

# Bases de Datos

## Carrera Programador full-stack

*Introducción*

# La Información como Recurso

- Los datos son observaciones que realizamos en el mundo real



- Son recopilados como **hechos ó evidencias**
- Adquieren significado a **partir de su procesamiento**

# La Información como Recurso

- Existen diferencias entre dato e información
  - Los datos son hechos o cosas del mundo, lo suficientemente importantes **como para ser registrados**. Son recopilados como **situaciones o evidencias**. Adquieren significado a partir de su procesamiento para convertirse en información
  - La información es el **resultado del análisis de los datos**. Por ejemplo, un satélite colecciona datos (valores numéricos), que una vez volcados en un mapa, son interpretados por un experto que identifica los diferentes usos del suelo en determinada región

# Necesidades de Almacenar Información

- Problema: Contamos con mucha información



- Y contamos con diferentes alternativas de almacenado:

- Un archivo de texto
- Varios archivos de texto
- Otras estrategias



# Necesidades de Almacenar Información

- En los comienzos de la computación como el objetivo era la resolución de problemas, se prestaba poca consideración al almacenamiento de datos
- Normalmente se archivaban en un formato específico (no-estándar) para cada aplicación. En otras palabras, cada programador trabajaba con un formato de datos propio

# Necesidades de Almacenar Información

- Los problemas comenzaron cuando fue necesario compartir información y los formatos que usaban las aplicaciones, no eran compatibles
- Por ejemplo, en una empresa había varias aplicaciones que guardaban datos en formatos diferentes:
  - **Compras** mantenía la información sobre los artículos comprados;
  - **Almacén**, sobre los que estaban en stock y
  - **Ventas** sobre los artículos vendidos.

# Necesidades de Almacenar Información

- Pero, por más que en los tres casos se habla de los mismos datos (los artículos que la empresa comercializaba), **no se podía compartir esa información**, ya que las aplicaciones que había desarrollado cada sector guardaban la información en un **formato distinto que servía en forma específica para cada caso**
- Para solucionar este inconveniente, las organizaciones se veían obligadas a **exportar los datos a los diferentes formatos** que requería cada aplicación, tarea que era de por sí bastante engorrosa y no siempre posible



# Necesidades de Almacenar Información



- Pero el principal problema era que se debía **mantener varias copias de la misma información** en los diferentes formatos que requerían las aplicaciones existentes
- Este procedimiento, inevitablemente creaba **redundancia de información**: imaginen en una empresa, la cantidad de veces que era necesario guardar el nombre de cada uno de los productos
- A su vez, **inconsistencia**: múltiples copias conllevan inevitablemente a errores. El **costo de mantenimiento** era altísimo: sí un cliente cambiaba de dirección había que cambiar el dato en tantos archivos como aplicaciones hubiera



# Necesidades de Almacenar Información



- Entonces surge la necesidad de crear un sistema al cual **acceden todas las aplicaciones** de la organización
- Se desea que, por ejemplo, al ingresar los datos de un cliente, este registro sirva a todos los sectores y que **si este cliente cambia de domicilio, todas las aplicaciones lo registren**
- En definitiva, se busca que todos los sistemas en una organización (ventas, contabilidad, compras, etc.) **compartan un solo almacén de datos**

# Bases de Datos

- Es así como surgen las bases de datos. Una base de datos es una entidad en la cual se pueden almacenar datos de **manera estructurada**, con la **menor redundancia posible**
- **Diferentes programas y diferentes usuarios** deben poder utilizar estos datos. Por lo tanto, el concepto de base de datos generalmente está relacionado con el de red



