

# UD1.- SISTEMAS DE ALMACENAMIENTO DE LA INFORMACIÓN

Gestión de Base de Datos

# Índice

- **1. Sistemas de almacenamiento de la información**
- **Introducción.**
  - **1.1. Ficheros**
    - 1.1.1. Tipos de ficheros y formatos
    - 1.1.2. Ficheros de texto plano
    - 1.1.3. Ficheros binario
  - **1.2. Bases de Datos**
    - 1.2.1. Elementos de una Base de datos.
    - 1.2.2. Usos de las bases de datos
  - **1.3. Los Sistemas Gestores de Base de Datos**
    - 1.3.1. Concepto de Sistema Gestor de Base de Datos
    - 1.3.2. Funciones de un SGBD
    - 1.3.3. El lenguaje SQL
    - 1.3.4. Tipos de SGBD

# Introducción

Un ejemplo de que el hombre siempre ha tenido la necesidad de almacenar información.

*Tablillas de arcilla con documentación económica usadas en la antigua Grecia.*



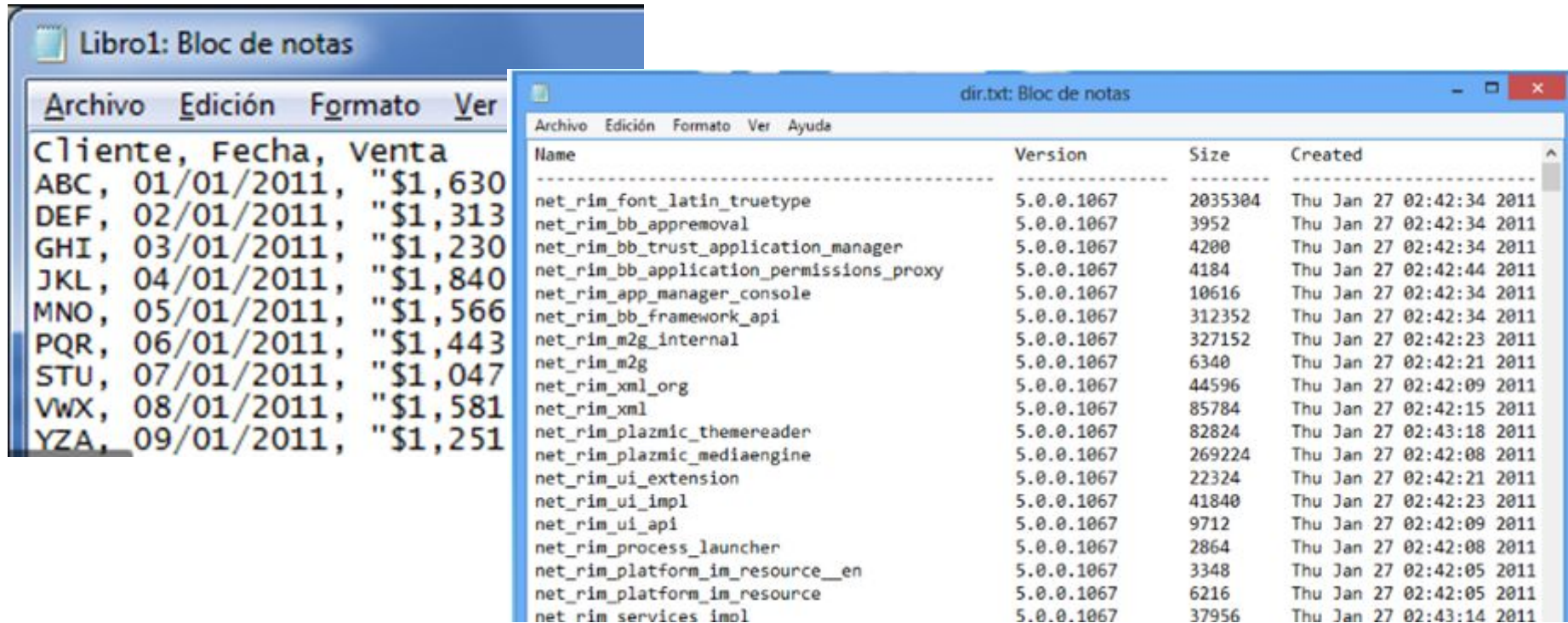
# Introducción

- Antes de la aparición de las aplicaciones informáticas, las empresas tenían como únicas herramientas de gestión de datos los ficheros con cajones, carpetas y fichas de cartón. **Ahora se utilizan SGBD. (Sistemas Gestores de Bases de Datos)**



# Introducción

- Antes de que surgieran las bases de datos el procesamiento automatizado de información se hacía mediante ficheros de texto plano organizados en columnas.



The image displays two screenshots of a Windows Notepad application, illustrating early data storage methods. The left window, titled 'Libro1: Bloc de notas', shows a text file with a table of sales data. The right window, titled 'dir.txt: Bloc de notas', shows a directory listing with columns for file name, version, size, and creation date.

**Libro1: Bloc de notas**

Cliente	Fecha	Venta
ABC	01/01/2011	"\$1,630
DEF	02/01/2011	"\$1,313
GHI	03/01/2011	"\$1,230
JKL	04/01/2011	"\$1,840
MNO	05/01/2011	"\$1,566
PQR	06/01/2011	"\$1,443
STU	07/01/2011	"\$1,047
VWX	08/01/2011	"\$1,581
YZA	09/01/2011	"\$1,251

**dir.txt: Bloc de notas**

Nombre	Version	Size	Created
net_rim_font_latin_truetype	5.0.0.1067	2035304	Thu Jan 27 02:42:34 2011
net_rim_bb_appremoval	5.0.0.1067	3952	Thu Jan 27 02:42:34 2011
net_rim_bb_trust_application_manager	5.0.0.1067	4200	Thu Jan 27 02:42:34 2011
net_rim_bb_application_permissions_proxy	5.0.0.1067	4184	Thu Jan 27 02:42:44 2011
net_rim_app_manager_console	5.0.0.1067	10616	Thu Jan 27 02:42:34 2011
net_rim_bb_framework_api	5.0.0.1067	312352	Thu Jan 27 02:42:34 2011
net_rim_m2g_internal	5.0.0.1067	327152	Thu Jan 27 02:42:23 2011
net_rim_m2g	5.0.0.1067	6340	Thu Jan 27 02:42:21 2011
net_rim_xml_org	5.0.0.1067	44596	Thu Jan 27 02:42:09 2011
net_rim_xml	5.0.0.1067	85784	Thu Jan 27 02:42:15 2011
net_rim_plazmic_themereader	5.0.0.1067	82824	Thu Jan 27 02:43:18 2011
net_rim_plazmic_mediaengine	5.0.0.1067	269224	Thu Jan 27 02:42:08 2011
net_rim_ui_extension	5.0.0.1067	22324	Thu Jan 27 02:42:21 2011
net_rim_ui_impl	5.0.0.1067	41840	Thu Jan 27 02:42:23 2011
net_rim_ui_api	5.0.0.1067	9712	Thu Jan 27 02:42:09 2011
net_rim_process_launcher	5.0.0.1067	2864	Thu Jan 27 02:42:08 2011
net_rim_platform_im_resource_en	5.0.0.1067	3348	Thu Jan 27 02:42:05 2011
net_rim_platform_im_resource	5.0.0.1067	6216	Thu Jan 27 02:42:05 2011
net_rim_services_imol	5.0.0.1067	37956	Thu Jan 27 02:43:14 2011

