

A graphic on the left side of the slide features four overlapping horizontal bars in purple, orange, yellow, and blue. The text 'Agencia de Aprendizaje a lo largo de la vida' is written across these bars in white. The orange bar has a white arrow pointing to the right.

Agencia de  
Aprendizaje  
a lo largo  
de la vida

# DJANGO

## Clase 18

Base de Datos - 1

# Les damos la bienvenida

Vamos a comenzar a grabar la clase

## Clase 17

### Django: Forms - 2

- Widgets
- Validaciones en formularios (is\_valid, cleaned\_data)
- Errores en formularios
- Mensajes

## Clase 18

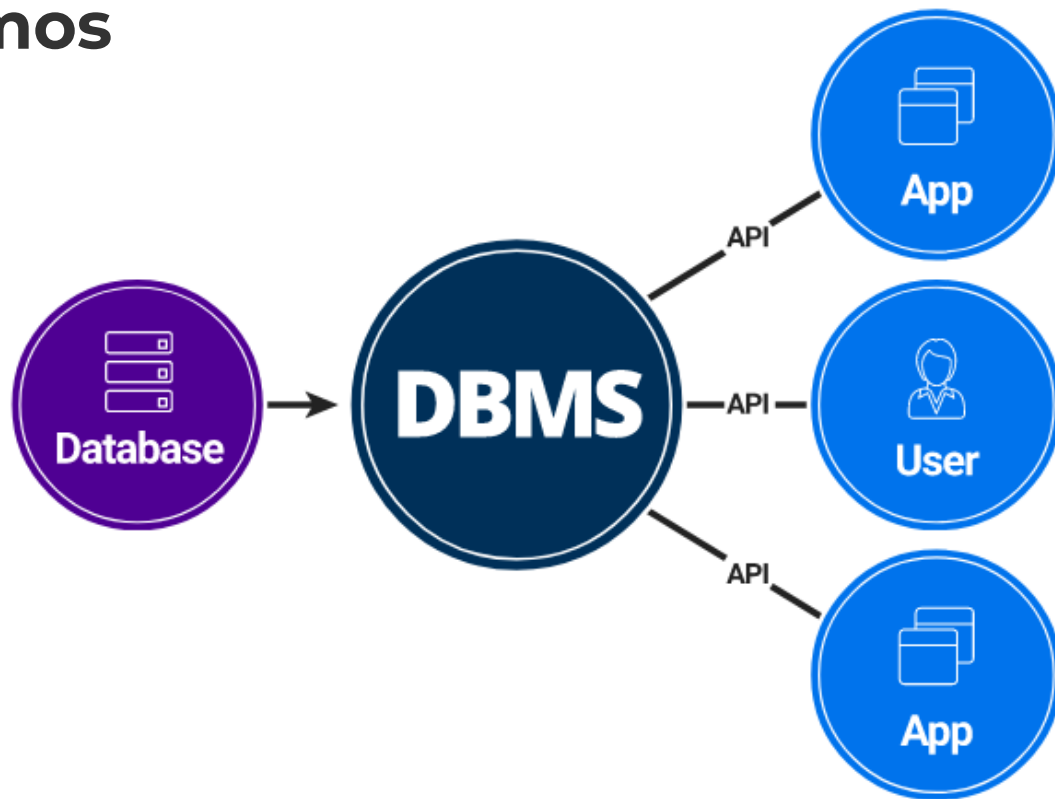
### Django: DB - 1

- Repaso general (relacionales, no relacionales, dbms, db, etc)
- Del modelo de clases al modelo de datos (der)
- PostgreSQL y PgAdmin
- Repaso DDL, DML.
- Repaso claves y relaciones.

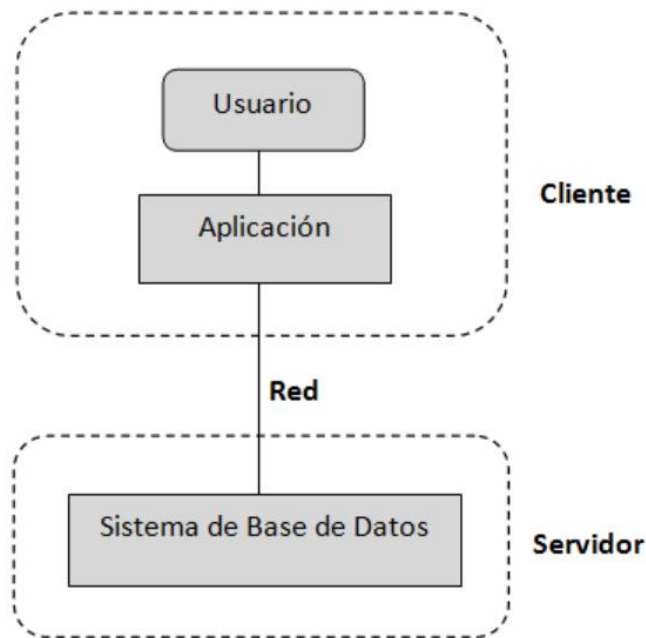
# ¿Qué es un Base de Datos?

Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. Es un forma de almacenar datos de manera eficiente para obtener la información necesaria en el momento que se desee.

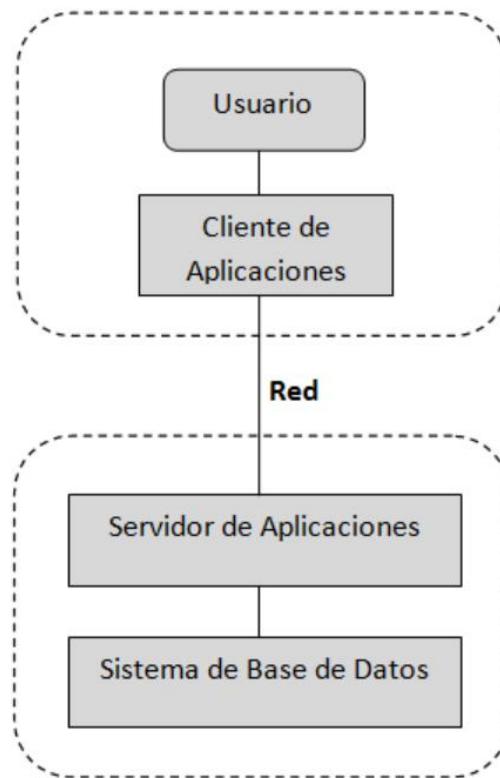
# Recordemos



# Capas



a) Arquitectura de dos capas



b) Arquitectura de tres capas

# SQL vs NoSQL



SQL

VS.



NoSQL



La diferencia fundamental entre ambos tipos de bases de datos radica en que las bases de datos NoSQL no utilizan el modelo relacional.

Debido al largo tiempo que llevan en el mercado, tienen un mayor soporte.

Formada por tablas que contienen campos estructurados

Necesitan más recursos, cuanto más compleja más procesamiento necesitará.

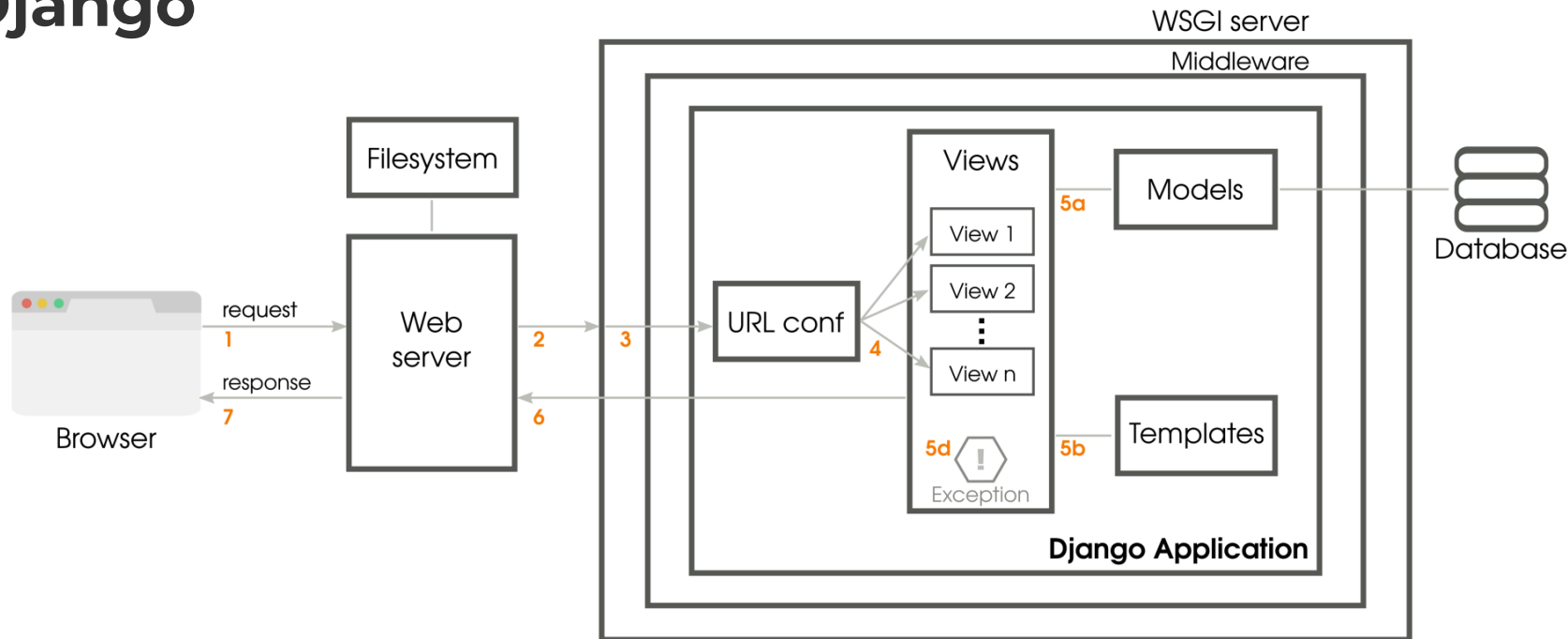
Optimización de consultas para grandes cantidades de datos.

