



# Unidad 0

## Introducción a las bases de datos

### Bases de datos

Customer
Customer_id
Firstname
Lastname
Address
Postal_code
Age
Gender
Email
Order_id
Invoice_id

Order
Order_id
Total
Product_id
Customer_id
Date_time
Remark

Product
Product_id
Product_name
Amount
Price
Description
Image
Date_time
Status
Statistic

# Índice



## Fundamentos de bases de datos

- ¿Qué es una base de datos?
- ¿Por qué es útil utilizar una base de datos?
- Modelos de base de datos

## Sistemas gestores de bases de datos (SGBD)

### Diseño y creación de una base de datos

#### Las primeras bases de datos

Los archivos o ficheros

#### Mapas conceptuales



## Fundamentos de bases de datos

En la actualidad, las bases de datos son un pilar esencial dentro del ámbito informático y tienen presencia en prácticamente todas las áreas del conocimiento. Desde la gestión empresarial hasta la investigación científica, pasando por el comercio electrónico y la administración pública, el almacenamiento y tratamiento de datos se ha convertido en una necesidad ineludible. La utilidad de las bases de datos es aún mayor cuando los volúmenes de información crecen, ya que proporcionan mecanismos eficientes para organizar, recuperar y analizar datos de manera estructurada.

### ¿Qué es una base de datos?

Una base de datos es un conjunto organizado de datos que se almacena de manera estructurada con el propósito de facilitar su consulta, modificación y administración. Su objetivo principal es ofrecer un sistema eficiente y seguro para gestionar la información, asegurando su integridad y disponibilidad.

Los datos contenidos en una base de datos pueden ser de cualquier naturaleza: desde registros numéricos y textos, hasta imágenes, sonidos o datos espaciales. Lo que distingue a una base de datos de un simple almacenamiento de información es su estructura organizada y su gestión sistemática. Estos dos aspectos son clave, ya que permiten un acceso más rápido a los datos, evitan redundancias innecesarias y garantizan la coherencia de la información almacenada.





### Ejemplo:

Imaginemos el caso de un urbanista que debe planificar la instalación de una red de transporte público en una ciudad en expansión. Para diseñar el trazado más eficiente de las rutas de autobuses o tranvías, necesitará utilizar datos como la distribución de la población, la ubicación de centros de trabajo y la densidad del tráfico en diferentes zonas. Si trabajara en un equipo multidisciplinar con expertos en movilidad, gestores municipales y analistas de datos, la mejor estrategia sería compartir la información de manera coordinada para tomar decisiones fundamentadas.

Del mismo modo que estos profesionales combinan sus conocimientos para diseñar un sistema de transporte eficiente, los datos también deben gestionarse de manera compartida y no como recursos aislados de cada usuario.

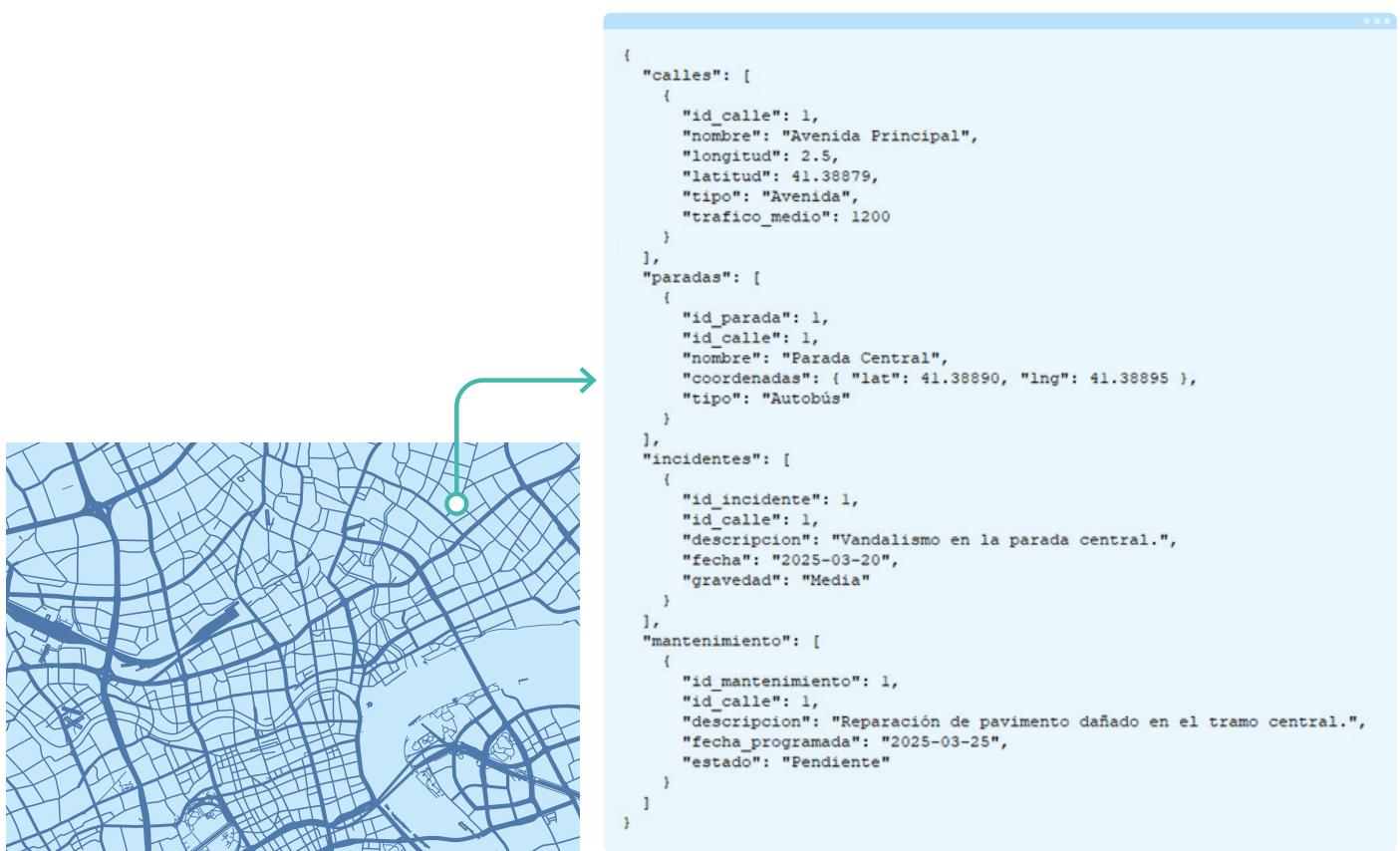


Imagen 1. Representación gráfica del problema de datos de la red de transporte

A simple vista, coordinar el uso de los datos en un equipo pequeño parece sencillo. Si el analista de movilidad ya cuenta con información sobre densidad de tráfico, podría compartir una copia con el urbanista, quien trabajaría con ella por su cuenta. Sin embargo, en este enfoque cada profesional maneja su propia versión de los datos, generando copias independientes que pueden evolucionar de manera diferente y, con el tiempo, dejar de estar sincronizadas.