# ¿Por qué utilizar una Base de Datos?

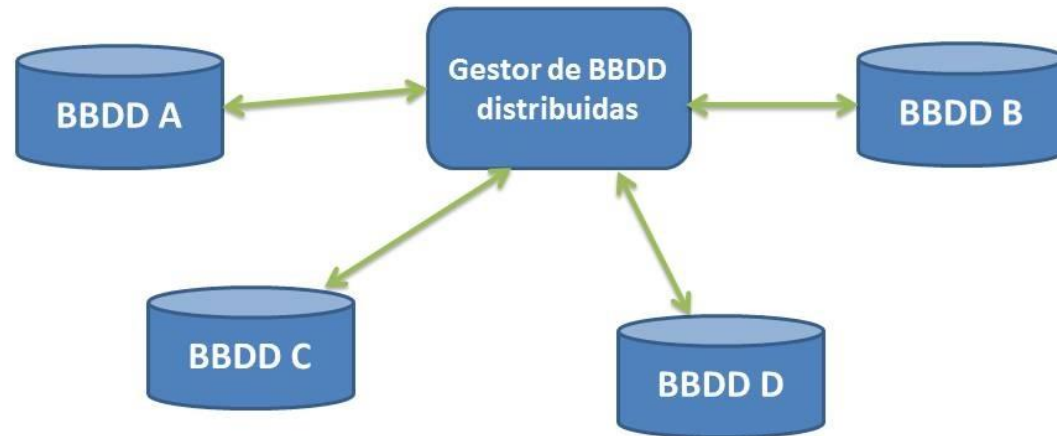
- Una base de datos proporciona a los usuarios el acceso a datos, que pueden visualizar, ingresar o actualizar, en concordancia con los privilegios de acceso que se les hayan otorgado



- Una base de datos puede ser local, es decir que puede utilizarla solo un usuario en un equipo, o puede ser distribuida, es decir que la información se almacena en equipos remotos y se puede acceder a ella a través de una red

# ¿Por qué utilizar una Base de Datos?

- Se hace más útil a medida que la cantidad de datos almacenados crece
- La principal ventaja de utilizar bases de datos es que múltiples usuarios pueden acceder a ellas al mismo tiempo. En la figura podemos ver distintas bases de datos (BBDD) distribuidas con un gestor que las administra



# Administración de Bases de Datos

- Rápidamente surgió la necesidad de contar con un sistema de administración para controlar tanto los datos como los usuarios
- La administración de bases de datos se realiza con un Sistema de Gestión de Bases de Datos (SGBD) también llamado **DBMS** (*DataBase Management System*)
- El DBMS es un conjunto de servicios (aplicaciones de *software*) que permite a los distintos usuarios un fácil acceso a la información y proporciona las herramientas para la manipulación de los datos encontrados en la base (insertar, eliminar, editar)

# Administración de Bases de Datos

- Un DBMS presenta las siguientes características:
  - **Independencia:** Los datos se organizan independientemente de las aplicaciones que los vayan a usar (independencia lógica) y de los ficheros en los que vayan a almacenarse (independencia física)
  - Los usuarios y las aplicaciones pueden acceder a los datos mediante el uso de **lenguajes de consulta** (p.ej. SQL)
  - **Centralización:** Los datos se gestionan de forma centralizada e independiente de las aplicaciones
  - **Consistencia e integridad** de los datos
  - **Fiabilidad** (protección frente a fallos) y **seguridad** (control de acceso a los datos)

# Tipos de Bases de Datos

- Existen diferentes tipos de bases de datos:

- Jerárquicas
- En red
- Orientadas a Objetos
- Documentales
- Relacionales



- Sin embargo, las bases de datos **relacionales** fueron adoptadas expresamente por casi todos los sistemas comerciales conocidos gracias a su **coherencia** y **facilidad de uso**

# Base de Datos Relacional

- El modelo relacional aporta las siguientes ventajas:
  - Independencia lógica y física de los datos, de las aplicaciones
  - Redundancia mínima
  - Acceso concurrente por parte de múltiples usuarios
  - Distribución espacial de los datos
  - Integridad de los datos
  - Consultas complejas por más de una condición
  - Seguridad de acceso y auditoría
  - Respaldo y recuperación
  - Acceso a través de lenguajes de programación estándar





# Base de Datos Relacional

- El modelo relacional tiene estas desventajas:
  - No soportan las relaciones complejas que existen en el mundo real
  - No tienen conocimiento semántico, es decir, no entienden qué significan los datos que guardan
  - Los tipos de datos son limitados (numéricos, alfanuméricos, etc.)
  - Tienen dificultades con el tiempo considerado como una sucesión natural. En general, solo se obtiene un registro de la información en determinado momento