Almacenamiento no estructurado, lo que permite una alta escalabilidad

Abierta y por lo tanto flexible y no necesita tantos recursos para ejecutarse

*Las NoSQL no son un sustituto de las SQL, sino que son una alternativa que ofrece otras posibilidades como el análisis de grandes cantidades de datos. Cuando los datos deben ser consistentes sin dar posibilidad al error se debe usar SQL.*

# Django



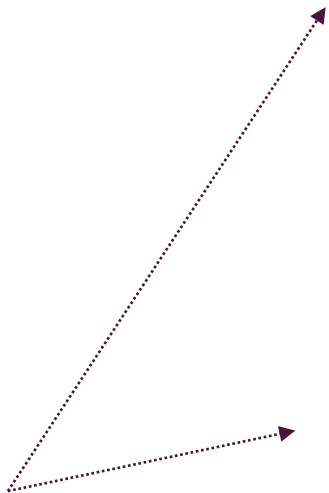


# MODELO DE CLASES VS. MODELO DE DATOS

*¿Cómo mantengo la persistencia de los objetos...?*

*...Mediante una base de datos*

*¿Cómo se puede hacer corresponder un modelo de objetos con un modelo de datos?*



■ En el modelo de clases tenemos:

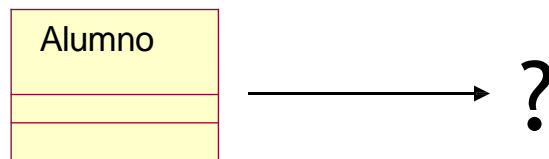
- *Clases*
- *Asociaciones*
- *Generalizaciones*
- *Atributos*
- *Métodos*

■ En el modelo de datos tenemos:

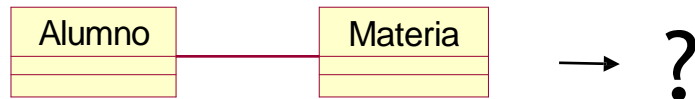
- *Entidades*
- *Relaciones*
- *Atributos*
- *Identificadores*

# ¿Cómo hago las transformaciones?

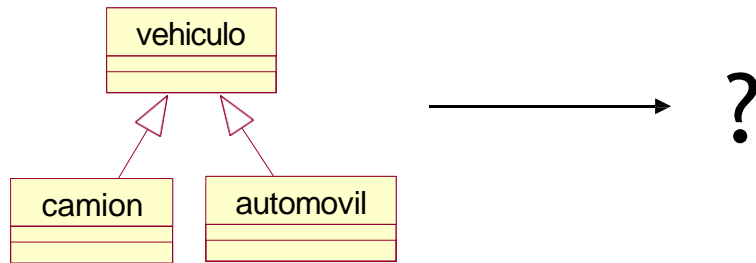
¿Todas las clases se transformarán en entidades?



¿Qué pasará con las asociaciones?

