

Agencia de  
Aprendizaje  
a lo largo  
de la vida

# Desarrollo Fullstack



# Les damos la bienvenida

Vamos a comenzar a grabar la clase

## Clase 29

### Node JS

- ▶ Error 404
- ▶ Controladores
- ▶ ENV

## Clase 30

### Base de Datos

- ▶ Introducción
- ▶ MySQL + Motor BBDD
- ▶ Tipos de Datos
- ▶ Diagramas ER

## Clase 31

### Base de Datos

- ▶ DML (SELECT, INSERT, UPDATE, DELETE)
- ▶ Manipulando datos de nuestra base
- ▶ Relaciones + JOINS

# BASES DE DATOS

Fundamentos



# ¿Qué es un dato?

Un **dato** es información que refleja el valor de una característica de un objeto real, sea concreto, abstracto o hasta imaginario.

Son **representaciones** de un determinado **atributo** o **variable** cualitativa o cuantitativa.

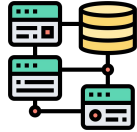
# ¿Qué es una Base de datos?

Es un **conjunto de datos almacenados y organizados** que pueden estar o no relacionados entre sí.

Estos datos pertenecen a un mismo contexto y su almacenamiento surge de la **necesidad de persistencia** de datos, es decir, guardar para usar después.

# Tipos de Bases de Datos

## Relacional



Los datos pertenecen a entidades, donde cada uno puede estar relacionado al dato de otra entidad diferente.

Estas entidades se organizan como un conjunto de tablas con columnas y filas donde las columnas son propiedades de la entidad y las filas se componen de los distintos valores de cada propiedad.

Utilizan el lenguaje SQL.

## No Relacional



Es un enfoque diferente, donde cada entidad es un objeto independiente con sus propiedades, atributos y valores.

No requieren una estructura definida ni muy diseñada ya que nuestros datos no van a estar relacionados entre sí.

Se las conoce como bases de datos NoSQL.

# SQL - Structured Query Language

El lenguaje de base de datos más conocido y utilizado de todos.

Es un lenguaje estándar para el manejo de información desde una base de datos relacional.

Se utiliza a través de un SGBD (Sistema Gestor de Base de Datos) que hacen su propia implementación del lenguaje SQL.



# SQL



# Sistemas Gestores de BBDD

Las bases de datos funcionan mediante gestores de bases de datos, diseñados para almacenar, gestionar y compartir información de manera eficiente.

Estos existen tanto para BBDD relacionales (MySQL, Postgre, SQL Server, MariaDB), como No relacionales (mongoDB, redis), entre otros.



# MySQL

Es un sistema de administración de bases de datos para bases de datos relacionales.

Podemos utilizar MySQL desde:

- CLI: MySQL Shell
- GUI: MySQL Workbench, HeidiSQL, PhpMyAdmin



# Instalamos MySQL y la interfaz gráfica con la que vamos a trabajar.

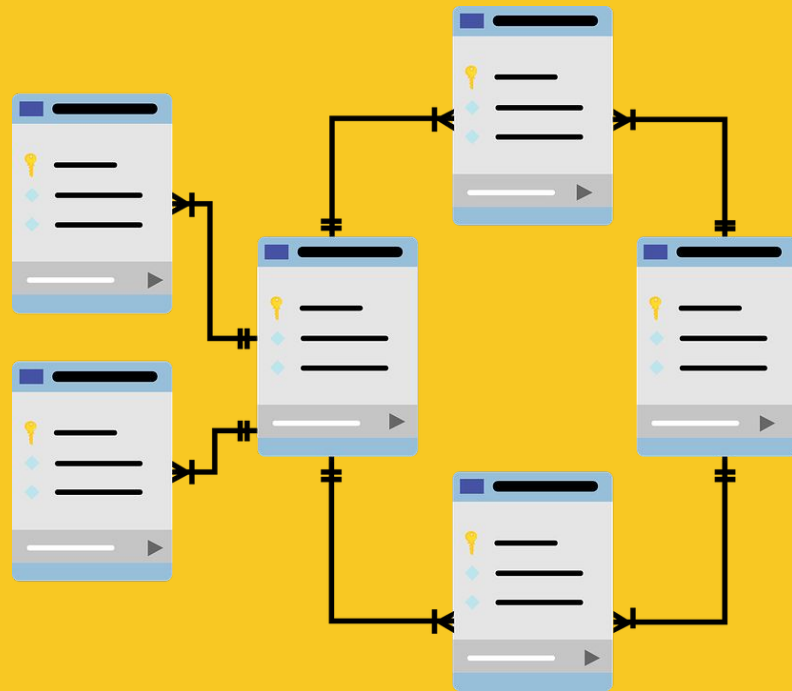
# Conceptos Básicos

Bases de datos relaciones

# Entidad

Las entidades representan cosas u objetos (ya sean reales o abstractos), que se diferencian claramente entre sí.