# Conceptos de Bases de Datos

- Una base de datos se compone de entidades (cosas u objetos del mundo real distinguibles de todos los demás objetos) que poseen atributos (propiedades o características de las que se quiere llevar registro)
- Las **entidades** pueden ser cosas concretas, como personas o libros, o abstractas, como un préstamo o una venta

# Conceptos de Bases de Datos

- Supongamos la entidad ALUMNO sus atributos podrían ser
  - Documento de Identidad
  - Nombre
  - Apellido
  - Edad
  - Materias cursadas
  - Notas obtenidas



# Conceptos de Bases de Datos

- Supongamos la entidad VENTA, sus atributos podrían ser:
  - Tipo de Factura
  - N° de Factura
  - Nombre del Vendedor
  - Nombre del Comprador
  - Artículos Comprados



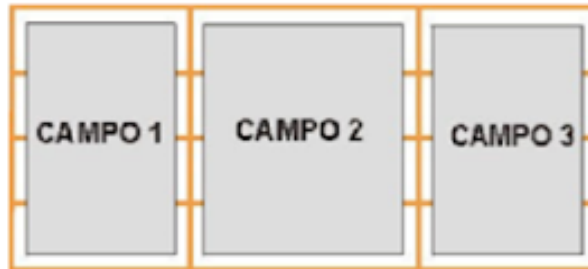
# Conceptos de Bases de Datos

- Una base de datos relacional siempre está **compuesta por tablas** que se conectan (relacionan) entre sí para compartir información
- Una tabla es un conjunto de datos sobre una entidad específica. Por ejemplo, en una base de datos de una empresa, se guarda en tablas diferentes información sobre los **clientes**, los **vendedores** y los **productos**



# Conceptos de Bases de Datos

- Las tablas organizan la información en columnas llamadas **campos** y filas llamadas **registros**



- Cada **registro** debe contener información de una sola unidad de análisis: por ejemplo, en la tabla CLIENTE, una fila por cada cliente



# Conceptos de Bases de Datos

- Los **campos** (columnas) **guardan los atributos** (propiedades o características) del registro
- Los campos **pueden ser de distintos tipos**: numéricos (edad, cantidad de hijos), alfanuméricos (nombre, dirección), fechas (fecha de nacimiento, fecha de ingreso a la compañía), booleano (posee auto propio, cumplió el servicio militar)
- Todos los datos registrados **en una columna deben ser del mismo tipo**
- El **nombre del campo debe ser único e inequívoco**: no se puede llamar a dos campos diferentes con el mismo nombre

# Conceptos de Bases de Datos

- En las bases de datos es necesario la creación de un campo denominado identificador que permita localizar el cliente
- En ningún momento dos filas de la misma tabla pueden tener el mismo identificador
- En este caso, **Nº de Cliente** es el atributo identificador que permite reconocer al comerciante en forma inequívoca

CLIENTE					
Nº de Cliente	Nº de Documento	Nombre	Apellido	Dirección	Teléfono
0023	10.551.985	Juana	Molina	Cabildo 110	4-784-6789
0056	11.346.897	Alberto	Alajuf	Lima 345	4-678-6782
0067	12.234.674	Maria	Jurez	Paz 897	3-457-8764
0098	12.458.432	Pedro	Colombo	Bolivia 877	6-567-8908



# Conceptos de Bases de Datos

- Las entradas de información deben ser atómicas. En otras palabras, debe ser información que no pueda ser divisible
- En una ficha papel es admisible registrar dos idiomas (inglés, francés) en un mismo casillero, pero en el modelo relacional esto no es admisible
- Una posible solución, aunque altamente desaconsejable, sería la creación de diferentes columnas: **idioma 1**, **idioma 2**, etc.
- **¿Por qué es desaconsejable?** Porque si alguna persona no habla dos idiomas, estaremos desperdiciando espacio (creamos una columna que no tendrá datos). Y si habla más de dos... **¿Dónde lo escribimos?**
- Este problema se denomina “de atributos repetidos o de grupos repetitivos”. Una solución a este problema es la creación de otra tabla con la información que se puede repetir