# Introducción



Los ficheros se diseñaban a medida para cada aplicación, **sin que existiera un formato común.**

## □ Ejemplo:

El software utilizado por el **departamento de recursos humanos** debía gestionar un fichero con **datos de empleados**, mientras la aplicación de **contabilidad mantenía** otro fichero distinto con los mismos datos organizados de otra forma.

# Introducción

## Problemas o inconvenientes de los ficheros de texto.

- ❑ **Redundancia** de datos
- ❑ Mal aprovechamiento del espacio de almacenamiento.
- ❑ Inconsistencia de información debida a la redundancia (si un dato cambiaba en el fichero de una aplicación, no cambiaba en los demás)
- ❑ Aislamiento de la información (imposibilidad de transferirla a otros programas a no ser que se desarrollara un software de migración específico.

**Todo esto se soluciona con los SGBD(Sistemas Gestores de Base de Datos)**

# 1.1. Ficheros

## ¿Qué tipo de información se almacena en nuestro PC?

- Música, películas, páginas web, datos administrativos, bancarios o contables, etc.
- Toda esta información está almacenada en los dispositivos de almacenamiento del ordenador (discos duros, pendrives, etc.).
- Se organizan utilizando los **ficheros** o **archivos**.
- Los **ficheros** son estructuras de información que crean los SO de los ordenadores para poder almacenar datos.
- Suelen tener un **nombre** y una **extensión**, que **determina el formato de la información que contiene**.



# 1.1.1. Tipos de ficheros y formatos

- El **formato** y **tipo de fichero** determina la forma de interpretar la información que contiene.
- Lo que se almacena en un **fichero** es una **ristra de bits** (ceros y unos). Es necesaria su interpretación para dar sentido a la información que almacena.
  - El formato es necesario, porque según este, el SO extraerá los bits y los mostrará por pantalla con el programa apropiado

# 1.1.1. Tipos de ficheros y formatos

- Descarga una imagen de Internet, y ábrela con el bloc de notas, pulsando con el ratón derecho sobre él y seleccionando la opción 'Abrir con'.

¿Qué ocurre? ¿Por qué?



# 1.1.1. Tipos de ficheros y formatos

## □ Ficheros de texto plano

Almacenan secuencias de caracteres correspondientes a una codificación determinada (ASCII, Unicode, etc). Son legibles mediante un software de edición de texto como el Bloc de Notas de Windows o el Vi de Linux o los editores de programación.

□ ¿Son legibles con los editores de programación?

# 1.1.1. Tipos de ficheros y formatos

## □ Ejemplos de ficheros de texto plano:

- Los ficheros con extensión **.txt**
- Los ficheros con extensión **.csv** donde los datos están separados por comas.
- **Ficheros de configuración:** Son ficheros cuyo contenido es texto sobre configuraciones del sistema operativo o de alguna aplicación. Estos pueden tener extensión **.ini**, **.inf** , **.conf**
- **Ficheros de código fuente:** Su contenido es texto con las instrucciones de los programas informáticos. Ejemplos: **.sql**, **.c**, **.java**
- **Ficheros de páginas web:** Las páginas webs son ficheros de texto con hipertexto que interpreta el navegador, **.html**, **.php**, **.css**, **.xml**

# Ficheros de texto plano

- Llamados también **ficheros ascii**.
- **Por ejemplo los .txt**
- Conéctate a Internet, y busca una tabla de códigos ascii de 8 bits. Observa las siguientes características:
  - Los 32 primeros caracteres, se llaman caracteres no imprimibles y se utilizaban tradicionalmente para el control de transmisiones.
  - La distancia entre mayúsculas y minúsculas es exactamente 32 caracteres.
  - Hay caracteres que son numéricos, y cuyo valor ascii es el resultado de sumarle 48. Por ejemplo,  $6 + 48 = 54$ . 54 es el código ascii del carácter '6'.

# Ficheros de texto plano

## El código ASCII

sigla en inglés de American Standard Code for Information Interchange  
(Código Estadounidense Estándar para el Intercambio de Información)

