



## Fastbook 03

# Tratamiento de Datos (Excel y SQL)

Introducción a las bases de datos



## 03. Introducción a las bases de datos



En este fastbook, conoceremos el significado de las bases de datos y las ventajas que nos ofrecen. Además, echaremos un breve vistazo por los distintos tipos de bases de datos que existen actualmente, entenderemos cuáles son los principales beneficios que nos aportan y aprenderemos cuándo debemos usarlas.

Haremos hincapié en soluciones cloud desarrolladas para adaptar el contexto de las bases de datos al mundo del big data, y veremos que, aunque no puedan considerarse como bases de datos, nos permitirán realizar casi las mismas operaciones sobre una cantidad inmensa de información y a un precio muy bajo.

*Autor: Carlos Manchón y Breogán Cid*

**Introducción a las bases de datos**

**Motivación**

**Tipos de bases de datos**

**Bases de datos no relacionales**

**Bases de datos documentales**

**Bases de datos de grafos**

**Bases de datos clave/valor**

**Bases de datos columnares**

**Bases de datos cloud**

**Conclusiones**

# Introducción a las bases de datos

**X** Edix Educación

---

Antes de adentrarnos en el maravilloso mundo de las bases de datos, tenemos que entender qué es una base de datos.

## Según la RAE

---

Una base de datos es un conjunto de datos organizado de tal modo que permite obtener con rapidez diversos tipos de información.

## Según Oracle

---

Es una recopilación organizada de información o datos estructurados que normalmente se almacena, de forma electrónica, en un sistema informático.

## Según Microsoft

---

Es una herramienta para recopilar y organizar información. Las bases de datos pueden almacenar información sobre personas, productos, pedidos u otras cosas.

Como vemos, los límites del concepto ‘base de datos’ son un poco difusos. Hemos realizado una pequeña búsqueda de las definiciones aportadas por grandes compañías y, aunque en todas ronda la misma idea, encontramos pequeños matices que las diferencian.

---

## Entonces, ¿a qué nos referimos cuando estamos hablando de una base de datos?

---

Si somos puristas y nos ceñimos al sentido estricto de la palabra, una base de datos es un **conjunto estructurado** de datos que representa a entidades (personas, cosas u objetos) y a sus interrelaciones.

---

## Si esto es así, ¿podemos considerar Microsoft Excel como una base de datos?

A pesar de que la información se distribuye en tablas y estas **se estructuran en columnas**, como sucede con las tablas de una base de datos (de carácter relacional), Microsoft Excel no es una base de datos, aunque mucha gente lo use como si así lo fuera.

---

Excel es una **hoja de cálculo** que puede servir como plantilla inicial para crear los datos que se quieran insertar en la base de datos real, pero para ser considerado como base de datos requiere de un software especial (SGBD) que gestione y administre esos datos.

---

## Vale, entonces... ¿Qué es un sistema gestor de base de datos (SGBD)?

---

SGBD es el **software** encargado de administrar una base de datos.

Algunos ejemplos conocidos podrían ser MySQL, SQL Server o PostgreSQL que, a pesar de ser conocidos como bases de datos, no lo son; más bien son sistemas gestores de base de datos que se encargan de administrar la base de datos que queremos crear, modificar, consultar, etc.

Normalmente, cuando se habla de base de datos, realmente se hace referencia al sistema gestor de base de datos.

---

**Para que nos entendamos: cuando normalmente hablamos de bases de datos, nos estaremos refiriendo tanto al conjunto de información ordenada como al software que la controla, gestiona y permite al usuario interaccionar con ella.**

# Motivación

**X** Edix Educación

---

Ahora que tenemos claro a qué nos estamos refiriendo al hablar de base de datos, deberíamos identificar **los principales beneficios** que nos pueden aportar para poder identificar si nos serán útiles en nuestro día a día.

Entre las **principales ventajas** que nos ofrecen las bases de datos, podemos destacar las siguientes.

1

Rapidez en el acceso a la información

Una vez que conocemos el lenguaje, podremos **acceder de forma rápida y directa** a toda la información almacenada.

2

Centralización y consistencia de la información

Siguiendo una **buenas políticas de diseño e implementación de base de datos**, lograremos que la información esté presente en un único lugar, por lo que evitaremos posibles problemas de actualización de información y conseguiremos que todos los usuarios trabajen con la misma versión de la información.

3

### Seguridad y mantenimiento

Mediante el uso de las bases de datos, estamos agregando una **capa extra de seguridad a nuestros datos**, ya que será el propio SGBD el que se encargue de controlar el acceso y los permisos de los usuarios.