# Una Tabla Compleja vs. Varias Tablas Simples

- Si ahora agregamos en la tabla CLIENTE, al VENDEDOR que atiende a cada uno de ellos, con todos sus datos personales, surgiría una **enorme cantidad de problemas**
- Además de tener que ingresar todos los datos del vendedor por cada cliente (**redundancia**), si algún empleado cambia de teléfono deberíamos actualizar la información en muchos registros, tantos como tenga a cargo el vendedor, lo que llevaría indefectiblemente a generar **inconsistencias** (que en un registro figure el teléfono viejo y en otro el nuevo)
- También se agrandaría el tamaño de la base, lo que haría **más costoso el medio de almacenamiento** (discos rígidos) y **más largas las búsquedas** (porque se necesita pasar por registros más largos)

# Una Tabla Compleja vs. Varias Tablas Simples



- Todos estos problemas se solucionan si se crea una tabla llamada “VENEDORES”, donde cada registro se corresponda con un empleado de la empresa del sector ventas
- Nos debemos asegurar que cada registro tenga su **identificador único** (podría ser el legajo interno de la empresa)

VENEDORES					
Nº de Vendedor	Nº de Documento	Nombre	Apellido	Sector	Interno
1	14.985.551	Juan	Aragón	Agroquímicos	23
2	11.897.346	Manuela	Sarquis	Fertilizantes	24

# Una Tabla Compleja vs. Varias Tablas Simples

- Se debería relacionar esta nueva tabla con la tabla CLIENTES
- Para relacionar la tabla VENDEDORES con la tabla CLIENTES, en la tabla CLIENTES creamos un campo donde se ubica el **Nro. de Vendedor** que nos remitirá a los datos personales del mismo
- A partir de esta relación, obtenemos para cada cliente, los datos del vendedor que le ha sido asignado

VENDEDORES					
Nº de Vendedor	Nº de Documento	Nombre	Apellido	Sector	Interno
1	14.905.551	Juan	Aragón	Agroquímicos	23
2	12.234.674	Manuela	Sánchez	Fertilizantes	24

CLIENTES						
Nº de Cliente	Nº de Documento	Nombre	Apellido	Dirección	Teléfono	Nº de Vendedor
0023	10.551.985	Juana	Molina	Calle 110	4-704-6789	1
0056	11.346.897	Alberto	Alvar	Lima 345	4-670-6782	1
0067	12.234.674	Maria	Juarez	Paz 897	3-457-6764	1
0090	12.456.432	Pedro	Colombo	Bolivia 877	6-567-8906	2

# Bases de Datos

## *Modelo de Datos*

Para construir una aplicación que maneje una BD es crucial el diseño de la misma: lo más importante es la elección correcta del **Modelo de Datos**

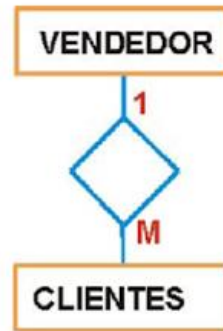
**Algunas de las consideraciones a tener en cuenta al momento de hacer el diseño de la BD son:**

- ✓La velocidad de acceso
- ✓El tamaño de la base de datos
- ✓El tipo de los datos
- ✓Facilidad de acceso a los datos
- ✓Facilidad para extraer los datos requeridos.

# Bases de Datos

## *Modelo de Datos*

Es un conjunto de **conceptos** utilizados para organizar los datos de interés y describir su estructura en forma comprensible para un sistema informático



Es una "descripción" de algo conocido como *contenedor* de datos (donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores

# Diseño de Bases de Datos

- Es el proceso por el que se determina la **organización** de una BD, incluidos *su estructura, contenido y las aplicaciones* que se han de desarrollar
- Desempeña un papel central en el empleo de los recursos de datos en la mayoría de las organizaciones
- Ha pasado a constituir parte de la **formación general de los informáticos**, en el mismo nivel que la capacidad de construir algoritmos usando un lenguaje de programación convencional