Caracteres de control ASCII				Caracteres ASCII imprimibles									ASCII extendido											
DEC	HEX	Símbolo ASCII		DEC	HEX	Símbolo	DEC	HEX	Símbolo	DEC	HEX	Símbolo	DEC	HEX	Símbolo	DEC	HEX	Símbolo	DEC	HEX	Símbolo	DEC	HEX	Símbolo
00	00h	NULL	(carácter nulo)	32	20h	espacio	64	40h	@	96	60h	`	128	80h	Ç	160	A0h	á	192	C0h	Ł	224	E0h	Ó
01	01h	SOH	(inicio encabezado)	33	21h	!	65	41h	A	97	61h	a	129	81h	ü	161	A1h	í	193	C1h	ł	225	E1h	ô
02	02h	STX	(inicio texto)	34	22h	"	66	42h	B	98	62h	b	130	82h	é	162	A2h	ó	194	C2h	Ł	226	E2h	Ò
03	03h	ETX	(fin de texto)	35	23h	#	67	43h	C	99	63h	c	131	83h	â	163	A3h	ú	195	C3h	ł	227	E3h	Ó
04	04h	EOT	(fin transmisión)	36	24h	\$	68	44h	D	100	64h	d	132	84h	ä	164	A4h	ñ	196	C4h	Ł	228	E4h	ô
05	05h	ENQ	(enquiry)	37	25h	%	69	45h	E	101	65h	e	133	85h	å	165	A5h	Ñ	197	C5h	ł	229	E5h	Ö
06	06h	ACK	(acknowledgement)	38	26h	&	70	46h	F	102	66h	f	134	86h	ä	166	A6h	ª	198	C6h	Ł	230	E6h	µ
07	07h	BEL	(timbre)	39	27h	'	71	47h	G	103	67h	g	135	87h	ç	167	A7h	º	199	C7h	ł	231	E7h	þ
08	08h	BS	(retroceso)	40	28h	(	72	48h	H	104	68h	h	136	88h	ê	168	A8h	¿	200	C8h	Ł	232	E8h	þ
09	09h	HT	(tab horizontal)	41	29h	)	73	49h	I	105	69h	i	137	89h	ë	169	A9h	®	201	C9h	ł	233	E9h	Û
10	0Ah	LF	(salto de línea)	42	2Ah	*	74	4Ah	J	106	6Ah	j	138	8Ah	è	170	AAh	¬	202	CAh	Ł	234	EAh	Ü
11	0Bh	VT	(tab vertical)	43	2Bh	+	75	4Bh	K	107	6Bh	k	139	8Bh	ï	171	ABh	½	203	CBh	ł	235	EBh	Ý
12	0Ch	FF	(form feed)	44	2Ch	,	76	4Ch	L	108	6Ch	l	140	8Ch	î	172	ACH	¼	204	CDh	Ł	236	ECh	ÿ
13	0Dh	CR	(retorno de carro)	45	2Dh	-	77	4Dh	M	109	6Dh	m	141	8Dh	ï	173	ADh	ı	205	CDh	ł	237	EDh	ŷ
14	0Eh	SO	(shift Out)	46	2Eh	.	78	4Eh	N	110	6Eh	n	142	8Eh	Ā	174	AEh	«	206	CEh	Ł	238	EEh	ˆ
15	0Fh	SI	(shift In)	47	2Fh	/	79	4Fh	O	111	6Fh	o	143	8Fh	Ă	175	AFh	»	207	CFh	ł	239	EFh	˘
16	10h	DLE	(data link escape)	48	30h	0	80	50h	P	112	70h	p	144	90h	Ė	176	B0h	⋮	208	D0h	Ł	240	F0h	±
17	11h	DC1	(device control 1)	49	31h	1	81	51h	Q	113	71h	q	145	91h	æ	177	B1h	⋮	209	D1h	ł	241	F1h	±
18	12h	DC2	(device control 2)	50	32h	2	82	52h	R	114	72h	r	146	92h	Æ	178	B2h	⋮	210	D2h	Ł	242	F2h	¼
19	13h	DC3	(device control 3)	51	33h	3	83	53h	S	115	73h	s	147	93h	ø	179	B3h	⋮	211	D3h	ł	243	F3h	¾
20	14h	DC4	(device control 4)	52	34h	4	84	54h	T	116	74h	t	148	94h	ò	180	B4h	⋮	212	D4h	Ł	244	F4h	¶
21	15h	NAK	(negative acknowle.)	53	35h	5	85	55h	U	117	75h	u	149	95h	ô	181	B5h	⋮	213	D5h	ł	245	F5h	§
22	16h	SYN	(synchronous idle)	54	36h	6	86	56h	V	118	76h	v	150	96h	ò	182	B6h	⋮	214	D6h	Ł	246	F6h	÷
23	17h	ETB	(end of trans. block)	55	37h	7	87	57h	W	119	77h	w	151	97h	ù	183	B7h	⋮	215	D7h	ł	247	F7h	˙
24	18h	CAN	(cancel)	56	38h	8	88	58h	X	120	78h	x	152	98h	ÿ	184	B8h	©	216	D8h	Ł	248	F8h	˚
25	19h	EM	(end of medium)	57	39h	9	89	59h	Y	121	79h	y	153	99h	Ö	185	B9h	⋮	217	D9h	ł	249	F9h	ˆ
26	1Ah	SUB	(substitute)	58	3Ah	:	90	5Ah	Z	122	7Ah	z	154	9Ah	Ü	186	BAh	⋮	218	DAh	Ł	250	FAh	˘
27	1Bh	ESC	(escape)	59	3Bh	;	91	5Bh	[	123	7Bh	{	155	9Bh	ø	187	BBh	⋮	219	DBh	ł	251	FBh	˙
28	1Ch	FS	(file separator)	60	3Ch	<	92	5Ch	\	124	7Ch		156	9Ch	£	188	BCh	⋮	220	DCh	Ł	252	FCh	˚
29	1Dh	GS	(group separator)	61	3Dh	=	93	5Dh	]	125	7Dh	}	157	9Dh	Ø	189	BDh	¢	221	DDh	ł	253	FDh	˚
30	1Eh	RS	(record separator)	62	3Eh	>	94	5Eh	^	126	7Eh	~	158	9Eh	×	190	BEh	¥	222	DEh	Ł	254	FEh	■
31	1Fh	US	(unit separator)	63	3Fh	?	95	5Fh	_				159	9Fh	f	191	BFh	γ	223	DFh	ł	255	FFh	■

# Ficheros de texto plano

## El código ASCII

Abrir el bloc de notas y teniendo pulsada la tecla alt teclear por ejemplo el número 33. Os debe salir el carácter ! porque el número 33 es su código ASCII. Si ahora hacéis la prueba con el número 34 os debe salir las “

**Y así con los demás.**

**Los ficheros de texto plano son los que están codificados en código ASCII**



# 1.1.1. Tipos de ficheros y formatos

## □ Ficheros binarios

Contienen información codificada en binario para su procesamiento por parte de aplicaciones. Su contenido resulta ilegible en un editor de texto plano como el bloc de notas o el Notepad.

**Ejemplos:** los archivos .exe, .pdf, .docx, .xlsx, .pptx, ficheros de imagen, audio y vídeo ( .jpg, .gif, .mp4, .avi), archivos de sistema (.dll) y ...

# Ficheros binarios

- **Generalmente los ficheros que componen una base de datos son de tipo binario.**

- Deben de tener una estructura lógica y organizada.
- Se sigue un estándar.
- La información de una base de datos se suele guardar en uno o varios ficheros:
  - MariaDB guarda la información de las tablas en los ficheros con extensión frm e ibd.

