

# Bases de Datos

Clase 1: Motivación

# Información General

- Profesores:
  - Nicolas Alvarado
  - Eduardo Bustos
  - Ariel Reyes
- Clases: W - 5, Talleres: W - 6
- Ayudantías: V - 5 (Se avisan los lunes)
- Web: Syllabus en GitHub

# Ayudantes jefes



## Lucas Fernández

Jefe de cátedra:

- Organizar ayudantías
- Evaluaciones y examen (incluye correcciones)
- Dudas de materia



## Francisca Ancic

Jefe de proyecto:

- Servidores
- Pautas y enunciados
- Correcciones y correcciones
- Dudas de proyecto

# Evaluaciones

## Controles

- Se realizaran 3 controles de materia y un examen
- El examen es obligatorio
- $NC = C1 + C2 + C3 + Ex / 4$

## Proyecto

- El curso tiene un proyecto semestral con tres entregas
- $NP = E1*0.2 + E2*0.4 + E3*0.4$
- Su nota está sujeta a la evaluación de pares

Para aprobar el curso  $NC > 3.95$  y  $NP > 3.95$

¿Por qué este curso?

# Un día cualquiera (8 de marzo)

8:30 am	Despierto, reviso Telegram
---------	----------------------------

11:30 am	Reviso TikTok
----------	---------------

12:00 pm	Se me acabó el pan, me pido un Cornershop
----------	---

12:30 pm	Reviso Hacker News ( <a href="https://news.ycombinator.com">news.ycombinator.com</a> )
----------	--

13:00 pm	Compro almuerzo y lo pago con tarjeta
----------	---------------------------------------

13:30 pm	Actualizo unos documentos en Google Drive
----------	---

14:15 pm	Aburrido, reviso Twitter
----------	--------------------------

15:00 pm	Reviso el precio de mis Bitcoins
----------	----------------------------------

...	...
-----	-----

# Un día cualquiera (8 de marzo)

8:30 am Despierto, reviso **Telegram**

11:30 am Reviso **TikTok**

12:00 pm Se me acabó el pan, **me pido un Cornershop**

12:30 pm Reviso **Hacker News (news.ycombinator.com)**

13:00 pm Compro almuerzo y lo **pago con tarjeta**

13:30 pm Actualizo unos documentos en **Google Drive**

14:15 pm Aburrido, reviso **Twitter**

15:00 pm Reviso el precio de mis **Bitcoins**

...

...

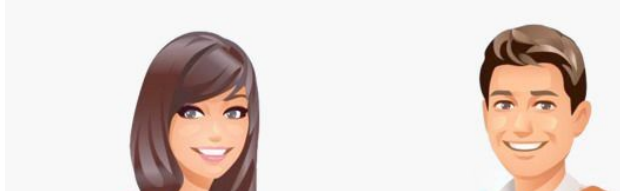
# Un día cualquiera (8 de marzo)

Todas estas actividades involucran alguna base de datos:

- Búsquedas en la web
- Datos públicos
- Redes sociales
- Métodos de pago
- Criptomonedas



Donde sea que trabajen,  
tendrán que interactuar con  
bases de datos



Oiga pero eso es tan noventero... Hoy día las bases de datos ya no se usan, lo que la lleva es

**Big Data**

**Cloud Computing**

**Blockchain**

**quantum computing**

**Microservices**

**Buzzword**

**Deep Learning**

**Internet of Things**

**Data Science**

Donde sea que trabajen,  
tendrán que interactuar con  
bases de datos

La competencia más buscada en trabajos de ciencia de datos?

¡Saber bases de datos!  
modelo relacional, SQL, noSQL

# Outline

- Qué son las bases de datos
- Por qué usar sistemas de bases de datos
- Roadmap del curso

# Bases de Datos

Necesitamos una forma de almacenar los datos



- Para poder procesarlos de forma eficiente
- Que no haya que programar desde 0
- Que sea portable y estándar

# Sistemas de Bases de Datos

Sistema de gestión de bases de datos (Database Management System - **DBMS**)

- Programa que facilite la lectura y almacenamiento de grandes volúmenes de datos.



# Sistemas de Bases de Datos



- Datos se almacenan en disco
- Pero los usuarios interactúan con una **capa lógica** (ej. tablas), fácil de acceder

ID Actor	Nombre Actor
1	Leonardo DiCaprio
2	Matthew McConaughey
3	Daniel Radcliffe
4	Jessica Chastain
...	...