El problema se agrava cuando aumenta el número de personas que necesitan acceder y modificar los mismos datos. Si cada usuario mantiene sus propios archivos y los comparte solo cuando alguien los solicita, se crean múltiples versiones con información potencialmente desactualizada o inconsistente. Esto suele ser un obstáculo frecuente en grandes organizaciones, donde diferentes departamentos pueden utilizar información similar sin siquiera estar en contacto.

Por ejemplo, pensemos en una base de datos con información sobre las paradas de autobús de la ciudad. El urbanista la utilizará para planificar nuevas rutas, mientras que el departamento de mantenimiento se centrará en registrar averías en las marquesinas y el equipo de seguridad querrá conocer las zonas con mayor incidencia de vandalismo.

Si cada grupo gestiona su propio archivo con la información que necesita, pueden surgir inconsistencias: el urbanista podría estar usando un mapa con paradas desactualizadas, mientras que el equipo de mantenimiento podría estar tomando decisiones basadas en ubicaciones incorrectas.

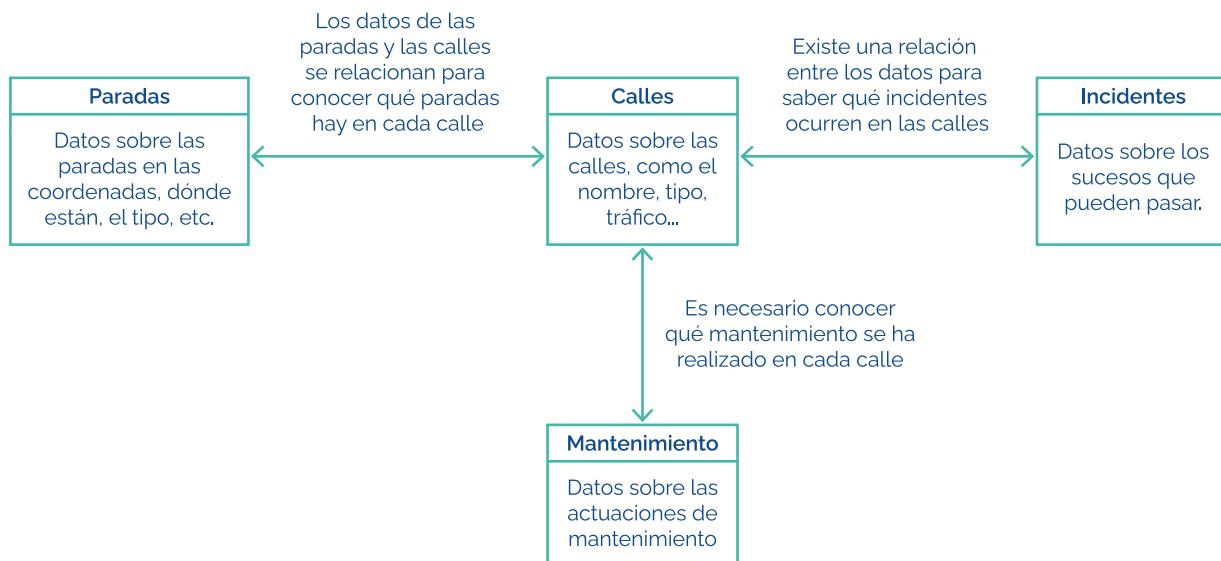


Imagen 2. Representación de la estructura de datos de la red de transporte

Supongamos ahora que el urbanista detecta que una parada fue eliminada y actualiza su archivo. Si los demás departamentos no reciben esa actualización, seguirán operando con información incorrecta.

Ocurre lo mismo si el equipo de seguridad añade datos sobre incidentes en cada parada: esa información podría ser útil para el urbanista al planificar nuevas rutas, pero si no existe una estructura centralizada, los datos quedarán fragmentados y serán difíciles de consolidar en el futuro.



En definitiva, cuando los datos trascienden el uso individual y se convierten en herramientas esenciales para múltiples usuarios, su gestión debe ser eficiente y organizada. Administrarlos manualmente a través de archivos independientes no es una solución viable a largo plazo. La forma óptima de gestionar estos datos es mediante bases de datos, que permiten un acceso centralizado, actualizado y estructurado.

Además, los sistemas gestores de bases de datos (SGBD) facilitan la interacción entre los datos y los distintos usuarios, garantizando que la información sea consistente y accesible para todos los que la necesiten.

## ¿Por qué es útil utilizar una base de datos?

El uso de bases de datos ofrece múltiples ventajas en comparación con un almacenamiento desorganizado de la información. Permiten una gestión más eficiente y segura de los datos, evitando problemas comunes como la duplicación innecesaria, la inconsistencia o la falta de acceso centralizado.

A continuación, se detallan algunas de sus principales ventajas:

### Beneficios en la gestión de los datos:

- > La información almacenada en una base de datos no está ligada a una única aplicación o usuario, lo que permite que pueda ser utilizada por diferentes sistemas sin depender de un formato específico.
- > Los datos pueden ser accesibles desde distintos dispositivos, aplicaciones y ubicaciones, lo que facilita su uso en múltiples contextos y por diferentes usuarios.
- > Una base de datos permite establecer mecanismos de control de acceso, copias de seguridad automatizadas y replicación de datos, lo que garantiza su integridad y minimiza el riesgo de pérdida de información. Al centralizar la gestión de los datos, cualquier modificación se sincroniza automáticamente, asegurando que todos los usuarios trabajen con información actualizada.
- > A diferencia de un almacenamiento basado en múltiples archivos dispersos, una base de datos evita la duplicación de datos, optimizando el espacio de almacenamiento y mejorando la velocidad de acceso.
- > Gracias a su estructura organizada, la inserción y actualización de la información es más rápida y precisa, evitando errores comunes en la gestión manual de ficheros.





### Beneficios en la explotación de la información:

- > Al garantizar que los datos están bien estructurados y sin duplicaciones, la información extraída es más precisa y fiable, lo que se traduce en mejores resultados en su análisis.
- > La base de datos **permite consultas optimizadas**, facilitando la recuperación de información de manera rápida y eficiente, sin necesidad de recorrer grandes volúmenes de archivos manualmente.
- > Más allá de almacenar datos, una base de datos está **diseñada para facilitar su análisis y explotación**, permitiendo descubrir patrones, realizar cálculos o generar **informes de manera sencilla**.

### Beneficios para los usuarios:

- > El usuario solo necesita conocer las herramientas y mecanismos de consulta adecuados, sin preocuparse por la organización interna de los datos.
- > La información almacenada en una base de datos puede ser reutilizada en distintos proyectos o compartida con otros usuarios sin perder su integridad ni generar duplicaciones innecesarias.