# Ficheros binarios

- Cuando instalamos el gestor de Bases de Datos MariaDB todos los ficheros binarios que lo componen se almacenan en C:\Archivos de programa\MariaDB

Equipo > Disco local (C:) > Archivos de programa > MariaDB 10.4 >

Nombre	Fecha de modificación	Tipo	Tamaño
bin	13/11/2019 10:45	Carpeta de archivos	
data	17/09/2021 14:46	Carpeta de archivos	
include	13/11/2019 10:45	Carpeta de archivos	
lib	13/11/2019 10:45	Carpeta de archivos	
share	13/11/2019 10:45	Carpeta de archivos	
COPYING	07/11/2019 5:52	Archivo	18 KB
CREDITS	07/11/2019 5:52	Archivo	3 KB
EXCEPTIONS-CLIENT	07/11/2019 5:52	Archivo	9 KB
README.md	07/11/2019 5:52	Archivo de origen ...	4 KB
THIRDPARTY	07/11/2019 5:52	Archivo	85 KB

# Ficheros binarios

- La carpeta **data** contiene todos los ficheros binarios de las Bases de datos que se han creado.

equipo > Disco local (C:) > Archivos de programa > MariaDB 10.4 > data >

Nombre	Fecha de modificación	Tipo
bd_actividad1	18/11/2020 9:39	Carpeta de
bd_actividad2	18/11/2020 8:32	Carpeta de
ejemplo1	16/06/2021 13:27	Carpeta de
empleados1	18/11/2020 9:57	Carpeta de
empresa2021	13/01/2021 8:26	Carpeta de
esquema4	16/11/2020 11:49	Carpeta de
exama	16/12/2020 10:11	Carpeta de
examb	16/12/2020 8:56	Carpeta de
ihg_partido	23/11/2020 11:57	Carpeta de

localhost\empresa2021\ - HeidiSQL 10.2.0.5599

Archivo Editar Buscar Herramientas Ir a Ay

Filtro de bases de Filtro de tablas

localhost

- > bd\_actividad1
- > bd\_actividad2
- > ejemplo1
- > empleados1
- > **empresa2021** 96,0 KiB
- > esquema4
- > exama
- > examb
- > ihg\_partido

# Índice

- **1. Sistemas de almacenamiento de la información**
  - 1.1. Ficheros
    - 1.1.1. Tipos de ficheros y formatos
    - 1.1.2. Ficheros de texto plano
    - 1.1.3. Ficheros binario
  - **1.2. Bases de Datos**
    - 1.2.1. Elementos de una Base de Datos.
    - 1.2.2. Usos de las bases de datos
    - 1.2.3. Evolución y tipos de base de datos
  - 1.3. Los Sistemas Gestores de Base de Datos
    - 1.3.1. Concepto de Sistema Gestor de Base de Datos
    - 1.3.2. Funciones de un SGBD
    - 1.3.3. El lenguaje SQL
    - 1.3.4. Tipos de SGBD

# Bases de Datos

- En el **entorno actual**, la **competitividad** y la **rapidez de maniobra** de una empresa son imprescindibles para su éxito.

Para conseguirlo existe cada vez **una mayor demanda de datos** y, por tanto, más **necesidad de gestionarlos adecuadamente**. Por lo que **las bases de datos (BD)** y **los sistemas gestores de bases de datos (SGBD)** se han convertido en elementos imprescindibles.

# Bases de datos.

- En **informática** se conoce como dato a cualquier **elemento informático que tenga relevancia para un usuario**. Desde su nacimiento, **la informática se ha encargado de proporcionar herramientas que faciliten la manipulación de los datos.**

- **Ejemplos de datos:**

	Nombre		Precio
Fecha de entrega		Apellido 2	
	Apellido 1		
		Fecha de nacimiento	
	DNI		
Nº de pedido			

## 1.2. Base de Datos

# ¿Qué es?

Una **Base de Datos** es una colección de información perteneciente a un mismo contexto (o problema), que está **almacenada de forma organizada** en ficheros binarios. La imagen de la izquierda muestra los ficheros binarios correspondientes a la BD empres2021 que se ha creado usando el SGBD MariaDB.

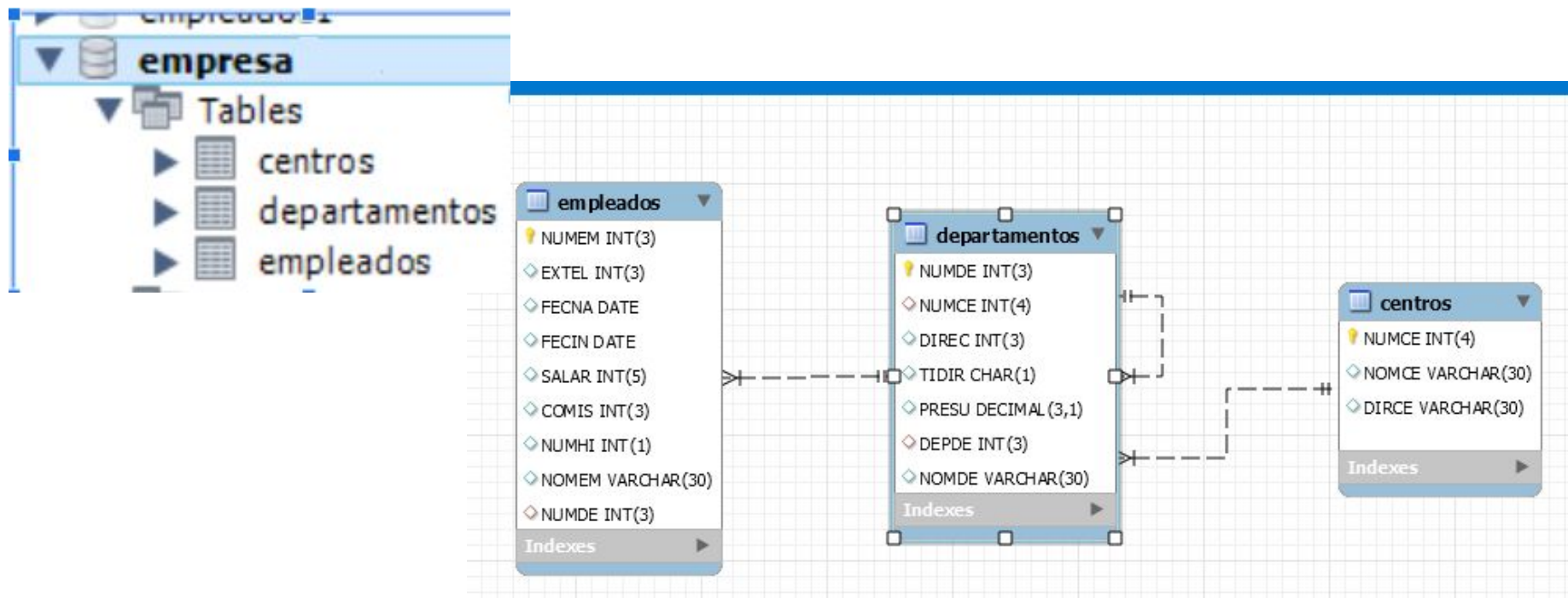
[illegible]



# 1.2. Bases de Datos

## Vamos a estudiar las BD relacionales.

- Una **BD relacional** está organizada mediante tablas. O que es lo mismo la información se guarda en tablas.
- Las tablas se relacionan formando vínculos o relaciones entre ellas como podéis apreciar en la imagen de la derecha.