4

### Variedad y especialización

Una de las más importantes, y que suele quedar en el olvido, es la **gran variedad de tipos distintos de bases de datos** que existen en el mercado, lo que hace que se adapten con mucha facilidad a cualquier necesidad que se nos presente.

# Tipos de bases de datos

X Edix Educación

---

Existen **diferentes tipos de clasificaciones** que nos permiten distinguir entre los distintos tipos de bases de datos, pero principalmente podremos dividirlas de la siguiente manera.

## Bases de datos relacionales

Su funcionamiento se basa en introducir datos en registros organizados en tablas. Gracias a esto, se pueden establecer las relaciones existentes entre datos de forma sencilla y cruzar rápidamente para **emitir los reportes y análisis necesarios**. En la mayoría de estas bases de datos se usa el **lenguaje SQL** (*structured query language*).

Existen numerosas **ventajas** que nos pueden hacer elegir el uso de una base de datos relacional, pero vamos a destacar algunas de ellas.

- 1 Poseen un **lenguaje de definición de datos** o DDL (*data description language*), que nos permite definir la base de datos.
- 2 También un lenguaje de manipulación de datos o DML (*data manipulation language*), que nos permite **insertar, modificar y borrar datos** en la base de datos.
- 3 Nos ofrece un **acceso controlado a la base de datos** mediante un lenguaje de control de datos o DCL (*data control language*). Con este lenguaje, podremos dar permisos sobre los elementos creados en la base de datos (GRANT) con la posibilidad de que estos sean posteriormente eliminados (REVOKE).

4

Nos garantiza la **disponibilidad de los datos**, aunque se haya producido un fallo, ya sea de hardware (máquina) como del software (debe ser capaz de reponerse a fallos propios).

5

También tendremos un **acceso concurrente a la información**, es decir, varios usuarios podremos acceder a la información, modificarla y/o borrarla simultáneamente, siempre garantizando la consistencia e integridad de los datos.

Para poder garantizar esa consistencia, la base de datos necesita que todas sus transacciones sigan el **principio ACID**, es decir, que cumplan una serie de propiedades: **atomicidad, consistencia, aislamiento y durabilidad**.



#### Atomicity (atomicidad)

Garantizar que, dentro de la secuencia de instrucciones, esta se haya ejecutado en su totalidad. Si se produjera un error en una de ellas, se debería invalidar todos los cambios producidos por las instrucciones anteriores.

### Consistency (consistencia) —

Asegurar que, tras la ejecución de la secuencia de instrucciones, el estado final en el que queda la base de datos es válido y consistente, cumpliendo con las restricciones que tuviera definidas la base de datos.

### Isolation (aislamiento) —

Una información puede ser modificada a la vez por un solo usuario. Esto evita que se produzcan mezclas entre los cambios realizados por distintas transacciones.

### Durability (durabilidad) —

O la garantía de que el resultado de la transacción completada persista en la base de datos, aunque se produzcan errores posteriores.

A la hora de usar las bases de datos relacionales, podremos elegir entre una multitud de **alternativas**, entre las que destacan:

- Oracle.
- IBM.
- Microsoft SQL Server.
- MaríaDB
- PostgreSQL

## Oracle



Oracle ofrece **distintos tipos de licencias** que varían según las características contratadas y que obviamente afectan de manera directa al coste de estas. Su papel ha sido fundamental en la proliferación de las bases de datos y de las relacionales, en particular. Durante un gran periodo de tiempo su cuota de mercado fuese casi total.

---

**En los últimos años, esta cuota se ha visto reducida por el empuje de la competencia.**

## IBM



IBM también ha tenido un papel principal en el desarrollo de las bases de datos con la **creación de su primera versión hace ya casi 30 años**. La solución empresarial que ofrece IBM es **DB2**, un motor de base de datos SQL multiplataforma.

## Microsoft SQL Server



En el caso de Microsoft, su software de gestión es Microsoft SQL Server que, aunque originalmente solo ha estado disponible para **sistemas operativos Windows**, desde 2016 se encuentra también disponible para Linux y Docker.

## MySQL



Se trata del SGBD open source más popular del mundo. Tuvo un auge muy importante con la llegada de la **World Wide Web (WWW)** y, por consiguiente, con el desarrollo de aplicaciones web (considerado como uno de los cuatro componentes del stack de desarrollo LAMP/WAMP).