## Modelos de base de datos

Dependiendo de cómo se estructuran y almacenan los datos, existen diversos modelos de bases de datos. Cada uno de estos modelos define un paradigma de almacenamiento, es decir, cómo se organizan los datos y las relaciones entre ellos. Las operaciones realizadas sobre la base de datos, como la eliminación, la modificación o la lectura de datos, dependen de la estructura empleada, y existen diferencias notables entre los principales modelos. Cada uno tiene sus propias ventajas e inconvenientes, que se deben considerar al elegir el más adecuado para un determinado propósito.





## Sistemas gestores de bases de datos (SGBD)

Los Sistemas Gestores de Bases de Datos (SGBD) son elementos fundamentales para aprovechar eficazmente una base de datos. Actúan como un intermediario entre los datos y las aplicaciones que los utilizan, facilitando las operaciones sobre esos datos.

Se puede resumir que un SGBD es una pieza de software compleja debido a que debe responder a diversas necesidades, muchas de ellas con requisitos elevados, como la eficiencia y el manejo de grandes volúmenes de datos. Por ejemplo, una base de datos moderna puede contener millones de registros y ser utilizada por miles de usuarios simultáneamente. Estos usuarios pueden acceder a la base de datos a través de diferentes aplicaciones, que no siempre son del mismo tipo.

Consideremos una base de datos que almacene información como números de teléfono, nombres de usuarios y coordenadas geográficas. Esta base de datos podría ser utilizada para crear un mapa de densidad de usuarios, desde una aplicación para generar un listín telefónico, o incluso desde una página web que permita buscar el número de teléfono de una persona. Aunque estas aplicaciones tienen objetivos distintos, todas comparten el acceso a la misma base de datos. El SGBD debe proporcionar la metodología adecuada para que cada aplicación obtenga solo la información que necesita de la base de datos.

Además de facilitar el acceso a los datos, el SGBD es una herramienta necesaria para la gestión y mantenimiento de la base de datos. La administración de bases de datos puede ser una tarea compleja, por lo que el SGBD debe ofrecer herramientas para hacer más sencilla esta gestión.

Para que un SGBD sea verdaderamente útil, debe cumplir con varios objetivos esenciales:

- > **Acceso transparente a los datos.** Los usuarios no deberían tener que preocuparse por detalles técnicos de la estructura interna de la base de datos.
- > **Protección de los datos.** Si la base de datos contiene información sensible, el SGBD debe garantizar que solo los usuarios autorizados puedan acceder a ella.
- > **Eficiencia.** El SGBD debe poder gestionar grandes volúmenes de datos y responder de manera eficiente a múltiples peticiones simultáneas. Esto implica que, incluso si muchos usuarios están accediendo a la base de datos a la vez, el sistema debe seguir funcionando con fluidez y sin demoras significativas.
- > **Gestión de transacciones.** Las transacciones son operaciones que se realizan sobre la base de datos como una unidad indivisible. Pueden incluir acciones como añadir, eliminar o modificar registros. El SGBD debe gestionar estas transacciones de forma que mantenga la integridad de los datos, incluso si se producen fallos durante el proceso (por ejemplo, por un error del sistema o un corte de energía).

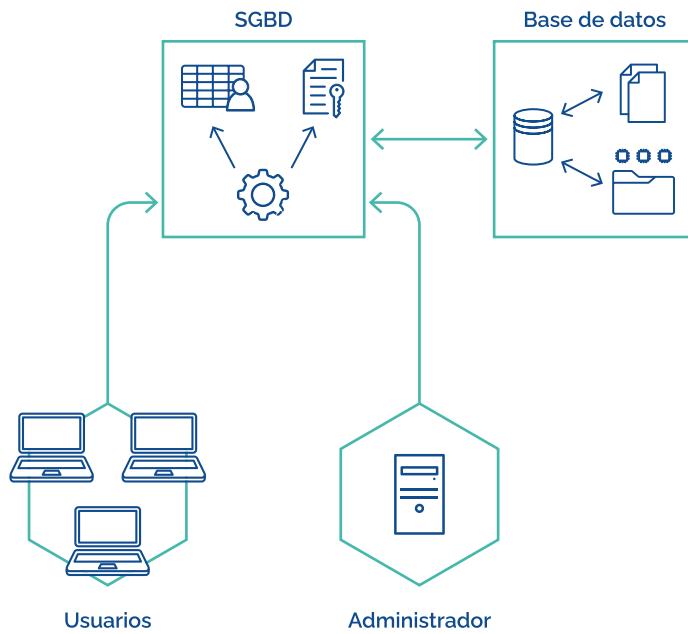
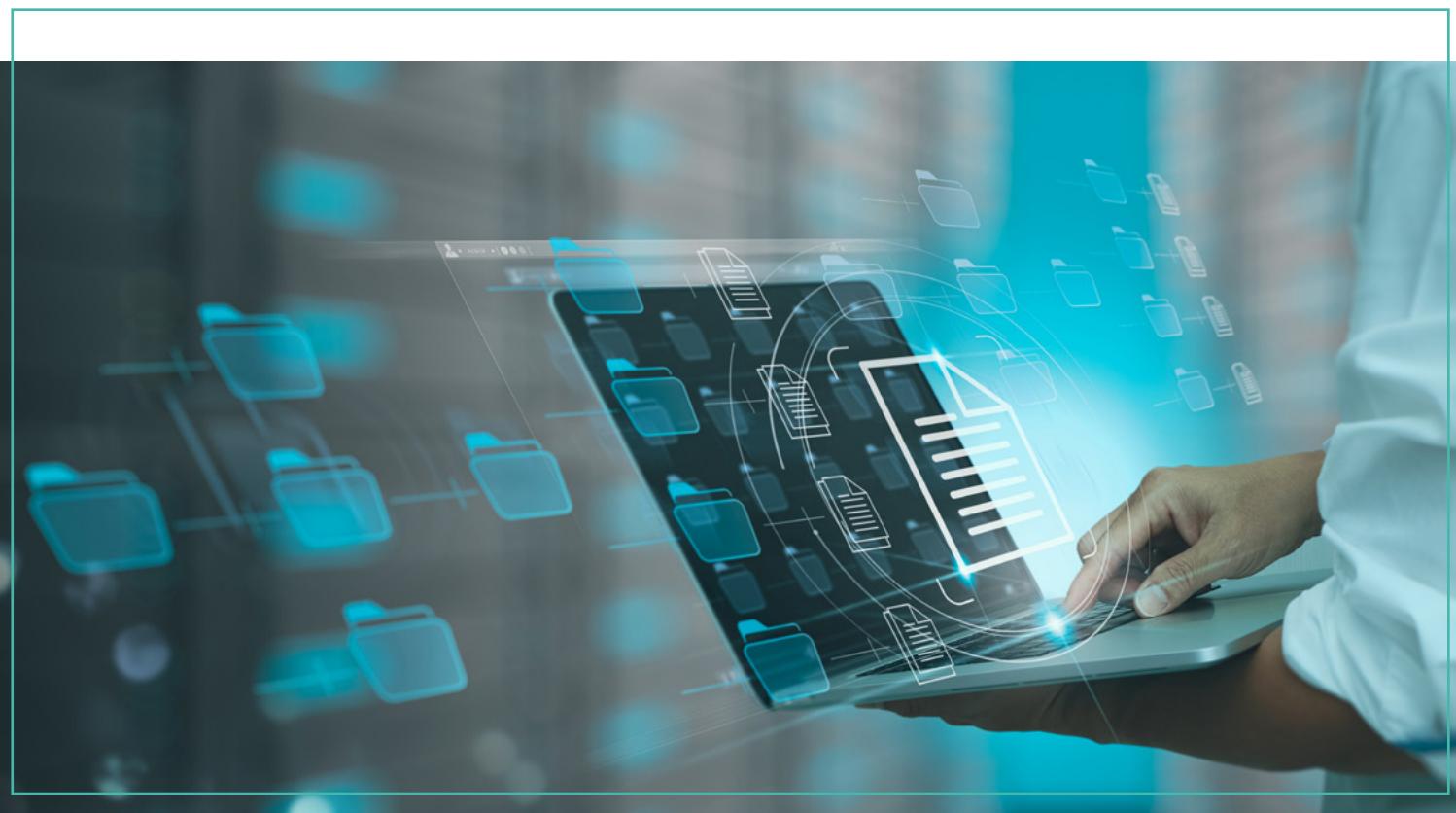