# 1.2.1 Elementos de una Base de Datos.

- **Actividad.-** Busca los siguientes conceptos y sus definiciones (en BDD). Posteriormente los pondremos en común en la clase y daremos una definición correcta.
- **Tabla, dato, tipo de dato, campo, registro, campo clave primaria (PK).**

## 1.2.1 Elementos de una Base de Datos.

## □ Ejemplos de tablas

Table: centros

**Columns:**

<b>NUMCE</b>	int(4) PK
<b>NOMCE</b>	varchar(30)
<b>DIRCE</b>	varchar(30)

	NUMCE	NOMCE	DIRCE
▶	10	SEDE CENTRAL	C/ ATOCHA, 820, MADRID
	20	RELACIÓN CON CLIENTES	C/ ATOCHA, 405, MADRID
	21	fdgfgfg	rtrtrr
	22	ghghg	ghfghgfh
*	NULL	NULL	NULL

Table: departamentos

**Columns:**

<b>NUMDE</b>	int(3) PK
<b>NUMCE</b>	int(4)
<b>DIREC</b>	int(3)
<b>TIDIR</b>	char(1)
<b>PRESU</b>	decimal(3,1)
<b>DEPDE</b>	int(3)
<b>NOMDE</b>	varchar(30)

[illegible]

# 1.2.1 Elementos de una Base de Datos.

## □ Ejemplos de tablas

Table: **empleados**

Columns:

**NUMEM** int(3) PK  
EXTEL int(3)  
FECNA date  
FECIN date  
SALAR int(5)  
COMIS int(3)  
NUMHI int(1)  
NOMEM varchar(30)  
**NUMDE** int(3)

	NUMEM	EXTEL	FECNA	FECIN	SALAR	COMIS	NUMHI	NOMEM	NUMDE
▶	110	350	1970-11-10	1985-02-15	1800	NULL	3	CESAR	121
	120	840	1968-06-09	1988-10-01	1900	110	1	MARIO	112
	130	810	1965-09-09	1981-02-01	1500	110	2	LUCIANO	112
	150	340	1972-08-10	1997-01-15	2600	NULL	0	JULIO	121
	160	740	1980-07-09	2005-11-11	1800	110	2	AUREO	111
	180	508	1974-10-18	1996-03-18	2800	50	2	MARCOS	110
	190	350	1972-05-12	1992-02-11	1750	NULL	4	JULIANA	121
	210	200	1970-09-28	1999-01-22	1910	NULL	2	PILAR	100
	240	760	1967-02-26	1989-02-24	1700	100	3	LAVINIA	111
	250	250	1976-10-27	1997-03-01	2700	NULL	0	ADRIANA	100
	260	720	1973-12-03	2001-07-12	720	NULL	6	ANTONIO	100

□ Una tabla es un conjunto de **registros** bajo un mismo nombre.

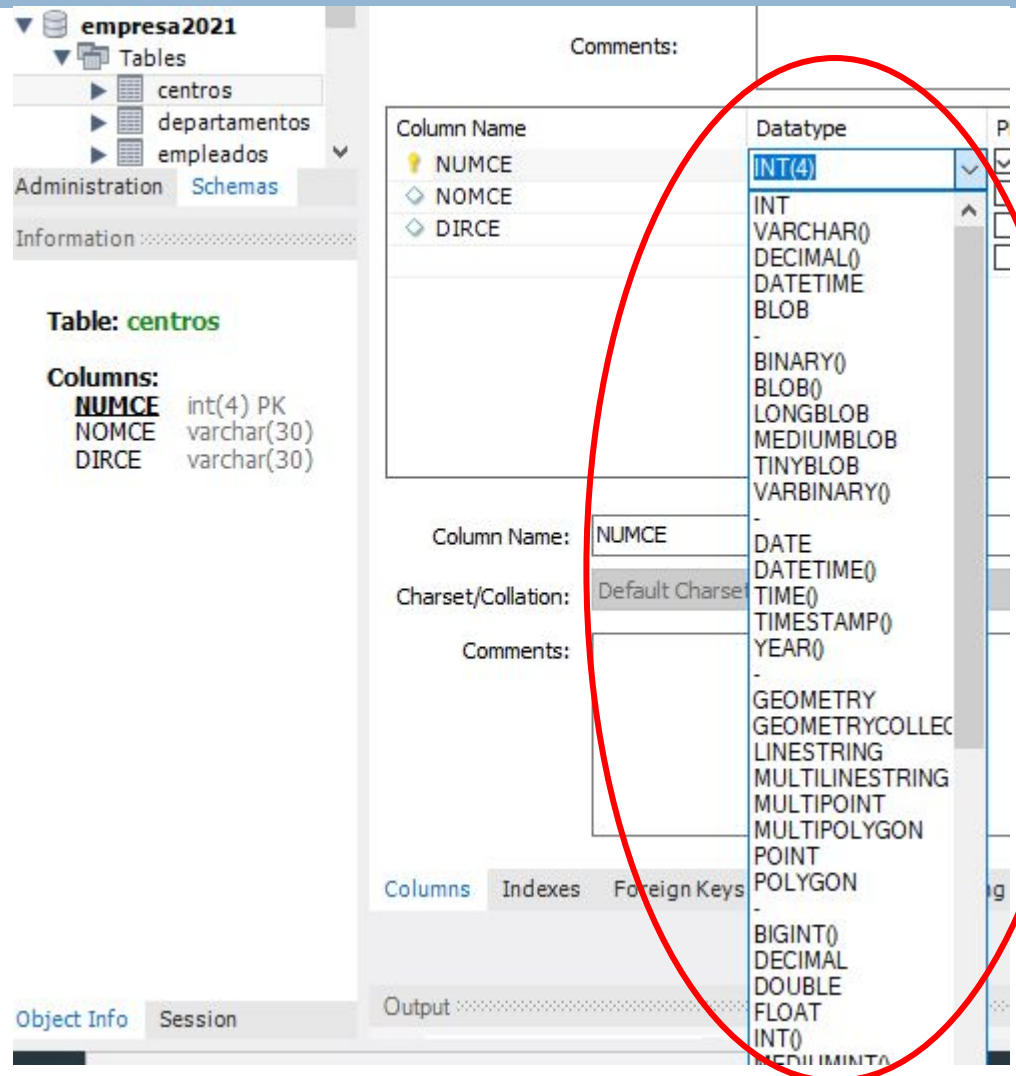
□ Por ejemplo, la tabla empleados almacena los datos de los empleados de una empresa.