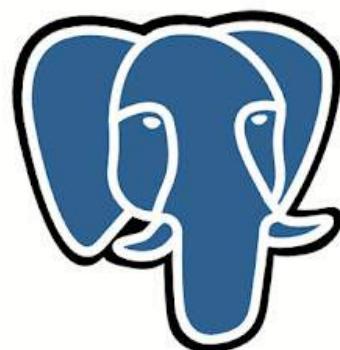
Al contrario que otros proyectos, como Apache (donde el software es desarrollado por una comunidad pública), **MySQL está patrocinado por una empresa privada** (Oracle) que posee el copyright de la mayor parte del código.

## MariaDB



Con la compra de MySQL por parte de Sun Microsystems, el fundador de MySQL y la comunidad de desarrolladores de software libre deciden crear un SGBD, derivado de MySQL, llamado MaríaDB. Es altamente compatible con MySQL pues surge con el objetivo de que se pueda migrar de un entorno a otro sin problemas.

## Postgre SQL



Postgre**SQL**

Se trata del SGBD de código abierto más potente del mundo. Detrás de él, existe una comunidad de desarrolladores muy importante. Surgió como un proyecto en la Universidad de Berkeley en 1982 en un intento de **implementación de un motor de base de datos relacional**.

# Bases de datos no relacionales

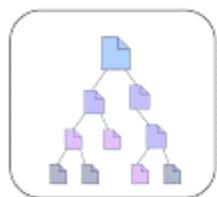
X Edix Educación

---

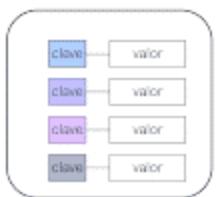
Están basadas en el almacenamiento en distintas estructuras que buscan optimizar los procesos de almacenamiento, mejorando la escalabilidad y el rendimiento de las bases de datos relacionales. A diferencia de las **bases de datos relacionales**, los datos no están almacenados en tablas, si no que el formato varía en función de la necesidad del usuario.

Podremos **clasificar las bases de datos no relacionales** (o también llamadas No-SQL), en 4 grandes tipos en función de la orientación:

- Documentales.
- Grafos.
- Clave/valor.
- Columnares.



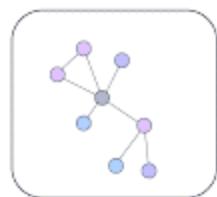
Documentales



Clave/valor



Columnares



Grafos

# Bases de datos documentales

X Edix Educación

---

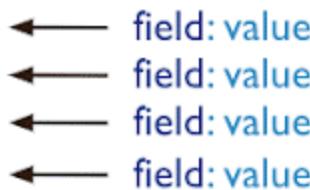
Como su nombre indica, son bases de datos orientadas al documento. El documento no es más que un contenedor de datos semiestructurados (como contrapunto del mundo relacional, donde se dice que la información está estructurada). La información es codificada dependiendo de la implementación, normalmente se realiza mediante XML, JSON, etc. Son excelentes **bases de datos para la escritura y consulta**, pues tienen una gran capacidad de indexación.

---

Las bases de datos más conocidas de este paradigma son  
MongoDB y CouchDB.

---

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```



The diagram shows a JSON object with four fields: 'name', 'age', 'status', and 'groups'. Four black arrows point from the text 'field: value' to each of these four fields respectively.

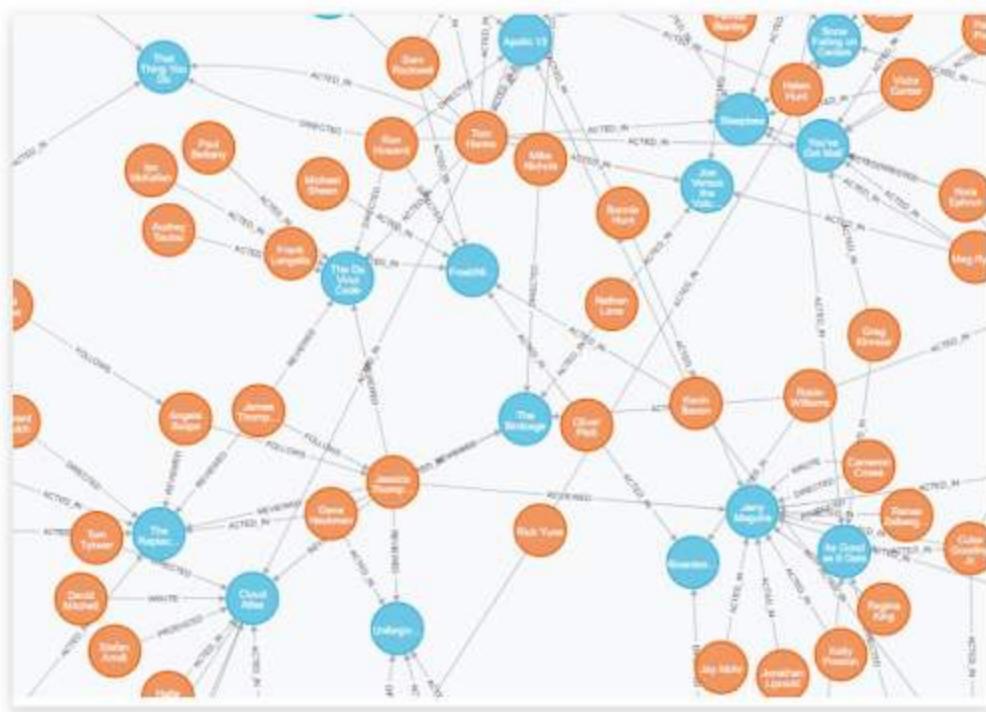
Ejemplo de documento, con distintos campos y valores.