ID Película	Nombre Película
1	Interstellar
2	The Revenant
3	Harry Potter
4	The Wolf of Wall Street
...	...

ID Actor	ID Película
1	2
1	4
2	1
3	3
...	...

# Cómo funciona un DBMS



¿Cuál es la mejor película de Christopher Nolan?



# Cómo funciona un DBMS



¿Cuál es la mejor película de Christopher Nolan?



```
SELECT movies,  
rating FROM Movies
```



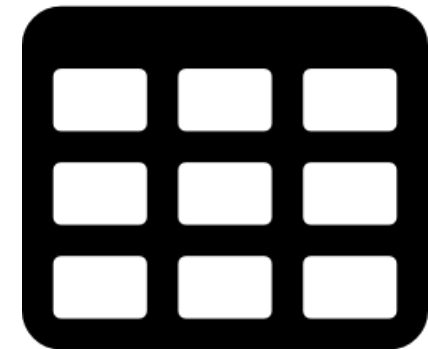
# Cómo funciona un DBMS



¿Cuál es la mejor película de Christopher Nolan?



```
SELECT movies,  
rating FROM Movies
```



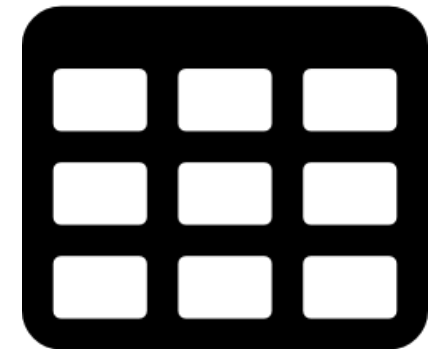
# Cómo funciona un DBMS



¿Cuál es la mejor película de Christopher Nolan?



```
SELECT movies,  
rating FROM Movies
```



Interstellar

# Por qué usar DBMS

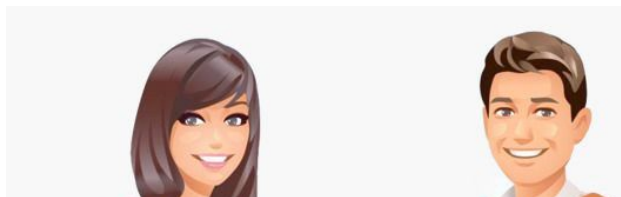
- Almacenar datos (insertar)
- Encontrar datos (búsquedas y consultas)
- Modificar datos (update)
- Asegurar la consistencia de los datos
- Seguridad y privacidad de los datos

En este curso usaremos:

- PostgreSQL

# Por qué usar DBMS

- Almacenar datos (insertar)
- Encontrar datos (búsquedas y consultas)
- Modificar datos (update)
- Asegurar la consistencia de los datos
- Seguridad y privacidad de los datos



Pero podríamos programar todo esto en python y nos ahorramos aprender?

Ah claro, pero tendríamos que trabajar mucho resolviendo cosas que ya han sido resueltas por otros más talentosos que nosotros.



En este curso trabajaremos  
principalmente con bases de  
datos relacionales

Lo primero, es aprender a  
cómo funciona el modelo  
relacional

# Modelo Relacional

El modelo de las bases de datos relacionales se basa en:

- Tablas (relaciones)
- Columnas de las tablas (atributos con sus tipos)
- Filas de las tablas (tuplas) que contienen los datos