Imagen 3. El papel del SGBD en el manejo de los datos

El SGBD se adapta a los diferentes modelos de base de datos que se empleen, ya que debe ser capaz de ofrecer las funcionalidades necesarias según el modelo subyacente. Por lo tanto, su diseño y características varían según el tipo de base de datos que gestionen, proporcionando diferentes métodos de acceso y manipulación de los datos.





## Diseño y creación de una base de datos

El diseño y la creación de una base de datos es un proceso muy importante para asegurar que la estructura, el almacenamiento y la gestión de los datos se realicen de manera eficiente y efectiva. Al igual que en la construcción de una casa, no se debe comenzar directamente por levantar las paredes. Antes, es necesario realizar estudios, análisis y un diseño detallado, con planos, para garantizar que la construcción sea sólida. Imagina que no se realiza ningún estudio previo, se construye la casa de manera improvisada y, con el tiempo, se desploma porque el terreno no era adecuado.

Con las bases de datos sucede lo mismo: si no se lleva a cabo una planificación adecuada, el sistema puede volverse ineficaz o insostenible en el futuro.

Para la creación de una base de datos, se siguen varias fases que deben realizarse con cuidado. Estas fases son:

- > **Fase 1. Diseño lógico.** En esta fase aún no se conoce el Sistema de Gestión de Bases de Datos que se va a utilizar. Aquí se modela el contenido de la base de datos sin tener en cuenta las especificaciones del SGBD, lo que implica una representación abstracta de la información a almacenar. El objetivo es crear una representación conceptual clara de los datos y sus relaciones. En esta fase interviene el diagrama E/R, que viene a ser el plano con los datos modelados.
- > **Fase 2. Diseño físico.** El diseño físico adapta el modelo conceptual de la fase 1 a las especificaciones del SGBD elegido, asegurando que la base de datos pueda implementarse de manera eficiente en el sistema. Este diseño aborda cómo los datos se almacenarán físicamente, incluyendo las estructuras de tablas, índices y claves. En esta fase se aplican diferentes procesos para garantizar que los datos cumplan las especificaciones concretas, como por ejemplo, el proceso de normalización.
- > **Fase 3. Implementación.** En esta fase, se lleva a cabo la creación de la base de datos propiamente dicha, siguiendo los esquemas definidos en las fases anteriores. Es cuando se incorporan los datos al sistema, creando las estructuras de datos y relaciones necesarias para asegurar la integridad y el acceso adecuado.
- > **Fase 4. Mantenimiento.** Una vez implementada la base de datos, se debe realizar un mantenimiento continuo para garantizar que siga funcionando correctamente. Esto incluye monitorear la actividad, actualizar los datos y adaptar la estructura de la base de datos a nuevos requisitos si es necesario.



## Las primeras bases de datos

En los primeros programas, el manejo y almacenamiento de datos era completamente diferente a lo que conocemos hoy. En lugar de almacenar los datos de manera estructurada y organizada, no se les consideraba como un componente dentro de un sistema, sino como algo directamente ligado a la aplicación específica. Esto significaba que los datos solían almacenarse en archivos con formatos cerrados, pensados para ser leídos y escritos exclusivamente por esa aplicación. Como resultado, el uso compartido y la posibilidad de acceder a esos datos desde otros sistemas era muy limitado.

Integrar datos de otros contextos o aplicaciones era complicado, ya que existía una conexión muy estrecha entre el software y los datos que manejaba. Las funciones del programa estaban diseñadas para trabajar solo con esos datos específicos, y todas las operaciones de acceso o modificación de los datos se gestionaban directamente dentro de la aplicación, sin el soporte de un Sistema de Gestión de Bases de Datos (SGBD).

Como no existían SGBDs en este contexto, la responsabilidad de gestionar la integridad de los datos o el acceso concurrente de varios usuarios recaía sobre la propia aplicación, lo que limitaba su capacidad para manejar operaciones más complejas.

Una de las ventajas de este enfoque, aunque limitada, era el rendimiento. Debido a que la aplicación estaba completamente optimizada para un propósito específico, en algunos casos el manejo de los datos podía ser más rápido que si se utilizara un SGBD. Sin embargo, este beneficio venía a costa de la flexibilidad, escalabilidad y la posibilidad de reutilizar los datos en otros sistemas. En resumen, el enfoque era eficiente en algunos aspectos, pero carecía de la capacidad de adaptarse o compartir datos más allá de la aplicación original.

## Los archivos o ficheros

Un archivo o fichero de datos es básicamente una colección de registros que comparten características comunes y están organizados con un propósito específico. Por ejemplo, un archivo que contenga los datos de una clase escolar incluiría los registros de todos los estudiantes de esa clase, cada uno con su propia información.

En el contexto de un equipo informático, un archivo es una estructura destinada a almacenar datos de manera organizada, permitiendo que estos datos puedan ser fácilmente recuperados, modificados, eliminados y luego almacenados nuevamente con los cambios realizados.

Si definimos una base de datos en términos más generales, podemos decir que es un conjunto de archivos relacionados entre sí, a los cuales se puede acceder mediante un SGBD. Estos archivos contienen datos que están lógicamente conectados y estructurados.