---

# Bases de datos de grafos



Son bases de datos orientadas a almacenar la información en estructuras en forma de grafos. Son útiles cuando la información almacenada tiene una gran interrelación entre ella. Poseen un **gran rendimiento a la hora de analizar y consultar volúmenes de datos gigantescos**. Como base de datos más conocida de este paradigma, tenemos Neo4j e HyperGraphDB.



Ejemplo de visualización de un grafo en Neo4j. Los nodos azules representan películas y los nodos naranjas a los actores/directores, las flechas representan las interacciones entre los actores/directores y las películas.

# Bases de datos clave/valor

X Edix Educación

---

Se tratan de **bases de datos no relacionales** que utilizan un modelo de almacenamiento muy simple. Las claves actúan como identificadores únicos y dentro del valor se almacena cualquier tipo de datos que se quiera tener, ya sean objetos simples o complejos.

- Son bases de datos **altamente escalables en horizontal**, presentando un gran rendimiento a la hora de leer y escribir datos.
  - Es una **tipología de base de datos NoSQL** muy popular por su simplicidad en cuanto a funcionalidad.
- 

Como bases de datos de este paradigma tenemos: **Cassandra, BigTable de Google, Dynamo de Amazon, etc.**

# Bases de datos columnares

X Edix Educación

---

Una base de datos columnar almacena la información en columnas (como veremos, en el mundo relacional, la información se almacena en filas). Están pensadas para grandes volúmenes de datos. Entre las principales bases de datos nos encontramos a Hbase de Apache, Cassandra (se trata de un híbrido, pues también está orientada a clave-valor).

---

**Pero entre tal cantidad de oferta, ¿cómo podremos elegir cuál es la base de datos que más nos conviene?**

Existe un teorema (**CAP**) que nos indica que en sistemas distribuidos es imposible garantizar a la vez consistencia, disponibilidad y tolerancia a particiones.

Entendamos primeramente estas **tres características**:

## Consistencia (Consistency)

---

Al realizar una consulta o una inserción en una base de datos, independientemente del nodo o servidor en el que estemos, deberíamos recibir la misma información.

### Disponibilidad (Availability)

Todos los clientes pueden leer y escribir en cualquier momento, ya que el sistema siempre está disponible.