# Lenguajes de Consultas

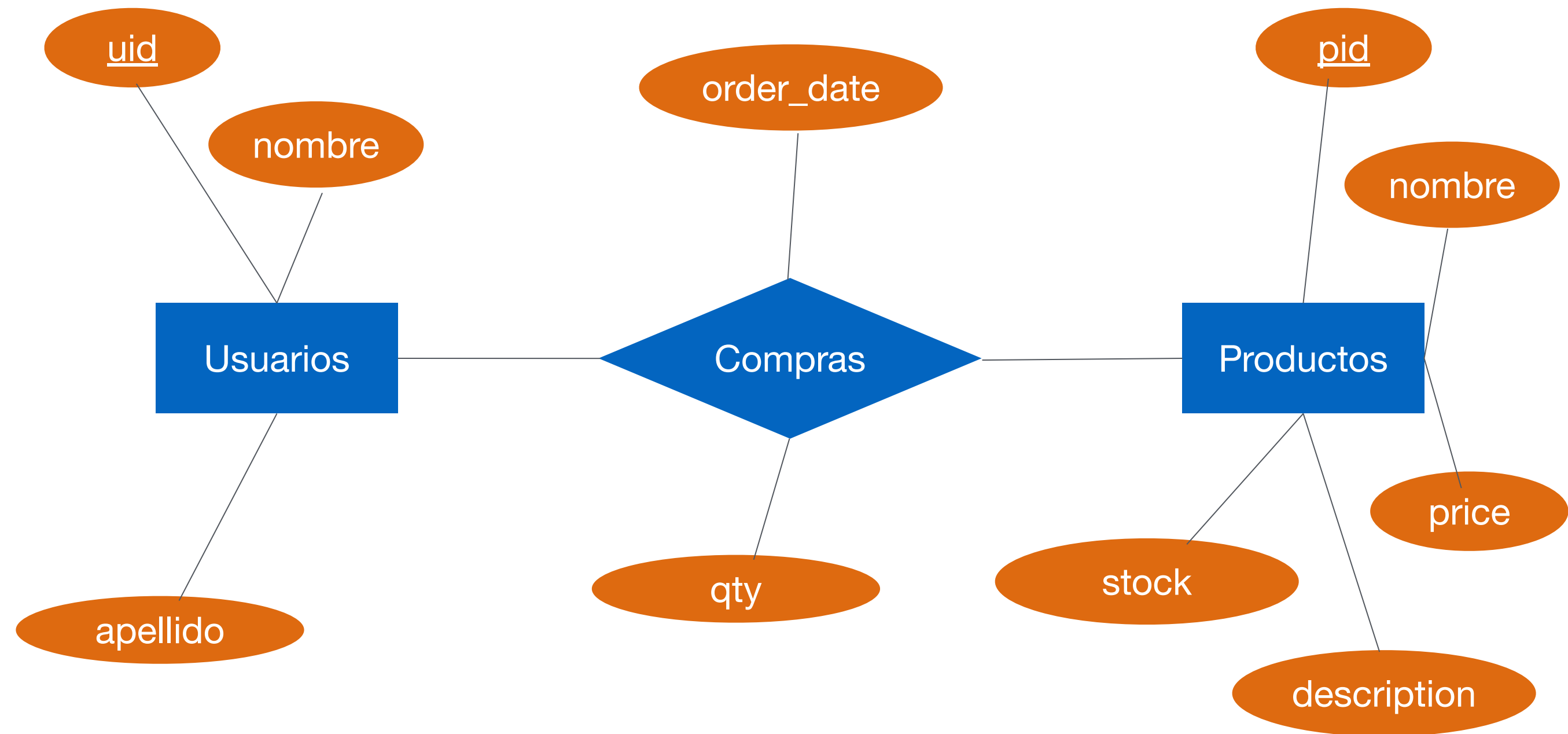
En este curso aprenderemos el lenguaje de consultas del modelo relacional: **SQL**

También aprenderemos el lenguaje teórico sobre el que se sustenta: **álgebra relacional**

Pero también hablaremos de otros sistemas fuera del modelo relacional

Y lo más importante...  
Aprenderemos a modelar los  
problemas

# Modelo Relacional



Y también aprenderemos  
conceptos avanzados sobre el  
lenguaje SQL

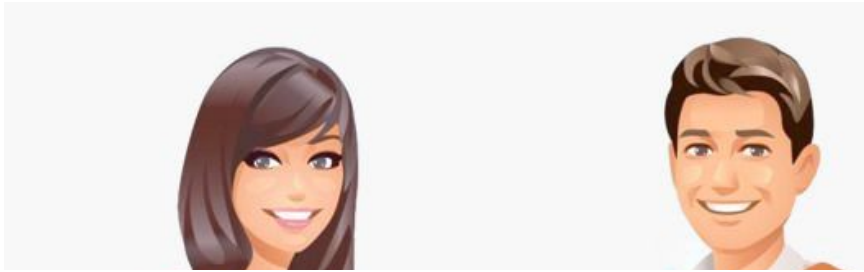
# En este curso

Aprenderemos a realizar consultas complejas.