Como ya se ha mencionado, los archivos se componen de registros, que son unidades de información generalmente asociadas a una entidad específica. Un registro está formado por varios campos, que son los elementos de datos básicos. Cada campo tiene propiedades como tamaño, longitud y tipo, y todos los campos dentro de un registro están relacionados entre sí. Por ejemplo, el registro de un empleado podría incluir campos como su nombre, dirección, fecha de nacimiento, entre otros, los cuales juntos proporcionan una visión completa de la entidad (en este caso, el empleado) en cuestión. Estos registros pueden ser procesados como una unidad por un programa, lo que facilita su manejo.

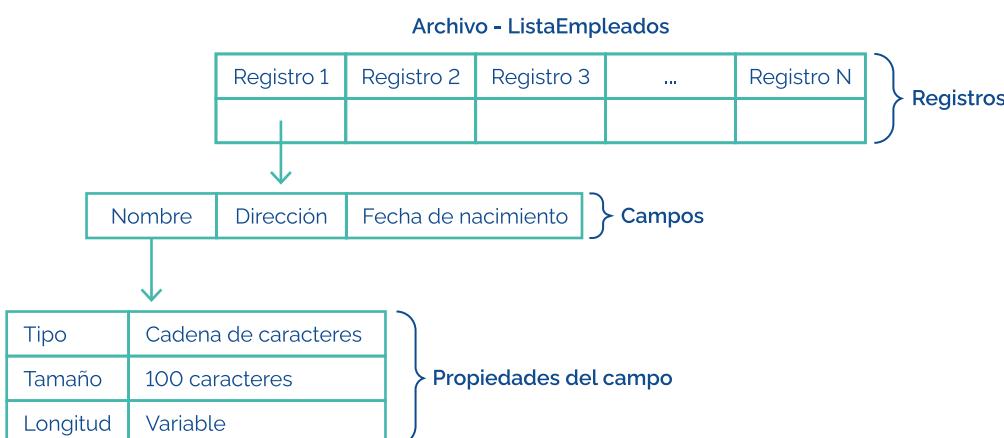


Imagen 4. Representación de registros y campos de una estructura de datos de empleados

Las operaciones que se pueden realizar sobre los registros de un archivo son las siguientes:

- > **Creación.** Implica establecer una estructura organizada para que el archivo sea fácilmente accesible y gestionable. Para que un archivo esté listo para usarse, debe existir físicamente, lo que significa que sus datos deben haberse almacenado en un soporte de almacenamiento, como un disco duro, y ser accesibles. La creación de un archivo requiere planificación, ya que implica decidir cómo se organizarán los datos y cómo se optimizará el espacio en el medio de almacenamiento. Un archivo puede crearse desde cero, a partir de otro archivo ya existente, o incluso como resultado de un cálculo.
- > **Consulta.** Esta operación permite al usuario acceder a los datos del archivo. A través de consultas, se puede recuperar información específica de uno o varios registros, según las necesidades del usuario. Es fundamental para cualquier tipo de análisis de datos, ya que permite extraer la información relevante de manera rápida.
- > **Actualización.** Esta operación asegura que el archivo siempre contenga datos actuales y correctos. La actualización abarca varias acciones: consulta de registros, inserción de nuevos registros, eliminación de registros obsoletos y modificación de los existentes.



- > **Clasificación.** Es una operación importante para organizar los registros del archivo de manera eficiente. Implica ordenar los registros según los valores de un campo específico, ya sea de forma ascendente o descendente. La clasificación puede realizarse de manera alfabetica (por ejemplo, por nombre) o numérica (por ejemplo, por fecha o ID), lo que facilita la búsqueda y el acceso a la información.
- > **Reorganización.** Con el tiempo, después de múltiples operaciones (como inserciones, eliminaciones o modificaciones), un archivo puede volverse menos eficiente en cuanto a su acceso y rendimiento. La reorganización consiste en crear una nueva copia del archivo con una estructura optimizada, eliminando los fragmentos innecesarios o reorganizando los datos para mejorar el rendimiento. Esto asegura que el acceso a los registros siga siendo rápido y eficiente.
- > **Destrucción.** Es la operación contraria a la creación de un archivo. Cuando un archivo se destruye, se elimina completamente, lo que significa que sus datos ya no estarán disponibles ni accesibles.
- > **Unión de archivos.** Es una acción de fusión que permite combinar varios archivos en uno solo. Los registros de diferentes archivos se intercalan entre sí siguiendo ciertos criterios de orden, lo que da como resultado un archivo único que contiene todos los registros de los archivos originales.
- > **Partición de archivos.** A diferencia de la unión de archivos, este proceso consiste en dividir un archivo grande en varios archivos más pequeños. Esto puede ser útil si el archivo original es demasiado grande o si se desea organizar los datos en diferentes categorías o grupos para facilitar su manejo.

Hasta este punto, hemos hablado de cómo los archivos están formados por registros lógicos a nivel estructural. Estos registros lógicos son las unidades de información que representan entidades específicas, como un empleado, un producto, o un cliente, y contienen varios campos de datos relacionados entre sí. Sin embargo, cuando hablamos de cómo se almacenan estos archivos en la memoria secundaria (como un disco duro o una unidad SSD), estamos tratando con registros físicos o bloques.

### Registros físicos o bloques

Un registro físico o bloque es la unidad mínima de datos que puede ser transferida en una operación de entrada/salida (E/S) entre la memoria central (RAM) y los dispositivos periféricos de almacenamiento (como discos duros, SSDs, etc.). En otras palabras, cuando se leen o escriben datos en un archivo desde o hacia el dispositivo de almacenamiento, se hacen en bloques de tamaño fijo.

Cada bloque tiene un tamaño determinado por el sistema de almacenamiento, y este tamaño no necesariamente coincide con el tamaño de un registro lógico. Por ejemplo, si un sistema tiene bloques de 4 KB, y cada registro lógico ocupa 1 KB, entonces un bloque físico puede contener varios registros lógicos. Sin embargo, si el registro lógico es más grande que el bloque físico, el registro completo tendrá que ser almacenado en varios bloques.



### Factor de bloqueo

El factor de bloqueo es un término utilizado para describir cuántos registros lógicos caben dentro de un bloque físico. Es decir, el factor de bloqueo nos indica la relación entre la unidad lógica de datos (registro) y la unidad física de almacenamiento (bloque).

- > Si el factor de bloqueo es menor que 1, implica que un bloque físico no puede contener ni un solo registro completo y en este caso se tendría que utilizar varios bloques físicos para almacenar un único registro lógico. Esto genera un mayor número de operaciones de entrada/salida, ya que cada operación de lectura o escritura tendría que acceder a múltiples bloques para recuperar o almacenar un solo registro lógico.
- > Si el factor de bloqueo es 1, significa que un bloque físico contiene exactamente un registro lógico. Esto sucede cuando los registros son de un tamaño similar o mayor que el tamaño del bloque físico.
- > Si el factor de bloqueo es mayor que 1, implica que un solo bloque físico contiene varios registros lógicos, lo que permite optimizar el uso del espacio de almacenamiento y reducir el número de operaciones de entrada/salida, ya que múltiples registros pueden ser procesados en una sola operación.

