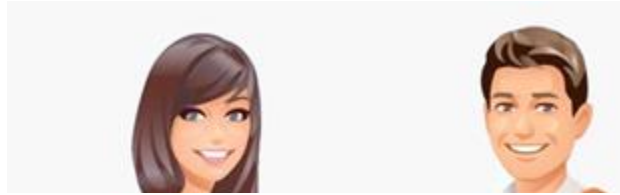


WHY DATABASE COURSE





**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

Un día cualquiera (6 de agosto)

| | |
|----------|---|
| 8:30 am | Despierto, reviso Telegram |
| 11:30 am | Reviso TikTok |
| 12:00 pm | Se me acabó el pan, me pido un Ubereats |
| 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 X |
| 14:30 pm | Reviso la sala de Bases de datos Buscacursos |

...

...

Un día cualquiera

Todas estas actividades involucran alguna base de datos:

- **Búsquedas en la web**
- **Datos públicos**
- **Redes sociales**
- **Métodos de pago**
- **Bases privadas**

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

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 físico**
- **Pero los usuarios interactúan con una capa lógica (ej. tablas), fácil de acceder**

Como funciona un DBMS

Queremos saber cuál es la mejor película de C. Nolan



Pregunta

¿Cuál es la mejor película de Christopher Nolan?



SELECT movies,
rating FROM Movies

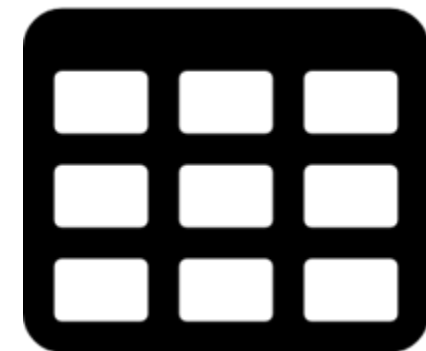
Lenguaje de
consulta



Servidor



Base de datos



Respuesta

Interstellar

| ID Película | Nombre Película | Año | Director | Calificación (IMDB) |
|-------------|--------------------------|------|-------------|---------------------|
| 1 | Interstellar | 2014 | C. Nolan | 8.6 |
| 2 | The Revenant | 2015 | A. Iñárritu | 8.1 |
| 3 | The Imitation Game | 2014 | M. Tyldum | 8.1 |
| 4 | The Theory of Everything | 2014 | J. Marsh | 7.7 |
| 5 | Oppenheimer | 2023 | C. Nolan | 8.3 |

¿Por qué usar DBMS?

Podríamos programar todo en Python y nos ahorramos este curso.



Deberíamos:

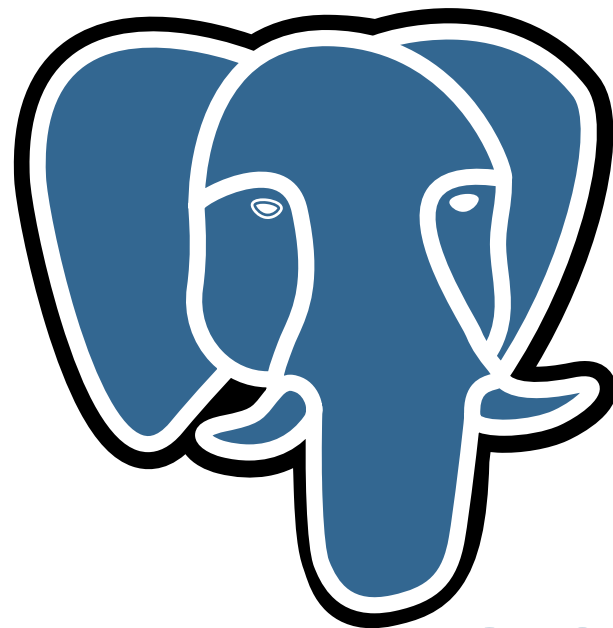
- **Almacenar y organizar los datos (insertar)**
- **Encontrar datos (búsquedas y consultas)**
- **Modificar datos (update)**
- **Asegurar la consistencia de los datos**

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

“Take the best and do the rest”

En este curso trabajaremos principalmente con bases de datos relacionales

(Pero también hablaremos de otros sistemas fuera del modelo relacional)



PostgreSQL

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**

En modelo relacional fue desarrollado por [Edgar F. Codd](#)