Aprenderemos como el DBMS ejecuta una consulta.

Y varias cosas más fundamentales para el uso práctico de un DBMS





¿Y qué hay de mi aplicación Web?

Las aplicaciones web suelen conectarse a algún sistema de bases de datos.

# SQL + Programación

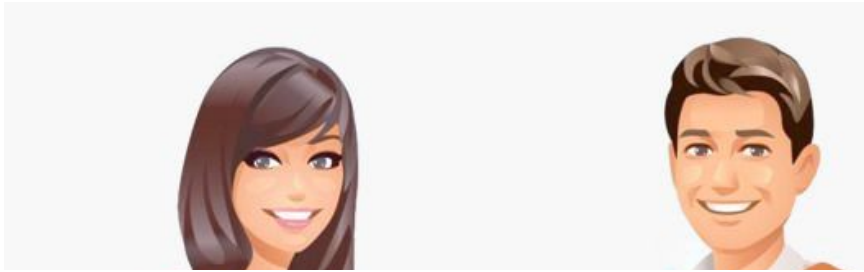
Es posible conectar un DBMS a los lenguajes de programación (Python, Java, C#, etc) y consultar la base de datos de forma programática.

Las aplicaciones que comúnmente usamos, se conectan a una base de datos (muchas de ellas, una base de datos SQL) .

# En este curso

Aprenderemos cómo consumir una base de datos desde un lenguaje de programación

En el proyecto, construiremos una aplicación web que hace uso de sistemas de bases de datos



Pero nos contaron que si usábamos un  
framework Web no hay que saber usar SQL

Quizás es posible, pero también es posible hacer una aplicación web en una interfaz gráfica sin saber cómo funcionan.

[Funciones](#)[Explora](#)[Suscripciones](#) [Plantillas](#)[Ayuda](#)[Entrar](#)

# Todo empieza con tu increíble página web

Wix combina la belleza con la más avanzada tecnología para crear tu increíble página web. Es fácil y gratis.

[Empieza ya](#)

De la misma forma que es posible  
armar un puente sin saber cómo  
hacerlo





# En este curso además...

Aprenderemos nociones de cómo funciona un sistema de bases de datos por dentro

- Índices
- Algoritmos internos
- Transacciones
- Recuperación de fallas

# ¿Es SQL suficiente?

Un sistema puede funcionar lento porque:

- La base de datos está mal indexada
- Los datos están mal modelados
- Hay una cantidad de datos demasiado grande
- ...

# ¿Es SQL suficiente?

Aplicaciones grandes, como una red social de millones de usuarios, o un laboratorio de millones de registros, necesita técnicas diferentes

Pero ojo, una aplicación típica puede funcionar perfecto con una base de datos SQL!

# En este curso

Aprenderemos tópicos sobre bases de datos NoSQL

- JSON y MongoDB
- Búsqueda de texto
- Herramientas de análisis de datos

Y lo más importante, cuando es pertinente usar cada una de estas tecnologías

¿Preguntas?

# Modelo relacional

# DBMS

DBMS relacionales Open Source

- PostgreSQL - Es la que usaremos nosotros
- MySQL - Usada ampliamente en ambientes de producción
- SQLite - Base de datos pequeña, usada generalmente en contextos de apps móviles



# DBMS

DBMS relacionales comerciales

- IBM DB2
- Microsoft SQL Server
- Oracle

# DBMS

## Otros DBMS

- Neo4J (Grafos)
- MongoDB (Documentos)
- Cassandra (Key Value - Column Store)
- Apache Jena (RDF)
- Redis (In memory Store)
- Base (Column Store)
- Titan DB (Grafos)
- ...

Modelos de datos

# Modelo de Datos

- Un Modelo es una notación para escribir datos
- En este curso se verán en detalles dos modelos:
  - Relacional
  - Semiestructurados (key-value, grafos)
- Modelo relacional es el modelo más usado en ambientes de producción, pero la necesidad de utilizar modelos semiestructurados ha ido aumentando en el tiempo.