### Ejemplo

Imagina un escenario en el que tenemos registros lógicos que representan la información de empleados en una empresa. Cada registro contiene los siguientes campos: nombre, dirección, número de teléfono, y salario. Supongamos que cada uno de estos registros ocupa 1 KB de espacio.

Si el sistema de almacenamiento está configurado para manejar bloques de 4 KB, el factor de bloqueo  $4\text{ KB} / 1\text{ KB} = 4$ . Esto significa que cada bloque físico puede contener hasta 4 registros lógicos, y al realizar una operación de lectura o escritura, el sistema leería o escribiría estos 4 registros a la vez en un solo bloque.

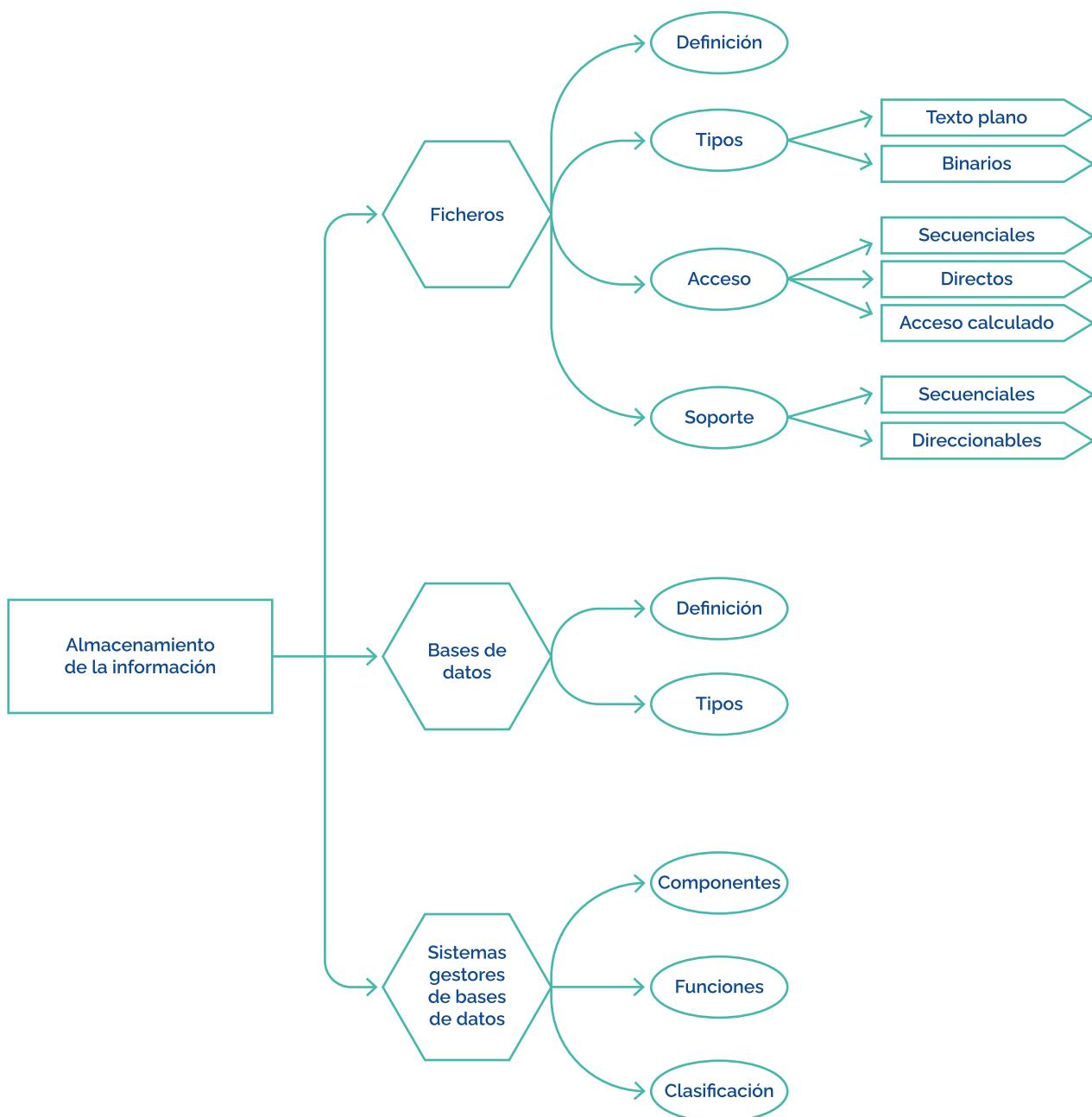
Por otro lado, si los registros son más grandes, digamos que cada uno ocupa 4 KB (el mismo tamaño que el bloque físico), el factor de bloqueo sería 1, ya que cada bloque físico solo puede contener un registro lógico.



## Mapas conceptuales

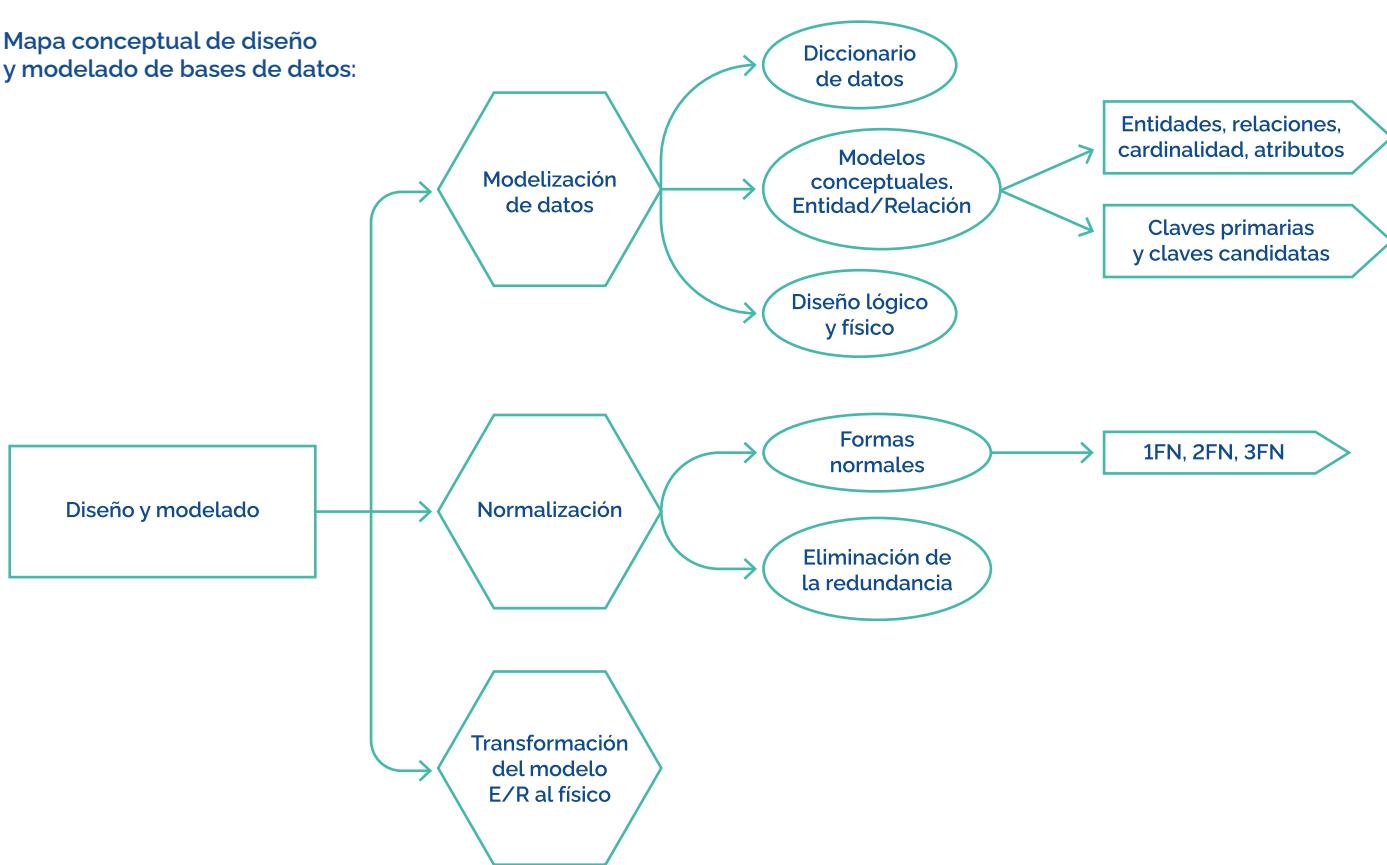
En esta sección, se presentan diversos mapas conceptuales resumidos que facilitan la visualización de los temas clave a tratar en este módulo.

