros
- Para enviar y obtener fechas a la base de datos utilizamos una cadena de caracteres ej 'yyyy-mm-dd', 'yyyymmdd', 'yyyy/mm/dd'

# Tipos de datos

- TIME
  - Formato Default: 'hh:mi:ss'

# Tipos de datos

- DATETIME
  - Formato Default: 'yyyy-mm-dd hh:mi:ss'

# Tipos de datos

- CHAR(M)
  - Soporta una cantidad fija de caracteres
  - Longitud no requerida (Default 1)
  - Completa con espacios hasta la longitud
  - Hasta 255 caracteres

# Tipos de datos

- **VARCHAR(M)**

- Sirve para guardar una cadena de caracteres variable
- Longitud requerida
- Longitud + 1 byte (cuando longitud es  $\leq 255$ ) o 2 bytes ( $>255$ )
- Hasta 65532 caracteres

# Tipos de datos

- TEXT

- Otras variantes: TINYTEXT, MEDIUMTEXT, LONGTEXT

# Tipos de datos

- BLOB

- Binary Large Object
- Nos sirve para guardar archivos de cualquier tipo en la base de datos
- Otras variantes: TINYBLOB, MEDIUMBLOB, LONGBLOB

# Sentencias DDL

---



# Nomenclatura

PALABRA RESERVADA OPCION1 | OPCION2 [OPCIONAL] <nombre>;

# Creación de una base de datos

```
CREATE DATABASE | SCHEMA [IF NOT EXISTS] <NOMBRE>;
```

```
DROP DATABASE | SCHEMA [IF EXISTS] <NOMBRE>;
```

```
USE <NOMBRE>;
```



```
CREATE SCHEMA IF NOT EXISTS universidad;  
DROP DATABASE universidad;  
USE universidad;
```

# Creación de tablas

CREATE TABLE <TABLA>

(<CAMPO\_1> <TIPO\_DATO\_1> [RESTRICCIONES\_CAMPO\_1],  
<CAMPO\_2> <TIPO\_DATO\_2> [RESTRICCIONES\_CAMPO\_2],  
...  
<CAMPO\_N> <TIPO\_DATO\_N> [RESTRICCIONES\_CAMPO\_N],  
  
[RESTRICCIONES\_TABLA]);



```
CREATE TABLE alumnos (legajo SMALLINT,  
    nombre VARCHAR(40),  
    apellido VARCHAR(40),  
    fecha_nacimiento DATE);
```

# Restricciones de campo

- NOT NULL
- PRIMARY KEY (Claves primarias simples)
- UNIQUE (Claves únicas simples)
- AUTO\_INCREMENT (Tipos numéricos. Debe ser PK. Sólo puede haber una)
- DEFAULT <VALOR>




```
CREATE TABLE alumno (legajo SMALLINT PRIMARY KEY AUTO_INCREMENT,  
                        nombre VARCHAR(40) NOT NULL,  
                        apellido VARCHAR(40) NOT NULL,  
                        fecha_nacimiento DATE);
```

# Restricciones de tabla

- [CONSTRAINT <NOMBRE>] PRIMARY KEY(<LISTA\_CAMPOS>)
- [CONSTRAINT <NOMBRE>] FOREIGN KEY(<LISTA\_CAMPOS>) REFERENCES <TABLA\_REF>(<LISTA\_CAMPOS\_REF>)
- [CONSTRAINT <NOMBRE>] UNIQUE(<LISTA\_CAMPOS>)
- [CONSTRAINT <NOMBRE>] INDEX(<LISTA\_CAMPOS>)





```
CREATE TABLE tipo_documento (tipo CHAR(3) PRIMARY KEY,  
                                descripcion VARCHAR(50));  
  
CREATE TABLE alumno (tipo_documento CHAR(3),  
                        numero_documento VARCHAR(15),  
                        nombre VARCHAR(40),  
                        apellido VARCHAR(40) NOT NULL,  
                        fecha_nacimiento DATE,  
                        CONSTRAINT alumno_pk PRIMARY KEY (tipo_documento, numero_documento),  
                        FOREIGN KEY (tipo_documento) REFERENCES tipo_documento(tipo));
```

# Borrado de tablas

`DROP TABLE <TABLA>;`



A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The text `DROP TABLE alumno;` is displayed in a light purple monospace font.

```
DROP TABLE alumno;
```

# Modificación de tablas

- `ALTER TABLE <TABLA> ADD COLUMN <CAMPO> <TIPO_DATO> [RESTRICCIONES_CAMPO];`
- `ALTER TABLE <TABLA> DROP COLUMN <CAMPO>;`
- `ALTER TABLE <TABLA> CHANGE <NOMBRE_CAMPO_VIEJO> <NOMBRE_CAMPO_NUEVO> <TIPO_DATO> [RESTRICCIONES_CAMPO];`



```
ALTER TABLE alumno ADD COLUMN domicilio VARCHAR(100) NOT NULL;
```

```
ALTER TABLE alumno DROP COLUMN nombre;
```

```
ALTER TABLE alumno CHANGE apellido nombre_apellido VARCHAR(100) NOT NULL;
```

# Modificación de restricciones

- `ALTER TABLE <TABLA> ADD PRIMARY KEY(<LISTA_CAMPOS>);`
- `ALTER TABLE <TABLA> DROP PRIMARY KEY;`
- `ALTER TABLE <TABLA> ADD [CONSTRAINT <NOMBRE>]  
UNIQUE|INDEX(<LISTA_CAMPOS>);`
- `ALTER TABLE <TABLA> DROP INDEX <NOMBRE>;` (Borra Unique e Index)
- `ALTER TABLE <TABLA> ADD [CONSTRAINT <NOMBRE>] FOREIGN  
KEY(<LISTA_CAMPOS>) REFERENCES  
<TABLA_REF>(<LISTA_CAMPOS_REF>);`
- `ALTER TABLE <TABLA> DROP FOREIGN KEY <NOMBRE>;`



```
ALTER TABLE alumno ADD PRIMARY KEY (tipo_documento, numero_documento);  
  
ALTER TABLE alumno ADD FOREIGN KEY (tipo_documento) REFERENCES tipo_documento(tipo);
```