# Modelo Relacional

Almacenar datos en tablas:

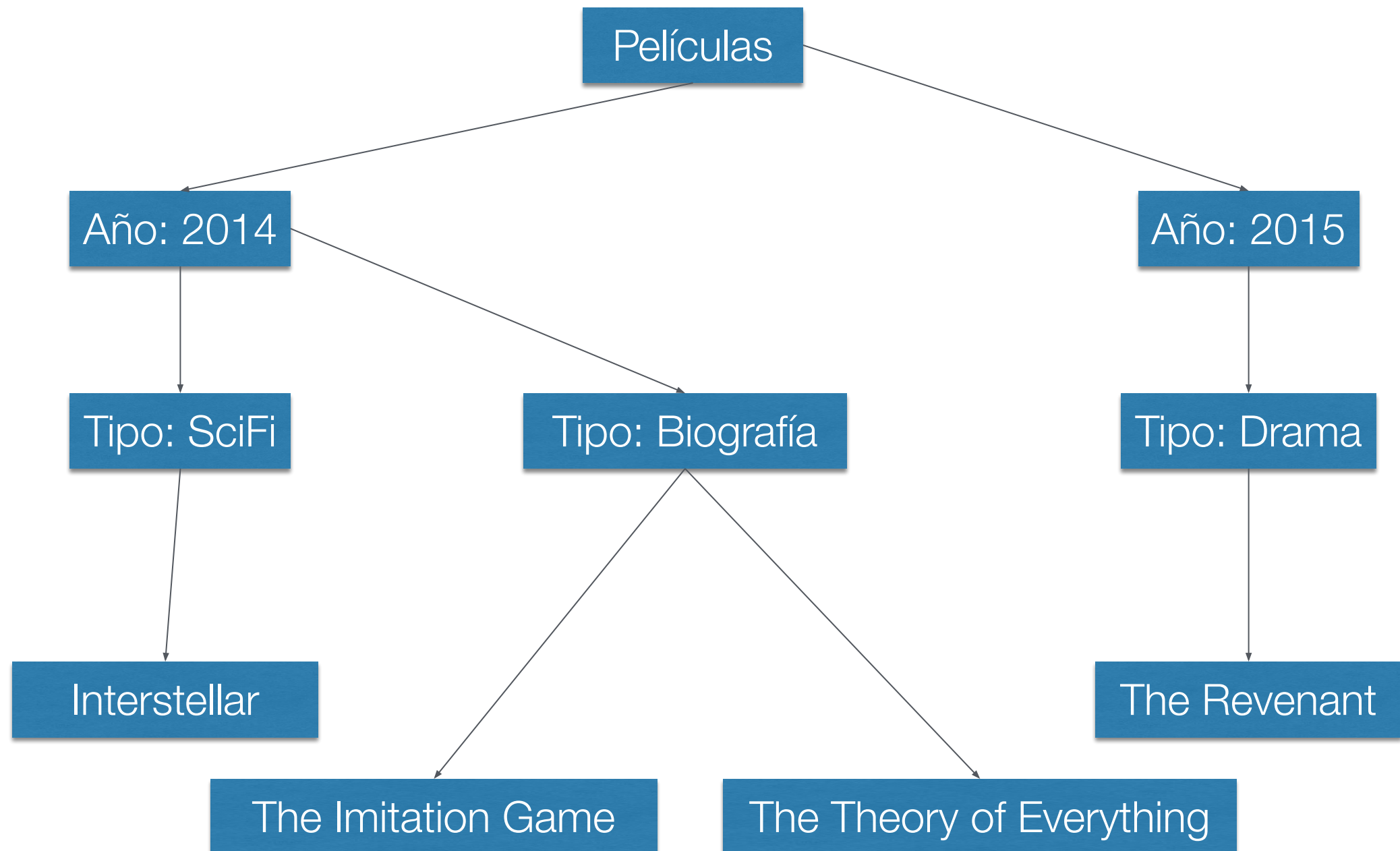
ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

# Modelo Relacional

- Aunque parecen arreglos o listas (de Python por ejemplo) existen muchas diferencias
- Generalmente, se asume que están en Disco y no en Memoria
- No podemos hacer todo lo que queramos con esto
- Vamos a ver como manejar estos datos:
  - SQL
  - Álgebra Relacional