# 1.2.1 Elementos de una Base de Datos.

- **Las tablas almacenan datos de manera organizada, ya que cada columna almacena un dato de distinto.**
- **Dato:** Información concreta sobre algún concepto o suceso.
  - Por ejemplo, 1996 es un número que representa un año de nacimiento de una persona.
- **Tipo de Dato:** es la naturaleza del campo.
  - Datos numéricos son con los que se pueden realizar cálculos aritméticos. Como por ejemplo: Números de hijos, precio, stock...
  - Datos alfanuméricos, que son los que contienen caracteres alfabéticos y/o dígitos numéricos. Ej: Apellido1, apellido2, DNI, Código Postal...

# 1.2.1 Elementos de una Base de Datos.

- **Tipo de Dato:**  
La naturaleza del campo, es decir, si va a ser un número entero, un decimal, una cadena de caracteres, una fecha..





# 1.2.1 Elementos de una Base de Datos.

- **Campo:** Identificador para toda una familia de datos. Cada campo pertenece a un tipo de datos.
  - Al campo también se le llama **columna**.
  - Por ejemplo, el campo o columna "FECNA" representa las fechas de nacimiento de los empleados que hay registrados en la tabla. Este campo pertenece al tipo de dato DATE.

The screenshot shows a database management interface with two panes. The left pane displays the table structure for 'empleados', and the right pane shows a 'Result Grid' with data rows. The 'FECNA' column is highlighted with a red circle in both views.

**Table: empleados**

**Columns:**

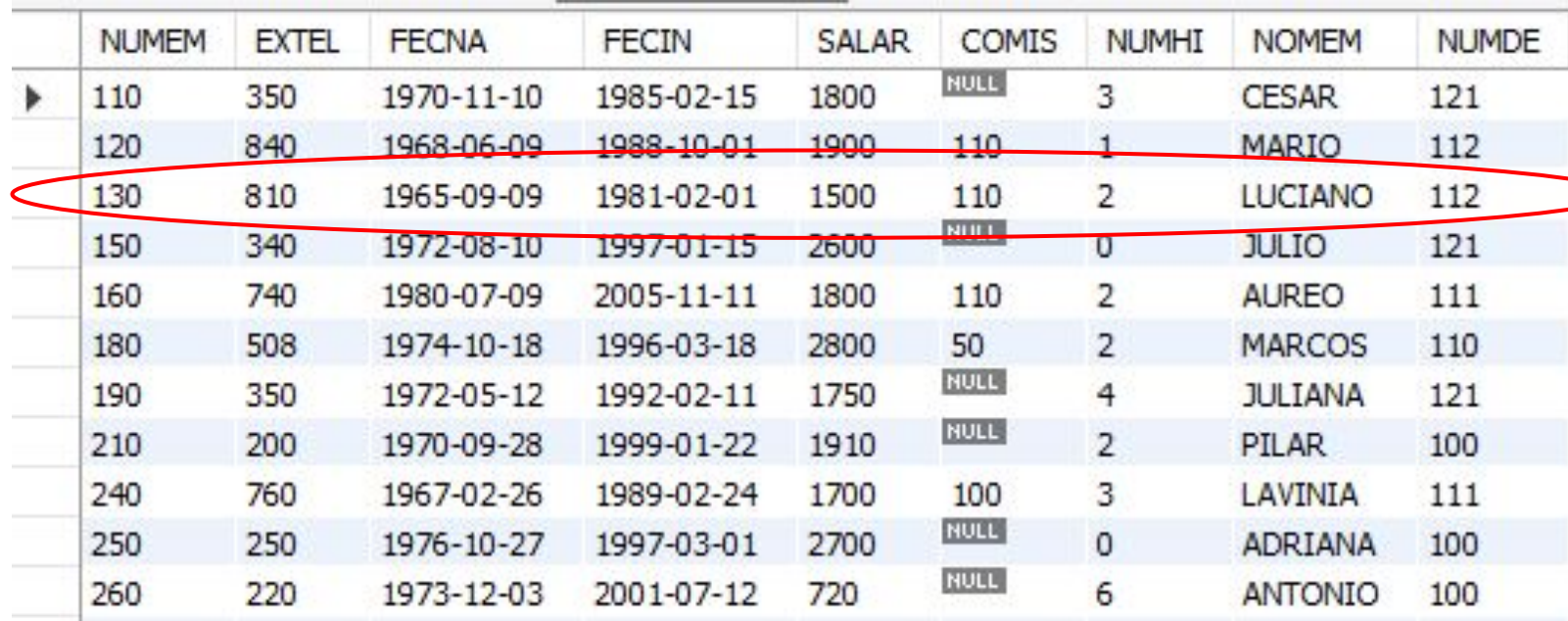
NUMEM	int(3) PK
EXTEL	int(3)
FECNA	date
FECIN	date
SALAR	int(5)
COMIS	int(3)
NUMHI	int(1)
NOMEM	varchar(30)
NUMDE	int(3)

**Result Grid**

	NUMEM	EXTEL	FECNA	FECIN	SALAR	COMIS	NUMHI	NOMEM	NUMDE
▶	110	350	1970-11-10	1985-02-15	1800	NULL	3	CESAR	121
	120	840	1968-06-09	1988-10-01	1900	110	1	MARIO	112
	130	810	1965-09-09	1981-02-01	1500	110	2	LUCIANO	112
	150	340	1972-08-10	1997-01-15	2600	NULL	0	JULIO	121
	160	740	1980-07-09	2005-11-11	1800	110	2	AUREO	111
	180	508	1974-10-18	1996-03-18	2800	50	2	MARCOS	110
	190	350	1972-05-12	1992-02-11	1750	NULL	4	JULIANA	121
	210	200	1970-09-28	1999-01-22	1910	NULL	2	PILAR	100
	240	760	1967-02-26	1989-02-24	1700	100	3	LAVINIA	111
	250	250	1976-10-27	1997-03-01	2700	NULL	0	ADRIANA	100
	260	220	1973-12-03	2001-07-12	720	NULL	6	ANTONIO	100
	270	800	1975-05-21	2003-09-10	1910	80	3	OCTAVIO	112
	280	410	1978-01-10	2010-10-08	1500	NULL	5	DOROTEA	130
	285	620	1979-10-25	2011-02-15	1910	NULL	0	OTILIA	122
	290	810	1977-11-20	2000-02-14	1700	NULL	2	GLORIA	120