### Tolerancia a particiones (Partition tolerance)

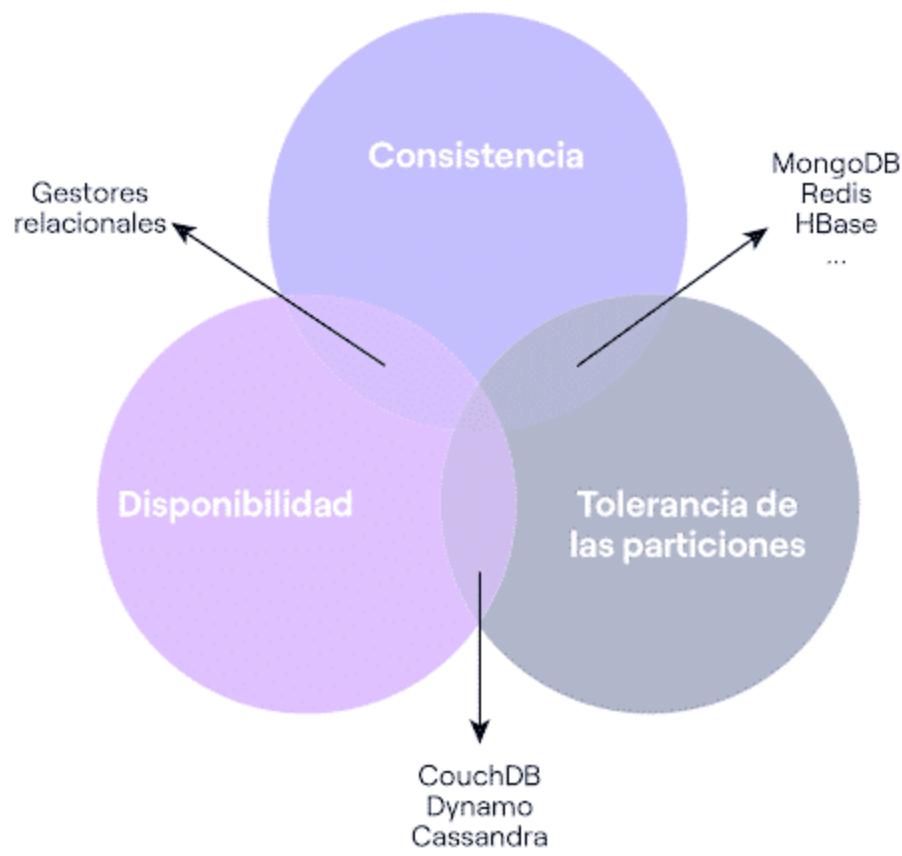
El sistema debe seguir funcionando, aunque existan fallos en los nodos de la red o no se puedan comunicar entre sí.

No todas las bases de datos NoSQL cumplen **los mismos puntos del teorema del CAP**. Así pues, existen varios tipos:

- **Disponibilidad más tolerancia a particiones.** Aunque no pueden garantizar la consistencia en todos los nodos de la red, sí podrían llegar de manera parcial a un subconjunto de la red.
- **Consistencia más tolerancia a particiones.** Para poder conseguir estas dos características, se sacrifica la disponibilidad.
- **Garantizan consistencia y disponibilidad**, pero no cumplen con la tolerancia a particiones.

---

Como curiosidad, existen bases de datos NoSQL que pueden cambiar su configuración para garantizar una de las características en detrimento de otra.



# Bases de datos cloud

**X** Edix Educación

---

Las plataformas cloud (Google, Amazon y Microsoft) nos **ofrecen su oferta tecnológica a través de servicios.**

---

Es la plataforma la que se encarga de gestionar o administrar el servicio contratado y nosotros solo tenemos que preocuparnos en usarlo.

---

Las **principales ventajas** de este paradigma de computación son claras:

## La infraestructura física desaparece

---

Es el proveedor el responsable del mantenimiento y disponibilidad de los sistemas.

## Reducción de costes

---

Solo pagamos por lo que usamos. Reducimos el coste de licencias software y el mantenimiento de las máquinas.

**Escalabilidad** —

Se adapta mejor a las necesidades que tengamos en cada momento.

Pero también tiene alguna que otra **desventaja**:

**Dependencia** —

Al proveedor al que le contratemos el servicio.

**Limitado** —

Normalmente, a ser servicios gestionados por el proveedor, tendremos un control limitado sobre las funcionalidades del sistema.

**Reticencia** —

Aunque cada vez menos, todavía existe cierto miedo entre las empresas a dar el salto a entornos cloud, su principal preocupación es la seguridad de la información, lo que muchas veces es su principal activo.

Entre las distintas opciones que nos ofrecen las distintas **plataformas cloud**, destacan:



- **Amazon RDS.** Un servicio que permite de una manera sencilla configurar, usar y escalar según la necesidad de una base de datos relacional. Dentro de su oferta, ofrece los siguientes motores (SGBD) de base de datos: PostgreSQL, MySQL, MariaDB, Oracle Database, SQL Server y Amazon Aurora.
- **Amazon DynamoDB.** Se trata de una base de datos NoSQL, orientada a clave–valor y documentos. Ofrece las ventajas ya observadas en las soluciones anteriores: alta escalabilidad, libre de administración...
- **Amazon DocumentDB.** Un servicio de base de datos NoSQL orientado a documentos y compatible con MongoDB.
- **Amazon Keyspaces.** Un servicio de base de datos compatible con Apache Cassandra, completamente administrado, de alta disponibilidad y escalable.