Lenguajes de

Consultas

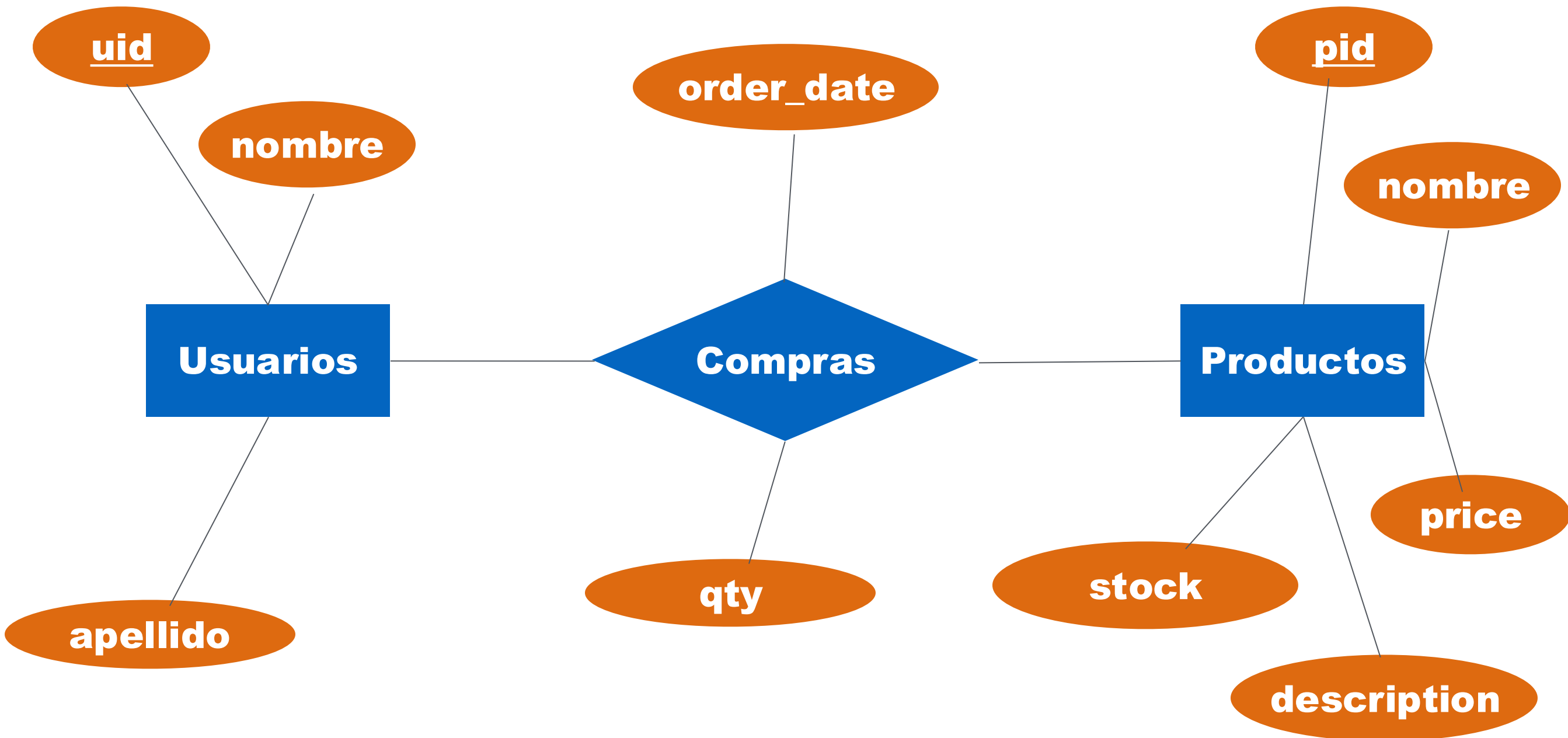
En este curso aprenderemos el lenguaje de consultas del modelo relacional SQL desarrollado inicialmente en IBM por Donald D. Chamberlin y Raymond F. Boyce en la década de 1970.



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

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

Modelo Entidad Relación





¿Y las

**aplicaciones web o
móviles suelen
conectarse a algún
sistema de bases de
datos.**

SQL + Programación

Es posible conectar un DBMS a los lenguajes de programación (PHP, 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) .

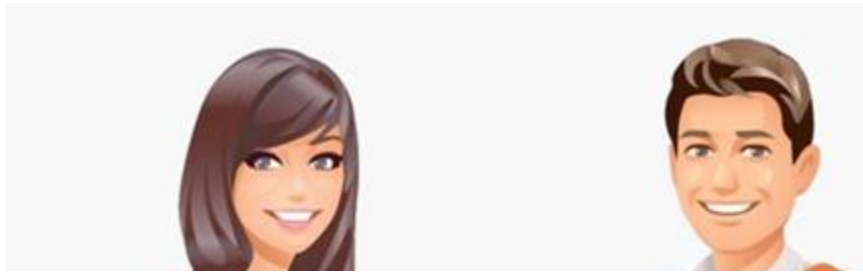
En este curso

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



En el proyecto, construirán 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. Efectivamente, es posible abstraerse de la Base de datos y de SQL con Frameworks y ORM.

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



En este curso además...

Aprenderán nociones de cómo funciona un sistema administrador de bases de datos por dentro:

- **Índices**
- **Algoritmos internos**
- **Transacciones**
- **Recuperación de fallas**
- **Rendimiento**

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**
- **...**

¿Con SQL resolvemos todos los tipos de problema?