# 1.2.1 Elementos de una Base de Datos.

- **Registro:** Es una fila de una tabla
- A los registros también se les llama **tuplas**
  - Lo señalado en la imagen es una fila



	NUMEM	EXTEL	FECNA	FECIN	SALAR	COMIS	NUMHI	NOMEM	NUMDE
▶	110	350	1970-11-10	1985-02-15	1800	NULL	3	CESAR	121
	120	840	1968-06-09	1988-10-01	1900	110	1	MARIO	112
	130	810	1965-09-09	1981-02-01	1500	110	2	LUCIANO	112
	150	340	1972-08-10	1997-01-15	2600	NULL	0	JULIO	121
	160	740	1980-07-09	2005-11-11	1800	110	2	AUREO	111
	180	508	1974-10-18	1996-03-18	2800	50	2	MARCOS	110
	190	350	1972-05-12	1992-02-11	1750	NULL	4	JULIANA	121
	210	200	1970-09-28	1999-01-22	1910	NULL	2	PILAR	100
	240	760	1967-02-26	1989-02-24	1700	100	3	LAVINIA	111
	250	250	1976-10-27	1997-03-01	2700	NULL	0	ADRIANA	100
	260	220	1973-12-03	2001-07-12	720	NULL	6	ANTONIO	100



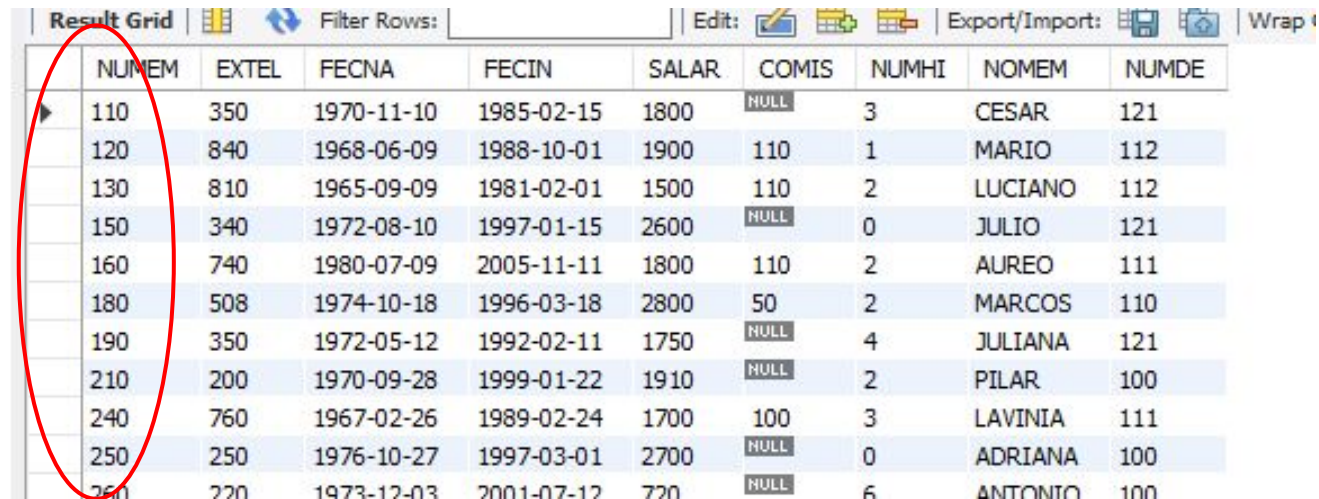
# 1.2.1 Elementos de una Base de Datos.

- ❑ **Campo Clave Primaria (PK):** Es un campo especial que identifica de forma única a cada registro o fila de una tabla. Toda tabla ha de tener un campo clave Primaria (PK)
- ❑ En nuestra tabla empleados el campo clave primaria (PK) es NUMEM, un campo que se ha definido para tal fin.

Table: **empleados**

Columns:

**NUMEM** int(3) PK  
EXTTEL int(3)  
FECNA date  
FECIN date  
SALAR int(5)  
COMIS int(3)  
NUMHI int(1)  
NOMEM varchar(30)  
**NUMDE** int(3)



	NUMEM	EXTTEL	FECNA	FECIN	SALAR	COMIS	NUMHI	NOMEM	NUMDE
	110	350	1970-11-10	1985-02-15	1800	NULL	3	CESAR	121
	120	840	1968-06-09	1988-10-01	1900	110	1	MARIO	112
	130	810	1965-09-09	1981-02-01	1500	110	2	LUCIANO	112
	150	340	1972-08-10	1997-01-15	2600	NULL	0	JULIO	121
	160	740	1980-07-09	2005-11-11	1800	110	2	AUREO	111
	180	508	1974-10-18	1996-03-18	2800	50	2	MARCOS	110
	190	350	1972-05-12	1992-02-11	1750	NULL	4	JULIANA	121
	210	200	1970-09-28	1999-01-22	1910	NULL	2	PILAR	100
	240	760	1967-02-26	1989-02-24	1700	100	3	LAVINIA	111
	250	250	1976-10-27	1997-03-01	2700	NULL	0	ADRIANA	100
	260	720	1973-12-03	2001-07-12	720	NULL	6	ANTONIO	100

# 1.2.2. Usos de las bases de datos

- Algunos ejemplos de sus usos más frecuentes:
  - **Bases de datos Administrativas:** Cualquier empresa necesita registrar y relacionar sus clientes, pedidos, facturas, productos, etc.
  - **Bases de datos de ámbito escolar** para almacenar toda la información de sus profesores, alumnos, cursos, asignaturas, calificaciones,....
  - **Bases de datos Contables:** También es necesario gestionar los pagos, balances de pérdidas y ganancias, patrimonio, declaraciones de hacienda...
  - **Bases de datos para motores de búsquedas:** Por ejemplo Google, tienen una base de datos gigantesca donde almacenan información sobre todos los documentos de Internet. Posteriormente millones de usuarios buscan en la base de datos de estos motores.
  - **Científicas:** Recolección de datos climáticos y medioambientales, químicos, genómicos, geológicos...
  - **Bases de datos para cualquier aplicación web** como por ejemplo facebook, youtube, twiter, moodle...
  - **Bases de datos para las aplicaciones de comercio electrónico.**

# 1.2.2. Usos de las bases de datos

- Algunos ejemplos de sus usos más frecuentes:
  - **Bibliotecas:** Almacenan información bibliográfica, por ejemplo, la famosa tienda virtual amazon o la biblioteca de un instituto.
  - **Censos:** Guardan información demográfica de pueblos, ciudades y países.
  - **Virus:** Los antivirus guardan información sobre todos los potenciales software maliciosos.
  - **Otros muchos usos:** Militares, videojuegos, deportes, etc.
  - ...