Estas entidades se representan en diagramas entidad/relación.



# Atributos

Definen o identifican las características de una entidad.

En el siguiente ejemplo tenemos un profesor que posee los atributos cédula, nombres, apellido y nro\_cubículo.



# Tipos de datos Numéricos

Tipo de datos	Rango de Valores	Espacio de almacenamiento
tinyint	0 hasta 255	1 byte
smallint	-32.768 Y 32.767	2 bytes
int	$-2^{31}$ A $2^{31}-1$	4 bytes
bigint	$-2^{63}$ A $2^{63}-1$	8 bytes
decimal (p, s) numeric (p, s)	$-10^{38}$ $1-10^{38}-1$	5 a 17 bytes
smallmoney	-214.748,3648 A 214,748.3647	4 bytes
money	-922.337.203.685.477,5808 A 922,337,203,685,477.5807	8 bytes
real	$-3,4^{38}$ A $-1,18^{38}$ , 0, y $1,18^{38}$ a $3,4^{38}$	4 bytes
FLOAT (n)	$-1,79^{308}$ A $-2,23^{308}$ , 0, y $2,23^{308}$ a $1,79^{308}$	4 bytes u 8 bytes

# Tipos de datos Alfanuméricos

Tipos de Datos Alfanuméricos		
Tipo	Definición	Bytes
Char (n)	Almacena de 1 a 255 caracteres alfanumericos, este valor viene dado por N, y es el tamaño utilizado en disco para almacenar dato. Es decir si definimos un campo como char (255), el tamaño real del campo sera de 255, aunque el valor solo contenga 100.	0 - 255
Varchar (n)	igual que el tipo char, pero este almacena unicamente los bytes que contenga el valor del campo.	0 - 255



# Tipos de datos de Fecha

Tipo de datos	Rango de Valores	Precisión	Espacio de almacenamiento
smalldatetime	01/01/1900 hasta 06/06/2079	1 minuto	4 bytes
datetime	01/01/1753 a 12/31/9999	0,00333 segundos	8 bytes
datetime2	01/01/0001 a 12/31/9999	100 nanosegundos	6 a 8 bytes
datetimeoffset	01/01/0001 a 12/31/9999	100 nanosegundos	8 a 10 bytes
date	01/01/0001 a 12/31/9999	1 día	3 bytes
time	00:00:00.0000000 a 23:59:59.9999999	100 nanosegundos	3 a 5 bytes

# Constraints o Restricciones

Se utilizan para limitar o establecer distintas características de un dato en una tabla.

## NOT NULL

El valor de ese campo no puede ser nulo o vacío.

## UNIQUE

El valor de ese campo no puede repetirse en la misma columna.

## DEFAULT

Valor por defecto si no se ingresa ningún valor.

## PRIMARY KEY

Identifica de forma única cada fila de la tabla.

## FOREIGN KEY

Identifica la relación entre campos de diferentes tablas.

# Lenguaje DDL

El lenguaje de definición de datos es el que se encarga de la modificación de la estructura de los objetos de la base de datos.

Nos permite modificar, borrar o definir las tablas en las que se almacenan los datos de la base de datos mediante las sentencias: **CREATE, ALTER, DROP y TRUNCATE.**

## CREATE

Crear una base de datos, tablas, vistas, etc.

## ALTER

Modificar la estructura, por ejemplo añadir o borrar columnas de una tabla.

## TRUNCATE

Elimina los datos de una tabla pero no la tabla.

## DROP

Eliminar los objetos de la estructura, por ejemplo un índice o una secuencia.

# Comandos Básicos

## Crear Base de Datos:

```
CREATE DATABASE prueba;
```

Puede fallar si la tabla ya existe.

```
CREATE DATABASE IF NOT EXISTS prueba;
```

## Eliminar Base de Datos:

```
DROP DATABASE prueba;
```

```
DROP DATABASE IF EXISTS prueba;
```

Una vez eliminada no se pueden recuperar los datos.

## Seleccionar Base de Datos:

```
USE prueba;
```

## Crear Tabla

```
CREATE TABLE usuarios (  
    id int primary key auto_increment,  
    usuario varchar(12) not null,  
    nombre varchar(15) not null,  
    edad tinyint not null  
);
```

# Comandos Básicos

## Modificar Tabla:

```
ALTER TABLE usuarios CHANGE name  
nombre;
```

```
ALTER TABLE usuarios ADD apellido  
varchar(15);
```

## Vaciar Tabla:

```
TRUNCATE usuarios;
```

## Mostrar Columnas:

```
DESCRIBE usuarios;
```

# No te olvides de dar el presente

## **Recordá:**

- **Revisar la Cartelera de Novedades.**
- **Hacer tus consultas en el Foro.**

**Todo en el Aula Virtual.**

# Gracias