# Datos Semiestructurados

Estructura Jerárquica



# Datos Semiestructurados

## XML

```
<Películas>
  <Año valor="2014">
    <Tipo valor="Biografía">
      <Película nombre="The Imitation Game" calificación="8.1">
      </Película>
      <Película nombre="The Theory of Everything" calificación="7.7">
      </Película>
    </Tipo>
    <Tipo valor="SciFi">
      <Película nombre="Interstellar" calificación="8.6">
      </Película>
    </Tipo>
  </Año>
  <Año valor="2015">
    <Tipo valor="Drama">
      <Película nombre="The Revenant" calificación="8.1">
      </Película>
    </Tipo>
  </Año>
</Películas>
```



# Datos Semiestructurados

Key - Value

```
{
  "2014": {
    "Biografía": [
      { "nombre": "The Imitation Game", "calificación": 8.1 },
      { "nombre": "The Theory of Everything", "calificación": 7.7 }
    ],
    "SciFi": [
      { "nombre": "Interstellar", "calificación": 8.6 }
    ]
  },
  "2015": {
    "Drama": [
      { "nombre": "The Revenant", "calificación": 8.1 }
    ]
  }
}
```

# Comparación

Ambos:

- Proveen solución para almacenar datos
- Son versátiles para modelar
- Ambos tienen lenguaje de consultas

Pero:

- Modelo relacional está definido por un **esquema**
- XML es más flexible, no está separado por un esquema

El modelo relacional al ser menos flexible es más simple pero también limitado

# Otros Ejemplos

- Bases de Datos orientados a objetos.
- Bases de Datos columnares.
- Bases de Datos de Grafos.

Tenemos que aprender las diferencias y  
cuándo usar qué!

# Modelo Relacional

# Modelo Relacional

Los datos se almacenan como tablas:

Películas

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

Distinguimos:

- **Relaciones:** a cada tabla le llamamos relación
- **Atributos:** son las columnas de la relación
- **Tuplas:** son las filas de la relación

# Modelo Relacional

## Películas

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

- En este caso tenemos la relación **Películas**
- Los atributos de la relación **Películas** son **ID\_Película**, **Nombre\_Película**, **Año**, **Categoría** y **Calificación (IMDB)**.

# Modelo Relacional

## Esquema

Un esquema es un conjunto de relaciones con sus atributos:

Películas(id, nombre, año, categoría, calificación)

Actor(id, nombre, edad)

Actuó\_en(id\_actor, id\_película)

# Modelo Relacional

## Esquema

Para denominar relaciones escribimos su nombre y luego sus atributos entre paréntesis:

**Películas(id, nombre, año, categoría, calificación)**



# Modelo Relacional

## Dominio

En la práctica, cada atributo tiene un dominio o tipo de dato (float, integer, string, date, ...)

Películas(id: **int**, nombre: **string**, año: **int**,  
categoría: **string**, calificación: **float**)

# Modelo Relacional

Instancia

Una **instancia** de un **esquema** es un conjunto de tuplas para cada relación del esquema

Esto es un esquema:

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
-------------	-----------------	-----	-----------	---------------------

# Modelo Relacional

## Instancia

Una **instancia** de un **esquema** es un conjunto de tuplas para cada relación del esquema

Esto es una instancia:

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	The Revenant	2015	Drama	8.1
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7

# Modelo Relacional

## Restricciones de integridad

Son restricciones que imponemos a un esquema que todas las instancias deben satisfacer

La restricción más importante son las **llaves**

Un conjunto de atributos forma una **llave** en una **relación** si no permitimos que existan dos tuplas para esa relación con los mismos valores en todos los atributos de la llave, y no hay un subconjunto de esos atributos que cumpla esa condición.

# Modelo Relacional

Ejemplo: Llaves

¿Cuál es la llave?

ID Película	Nombre Película	Año	Categoría	Calificación (IMDB)
1	Interstellar	2014	Fantasía	8.6
2	Batman	2005	Acción	8.3
3	The Imitation Game	2014	Biografía	8.1
4	The Theory of Everything	2014	Biografía	7.7
5	Batman	1995	Acción	5.4

Nota:

- Batman Begins (2005)
- Batman Forever (1995)

# Modelo Relacional

## Llaves

Cuando escribimos las relaciones subrayamos las llaves

Películas(id, nombre, año, categoría, calificación)

# Llaves

## Terminología

**Super llave (superkey):** cualquier conjunto de atributos que determina a todo el resto

**Llave (candidata/minimal):** cualquier conjunto de atributos que determina a todo el resto, y ninguno de sus subconjuntos es una super llave