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

Aprenderán tópicos sobre bases de datos NoSQL

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

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

Estado del Arte

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 relacionales comerciales

- Oracle Database
- Microsoft SQL Server
- IBM Db2
- Teradata
- Amazon Aurora (versión comercial de MySQL y PostgreSQL)
- Microfocus Vertica
- IBM Informix
- IBM Performance server Netezza
- SAP Sybase ASE
- Actian Ingres / Actian X
- MariaDB ClustrixDB

Otros DBMS

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

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

Almacenar datos en tablas:

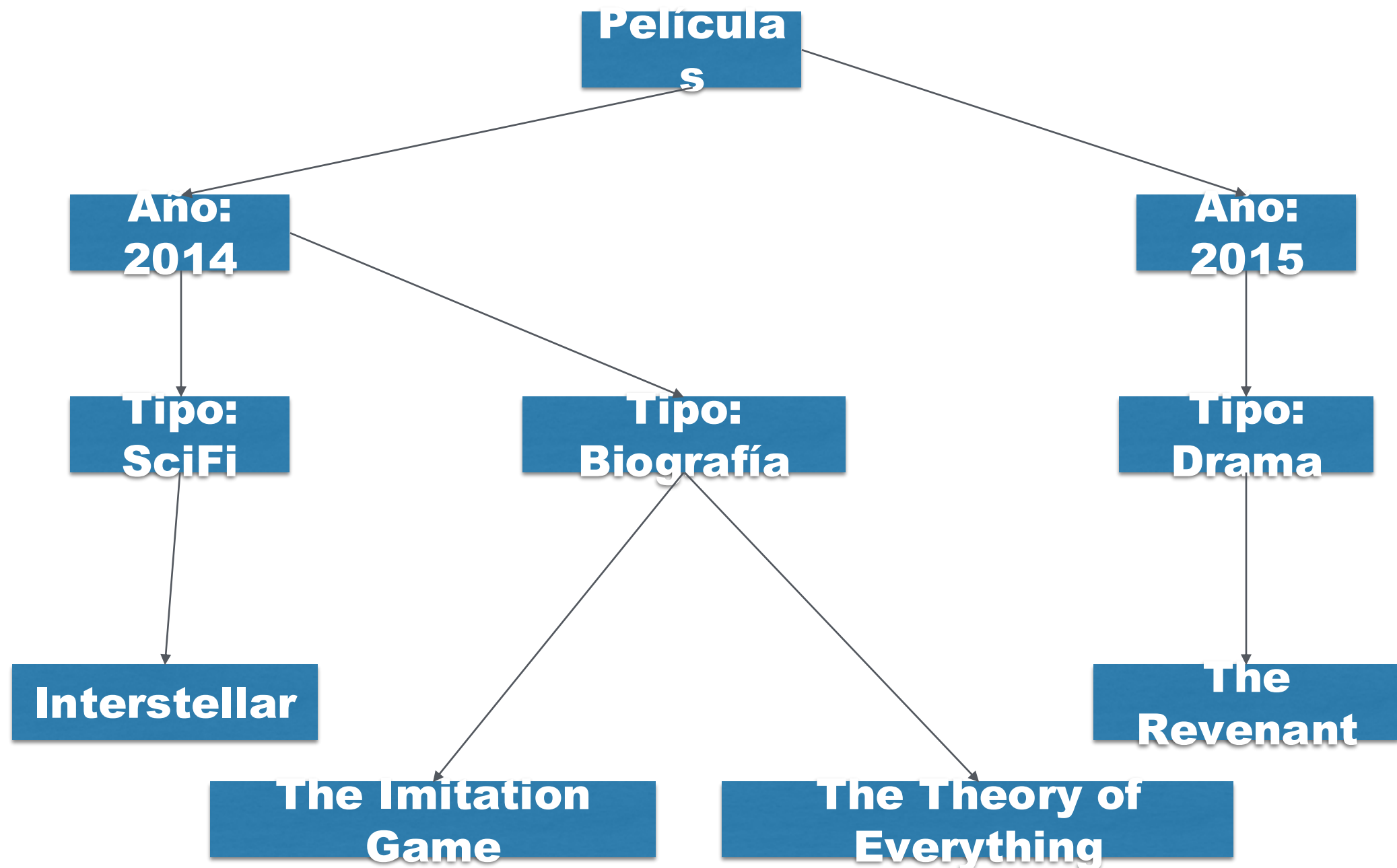
| ID Película | Nombre Película | Año | Categoría | Calificación (IMDB) |
|-------------|--------------------------|------|-----------|---------------------|
| 1 | Interstellar | 2014 | Fantasía | 8.6 |
| 2 | The Revenant | 2015 | Drama | 8.1 |
| 3 | The Imitation Game | 2014 | Biografía | 8.1 |
| 4 | 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**

Estructura Jerárquica



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>
```


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!**

Conclusión

- **Para almacenar y utilizar colecciones de datos ordenados existen las bases de datos.**
- **Varios modelos**
 - **desde estructurados y poco flexibles**
 - **Modelo relacional**
 - **a más flexibles con menos estructuras**
 - **Jerárquica**
 - **XML**
 - **Key value**
 - **Json**
- **Para manejar bases de datos se usa un “motor” o SABD (DBMS)**
 - **Mysql/postgresql/MaríaDB**
 - **Oracle**
 - **Mysql**
 - **DB2**
 - **MongoDB**
 - **....**