# Bases de Datos NoSQL

- Con la llegada de Internet, los servicios en la nube y millones de usuarios, también llegaron los problemas de alta escalabilidad
- Se pueden adaptar los modelos relacionales para hacerlos escalar en entornos difíciles pero se hacen cada vez menos intuitivos
- **NoSQL** es un término que describe BD no relacionales de alto desempeño





# Bases de Datos NoSQL

- Las **BD NoSQL** intentan atacar este problema proponiendo una estructura de almacenamiento más versátil a costa de perder ciertas funcionalidades de las BD relacionales:
  - manejo de transacciones
  - incapacidad de ejecutar consultas a más de dos tablas
  - duplicidad de datos
- Las **BD NoSQL no imponen** una estructura de datos y pueden almacenar los **datos en diferentes formatos**

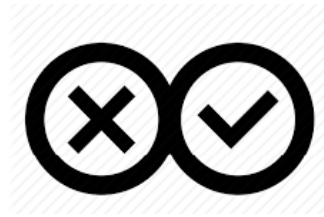
# Bases de Datos

## Carrera Programador full-stack

*Preguntas y Ejemplo*

# Introducción

## *Quiz*



**Marque sólo una opción correcta**

1- ¿Qué es una base de datos?

- A. Una base de datos es una colección de datos relacionados con un significado implícito
- B. Una base de datos es un DBMS
- C. Una base de datos es un software de gestión de datos
- D. Una base de datos es un complemento a un servidor de aplicaciones
- E. Una base de datos es una colección de datos relacionados sin un significado en particular

# Introducción

## Quiz



2- ¿Qué es un DBMS?

- A. Un DBMS es una base de datos que almacena datos de empresa
- B. Un DBMS es un software cuyo propósito es almacenar datos en la computadora local
- C. Un DBMS es un sistema de software de propósito general que facilita el proceso de definir, construir y manipular bases de datos para diferentes aplicaciones
- D. Un DBMS es una base de datos estándar del mercado

# Introducción

## *Quiz*

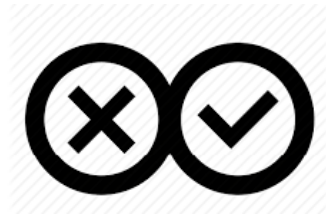


3- ¿Cuál de las siguientes frases es verdadera?

- A. Una BD es de gran tamaño y poco compleja
- B. Una BD puede ser de cualquier tamaño y complejidad
- C. Una BD es pequeña y muy compleja

# Introducción

## *Quiz*



4- ¿ Qué significa redundancia?

- A. Significa repetir la misma información más de una vez
- B. Significa que la misma información no se repite
- C. Significa no duplicar archivos

# Introducción

## *Análisis de Caso: Escuela*



- En la escuela hay muchos **alumnos**
- Cada alumno tiene solo un **maestro**, pero un maestro tiene varios alumnos a cargo
- Cada alumno tiene un solo **responsable**, pero una persona puede ser responsable de más de un alumno a la vez. Un ejemplo típico, un padre con más de un hijo en la escuela
- Cada responsable a cargo puede tener varios teléfonos. Un teléfono personal, varios laborales, un móvil
- La escuela ofrece varias **actividades complementarias** optativas, como ser cerámica, francés e inglés. Un alumno puede hacer varias **actividades complementarias**. Obviamente, en cada actividad complementaria estarán inscriptos muchos alumnos
- Por razones de privacidad, en una tabla aparte se desea tener la información acerca de la **salud del alumno**, como ser su grupo y factor sanguíneo, si tuvo apendicitis, si es alérgico a la penicilina, etc.

# Introducción

## *Análisis de Caso: Escuela*



1. ¿Qué tablas se requieren para administrar los registros de una escuela?
2. ¿Cuántas filas tendrá la tabla ALUMNO?
3. ¿Qué campos serán necesarios para guardar la información de los alumnos?
4. ¿Cuál será el identificador de la tabla MAESTRO?
5. ¿En la tabla ALUMNO, qué diferencia habría entre alumnos que en el campo “alérgico a la penicilina” tuvieran las siguientes repuestas: no, si y nulo?
6. ¿En la tabla ALUMNO y considerando que muchos alumnos, tienen más de un hermano, se podrían guardar en un mismo campo los nombres de todos los hermanos?