**Llave primaria:** una llave candidata que queremos destacar (la subrayada en el esquema)

# Llaves

Ejemplo

**Persona(id, rut, nombre)**

**Llave primaria:** id



# Llaves

Ejemplo

**Persona(id, rut, nombre)**

**Llave primaria:** id

**Llaves candidatas:**

- id
- rut

# Llaves

Ejemplo

**Persona(id, rut, nombre)**

**Llave primaria:** id

**Llaves candidatas:**

- id
- rut

**Superllaves:**

- id
- rut
- id,rut
- id,nombre
- rut,nombre
- id,rut,nombre

# Surrogate Key

**Persona(id, rut, nombre)**

**Surrogate key:** una llave genérica que simplifica cosas

- **id**

En nuestro ejemplo id es mas fácil de manejar que rut

# Cómo consultar bases de datos

# Consideremos la siguiente instancia

actores

id	nombre	edad
1	Leonardo DiCaprio	41
2	Matthew McConaughey	46
3	Daniel Radcliffe	27
4	Jessica Chastain	39
...	...	...

actuo\_en

id_actor	id_pelicula
1	2
2	1
4	1
3	3
1	5
...	...

películas

id	nombre	año	categoría	calificación	director
1	Interstellar	2014	SciFi	8.6	C. Nolan
2	The Revenant	2015	Drama	8.1	A. Iñárritu
3	Harry Potter	2011	Fantasía	8.1	D. Yates
4	The Theory of Everything	2014	Biografía	7.7	J. Marsh
5	Inception	2010	Adventure	8.8	C. Nolan
...	...	...	...	...	...

# Ejemplos de Consultas

## Ejemplo 1

Liste el nombre de todos los actores

nombre
Leonardo DiCaprio
Matthew McConaughey
Daniel Radcliffe
Jessica Chastain

# Ejemplos de Consultas

## Ejemplo 2

Liste el nombre y la calificación de todas las películas

nombre	calificacion
Interstellar	8.6
The Revenant	8.1
Harry Potter	8.1
The Theory of Everything	7.7
Inception	8.8

# Ejemplos de Consultas

## Ejemplo 3

Liste el nombre y la calificación de todas las películas con calificación inferior a 8.5

nombre	calificacion
The Revenant	8.1
Harry Potter	8.1
The Theory of Everything	7.7



# Ejemplos de Consultas

## Ejemplo 4

Liste todas las películas de Nolan

id	nombre	año	categoria	calificacion	director
1	Interstellar	2014	SciFi	8.6	C. Nolan
5	Inception	2010	Adventure	8.8	C. Nolan

# Ejemplos de Consultas

## Ejemplo 5

Liste todos los id de los actores de la película “Interstellar”

id
2
4

# Ejemplos de Consultas

## Ejemplo 6

Liste cada actor junto a todas las películas en las que ha actuado

id	nombre	nombre_pelicula
1	Leonardo DiCaprio	The Revenant
1	Leonardo DiCaprio	Inception
2	Matthew McConaughey	Interstellar
3	Daniel Radcliffe	Harry Potter
4	Jessica Chastain	Interstellar

# Ejemplos de Consultas

## Ejemplo 7

Liste todas las películas en que actúe Leonardo DiCaprio y que sean dirigidas por C. Nolan

nombre
Inception

# Ejemplos de Consultas

## Ejemplo 8

Liste todas las películas y la calificación en que actúe Leonardo DiCaprio o que sean dirigidas por C. Nolan

nombre	calificacion
Interstellar	8.6
The Revenant	8.1
Inception	8.8

# Ejemplos de Consultas

## Ejemplo 9

Liste el nombre de todos los actores y directores

nombre
Leonardo DiCaprio
Matthew McConaughey
Daniel Radcliffe
Jessica Chastain
C. Nolan
A. Iñárritu
D. Yates
J. Marsh

# Ejemplos de Consultas

## Ejemplo 10

Liste el nombre de todos los actores dirigidos por C. Nolan y A. Iñárritu

id	nombre	edad
1	Leonardo DiCaprio	41

# Ejemplos de Consultas

¿Qué podemos concluir?

- Los resultados de las consultas también son tablas
- Parecen haber operaciones en común