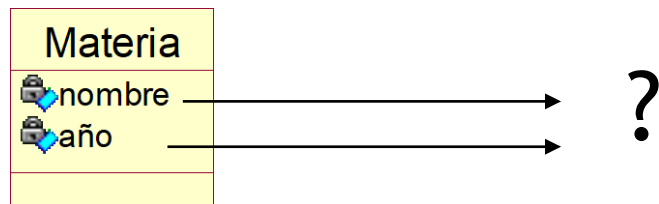


¿Qué pasará con las generalizaciones?



# ¿Cómo hago las transformaciones?

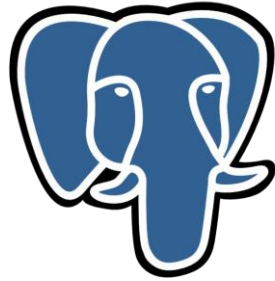
*¿y con los atributos de las clases?*



*¿y con las operaciones?*



# Herramientas que usaremos

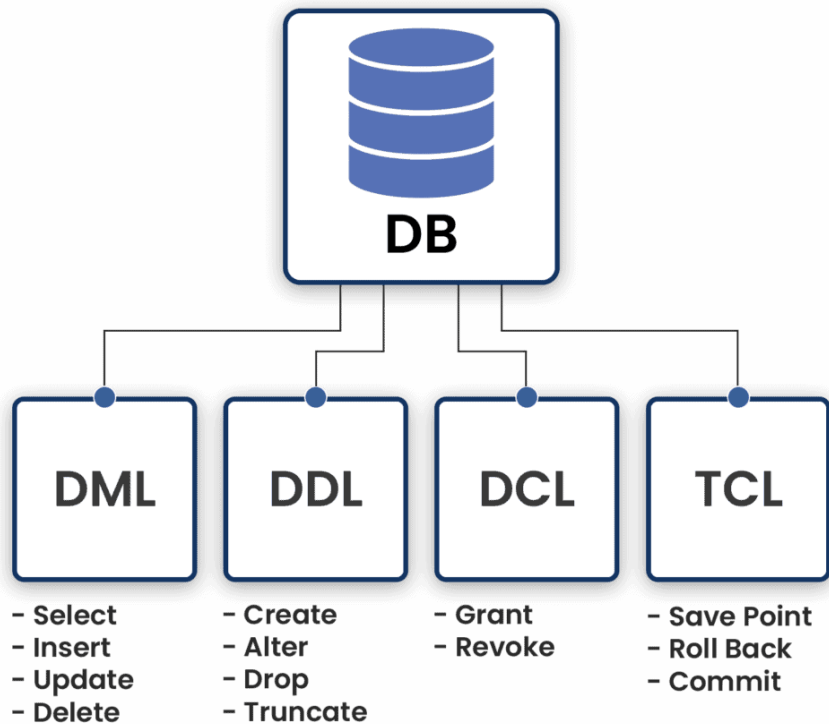


PostgreSQL

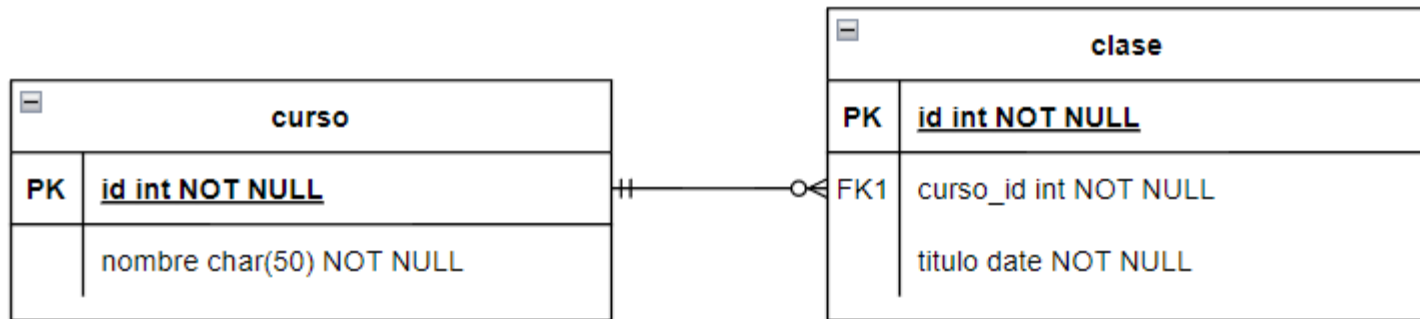


pgAdmin 4

# SQL



# Claves primarias y foráneas



**No te olvides de completar la  
asistencia y consultar dudas**

# Recordá:

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

**TODO EN EL AULA VIRTUAL**