Mapa conceptual del almacenamiento de la información:

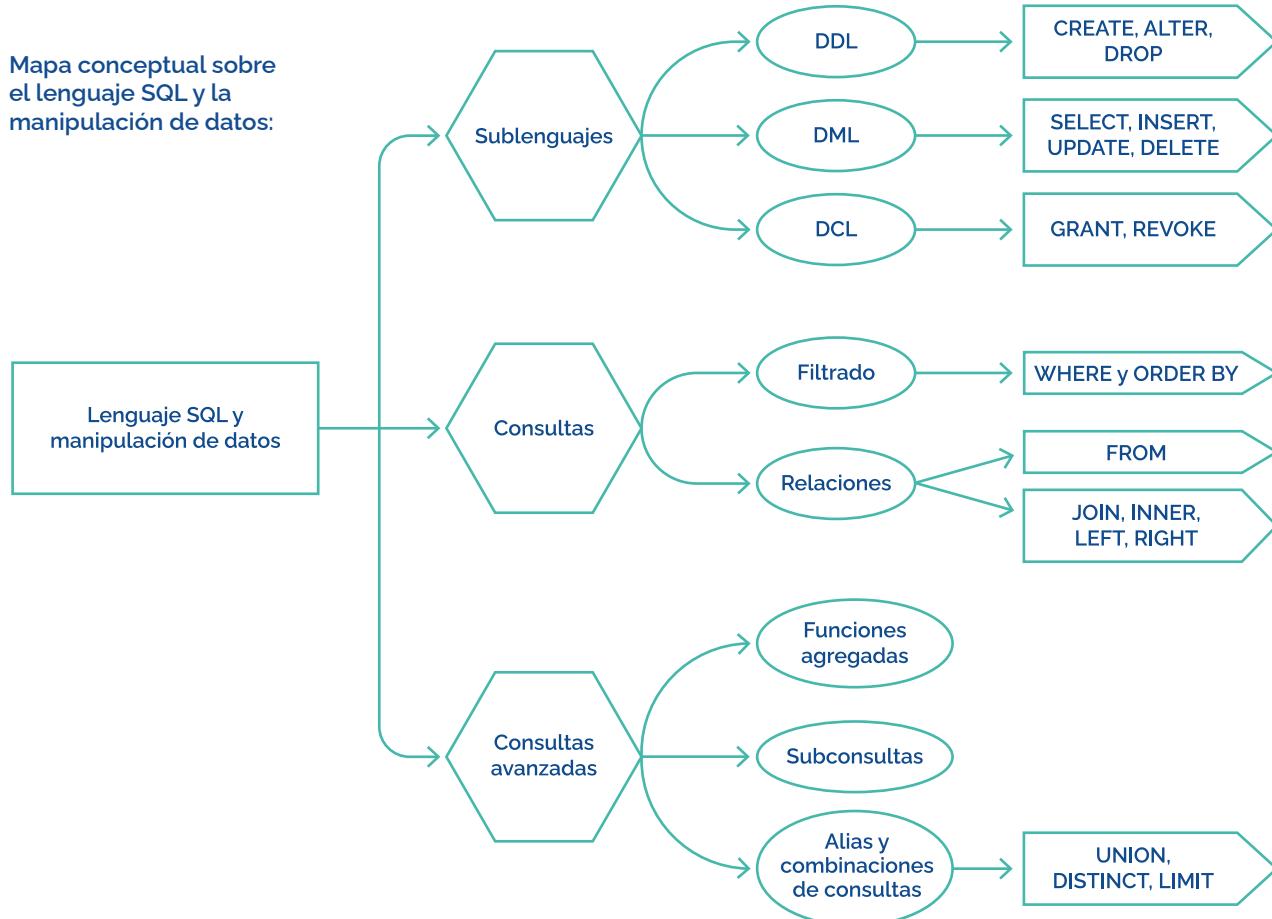




**Mapa conceptual de diseño y modelado de bases de datos:**

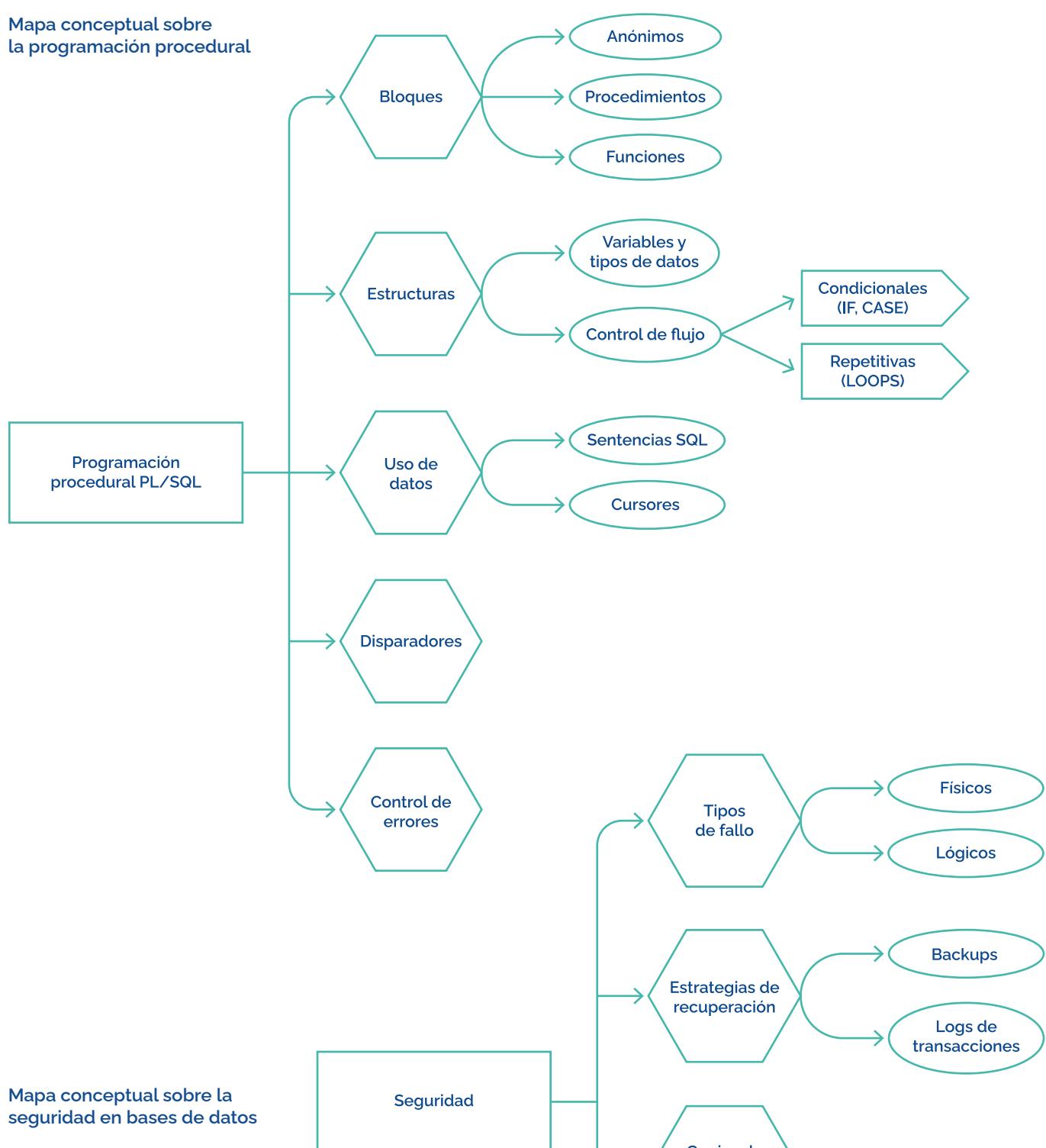


**Mapa conceptual sobre el lenguaje SQL y la manipulación de datos:**