- **Cloud SQL.** Un servicio de bases de datos relacional. Ofrece los motores de bases de datos MySQL, PostgreSQL y SQL Server. Como principales características podemos destacar su alta disponibilidad (SLA > 99,95%), que es fácilmente integrable con el resto de los servicios de Google Cloud y está completamente administrada.
- **Cloud Spanner.** Se trata de una base datos relacional nativa de entorno cloud. Escalable a nivel mundial, disponibilidad > 99,999%, garantiza transacciones ACID (coherencia inmediata).

- **Cloud Datastore.** Base de datos NoSQL orientada a documentos, transacciones atómicas (no garantiza ACID), capaz de escalar a nivel de Terabytes.
- **Cloud Bigtable.** Base de datos NoSQL nativa de la nube orientada al paradigma clave valor. Es altamente escalable con baja latencia y alto rendimiento. Como curiosidad, es usada por servicios internos de Google como Google Maps y Gmail.
- **BigQuery.** Se trata del datawarehouse analítico de Google que, a pesar de no ser un tipo de base de datos como tal, soporta búsquedas realizadas con SQL sobre una gran cantidad de información de manera rápida y sin tener que preocuparnos del mantenimiento del servidor ni de su escalabilidad.



- **Azure SQL Database:** Una base de datos relacional, serverless, con almacenamiento escalable. Es un servicio administrado que incluye inteligencia artificial permitiendo optimizar el rendimiento y la durabilidad.
- **Azure Database for MySQL/MariaDB/PostgreSQL:** Servicio de base de datos relacional, totalmente administrado para los motores de bases de datos MySQL, PostgreSQL y MariaDB. Proporciona alta disponibilidad y escalado.
- **Azure Cosmos DB:** Base de datos NoSQL, totalmente administrada, alta disponibilidad (99,999%) y baja latencia de respuesta. Dispone de una escalabilidad inmediata y automática. Permite conectarse a Cassandra y MongoDB a través de sus respectivas API.
- **Azure Cache for Redis:** Almacén de datos en memoria totalmente administrado.

Podemos observar que, a pesar de la gran oferta que se nos presenta, **los servicios ofertados son bastante equivalentes**, por lo que la elección de su uso dependerá, sobre todo, de nuestros gustos personales o la experiencia que tengamos en cada plataforma.

# Conclusiones

X Edix Educación

---

Como pequeño resumen de lo comentado en este fastbook, me gustaría destacar qué hemos aprendido en esta unidad.

- Hemos visto a qué hace referencia el término '**base de datos**' y los conceptos clave que nos hacen falta para poder identificarlas.
- También hemos descubierto los **beneficios** que nos aportan y las principales motivaciones para usarlas.
- Ya conocemos las **principales tipologías** de las bases de datos presentes en el mercado y los principales representantes de cada una de ellas.
- Sabemos que las bases de datos relacionales usan el **lenguaje SQL** y que todas sus transacciones cumplen el **principio ACID**. Dentro de todo el catálogo de bases de datos, destacaremos PostgreSQL, por ser la más potente actualmente, principal motivo de ser la elegida en esta formación.
- Para solucionar algunos **problemas de rendimiento**, surgen las bases de datos no relacionales, entre las que podremos distinguir cuatro grandes tipos en función del tipo de estructura usada para el almacenamiento de información.
- Las **bases de datos documentales** son excelentes a la hora de la escritura y consulta, ya que tienen una gran capacidad de indexación.

- Las **bases de datos de grafos** se usan para almacenar información interrelacionada entre ella. Por ejemplo, sería la más indicada si queremos almacenar el listado de películas rodadas en 2022 y los actores en las que en ellas participan. Este tipo de bases de datos ofrece una navegación más eficiente entre relaciones que en un modelo relacional.
- Las **bases de datos clave-valor** son las bases de datos más populares dentro de las no relacionales. Además, son excelentes tanto en la lectura como en la escritura.
- Las **bases de datos columnares** son especialmente útiles cuando estamos trabajando con grandes volúmenes de información o tenemos que realizar operaciones de cálculo sobre elementos de la misma dimensión (valores máximos, mínimos, medianas...).
- También hemos visto que las **principales plataformas cloud** nos ofrecen numerosas alternativas para que podamos usar bases de datos, pero las soluciones que ofertan son bastante parecidas entre una plataforma y otra, por lo que independientemente de la plataforma en la que estemos trabajando, nuestro trabajo será prácticamente el mismo.

¡Enhorabuena! Fastbook superado

edix

Creamos Digital Workers