Mapa conceptual sobre la programación procedural

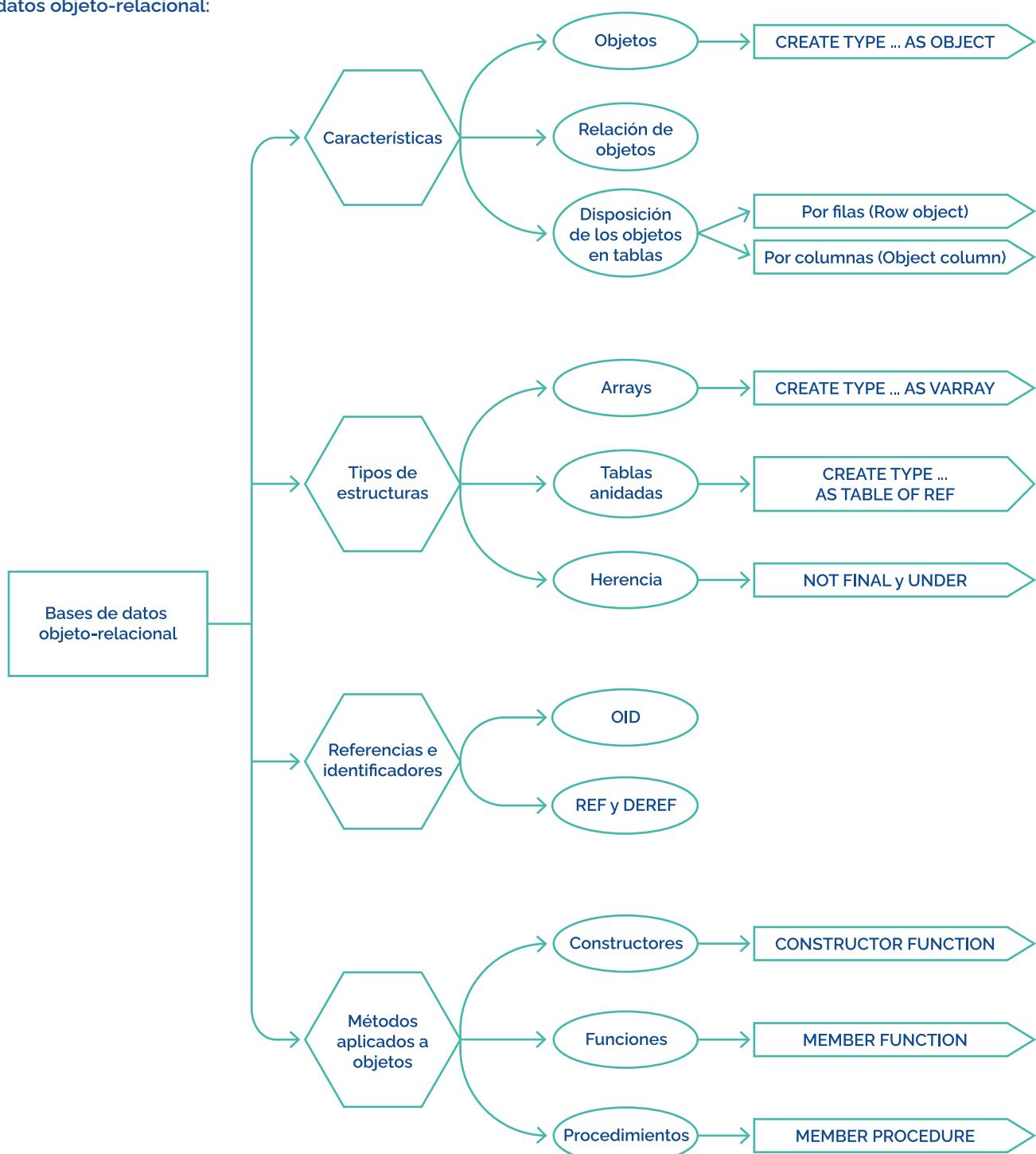


Mapa conceptual sobre la seguridad en bases de datos





**Mapa conceptual sobre las bases de datos objeto-relacional:**





 [www.universae.com](http://www.universae.com)