## 1.2.2. Usos de las bases de datos

---

- <https://www.evaluandosoftware.com/ranking-bases-datos-las-10-mas-grandes/>

# Índice

- **1. Sistemas de almacenamiento de la información**
  - 1.1. Ficheros
    - 1.1.1. Tipos de ficheros y formatos
    - 1.1.2. Ficheros de texto plano
    - 1.1.3. Ficheros binario
  - 1.2. Bases de Datos
    - 1.2.1. Elementos de una Base de Datos.
    - 1.2.2. Usos de las bases de datos
    - **1.3. Los Sistemas Gestores de Base de Datos**
      - 1.3.1. Concepto de Sistema Gestor de Base de Datos
      - 1.3.2. Funciones de un SGBD
      - 1.3.3. El lenguaje SQL
      - 1.3.4. Tipos de SGBD

# 1.3.1. Concepto sistema gestor de base de datos

- SGBD – Sistema Gestor de Base de Datos
  - Es una *aplicación* que permite a los usuarios **definir, crear y mantener** bases de datos, proporcionando **acceso controlado** a las mismas.
  - Sirve de interfaz entre *usuario* y las *bases de datos*.
  - Por un lado tenemos los **datos organizados** según ciertos criterios.
  - Por otro, un software que nos permite su **gestión**.

# 1.3.1. Concepto sistema gestor de base de datos


- Tipos de Sistemas Gestores de Bases de Datos
  - **SGBD Ofimáticos:** manipulan BD pequeñas orientadas a almacenar datos domésticos o de pequeñas empresas. Ejemplos típicos Microsoft Access y LibreOffice Base
  - **SGBD Corporativos:** tienen la capacidad de gestionar BD enormes, de medianas o grandes empresas con una carga de datos y transacciones que requieren de un servidor de gran capacidad. Un ejemplo típico es MySQL/MariaDB, ORACLE, ...

## 1.3.1. Concepto sistema gestor de base de datos

ORACLE®

  
MySQL®


 SQLite

 mongoDB

  
cassandra



## 1.3.2. Funciones de un SGBD

- 1. Permiten a los usuarios **almacenar, acceder y actualizar** datos de forma sencilla y con un gran rendimiento.
- 2. Garantizan la **integridad** de los datos, respetando las reglas inherentes del SGBD y las restricciones que dicte el programador de la base de datos. *Por ej SGBD relacional **nunca aceptará que una tabla tenga filas o registros con el mismo valor de LLAVE PRIMARIA.***
- 3. Integran **un sistema de seguridad** que garantiza el acceso a la información exclusivamente a aquellos usuarios que dispongan de autorización.
- 4. Proporcionan un **diccionario de metadatos**  `information_schema` que contiene los esquemas de las bases de datos (cómo están estructurados los datos en tablas, registros y campos, las relaciones entre los datos, usuarios, permisos, etc.).

## 1.3.2. Funciones de un SGBD

- 5. Permiten el uso de **transacciones**, garantizan que todas las operaciones de la transacción se realicen correctamente, y en caso de alguna incidencia, deshacen los cambios sin ningún tipo de complicación adicional.

*Ej: Aumentar la nota el 30% a todos los alumnos. Si sólo la aumentará a unos cuantos la BD quedaría incorrecta.*

- 6. Ofrecen **estadísticas** sobre el uso del gestor, registrando operaciones efectuadas, consultas solicitadas, operaciones fallidas y cualquier tipo de incidencia. Es posible de este modo, **monitorizar** el uso de la base de datos, **y permiten analizar hipotéticos malfuncionamientos.**
- 7. Permiten la **conurrencia**, es decir, varios usuarios trabajando sobre un mismo conjunto de datos.

## 1.3.2. Funciones de un SGBD

- 8. Incorporan herramientas para la salvaguarda y **restauración** de la información en caso de desastre. Algunos gestores, tienen sofisticados mecanismos para poder establecer el estado de una base de datos en cualquier punto anterior en el tiempo.

# SGBD relacionales

- **Microsoft SQL Server.**
- **Oracle.**
- **MariaDB**
- **MySQL.**



# SGBD NoSQL



# SGBD NoSQL



mongoDB

- Con la llegada de Internet, suponen una alternativa al modelo clásico de bases de datos SQL relacionales. Empresas como Google, Facebook, Amazon o Twitter comienzan a usarlas debido a problemas que el enfoque relacional no podía solucionar.
- Son idóneas si los datos a almacenar no siguen una estructura fija.

# SGBD NoSQL



- Orientada a documentos (JSON).

## JSON: JavaScript Object Notation

```
{
  "_id" : ObjectId("5e4c60f974a0a6f227bcdec4"),
  "dni" : "12345678Z",
  "nombre" : "Manuel",
  "fNacimiento" : "1896-04-23T10:26:00.996Z",
  "validado" : true,
  "salario" : 1245.65,
  "propiedades" : [
    "casa",
    "coche"
  ],
  "mascotas" : [
    {
      "nombre" : "noa",
      "tipo" : "perro"
    },
    {
      "nombre" : "perico",
      "tipo" : "loro"
    }
  ]
}
```

Diagram illustrating JSON data structure with annotations:

- ObjectId
- texto
- fecha y hora formato en ISO 8601
- booleano
- número (enteros, decimales, ...)
- array
- objeto



# SGBD NoSQL



mongoDB

## Jerarquía del almacenamiento de la información

**SQL**  
↓  
Tablas  
↓  
Filas o registros  
↓  
Columnas o Campos

**noSQL**  
↓  
Colecciones  
↓  
Documentos JSON  
↓  
Propiedades

# 1.3.3. El lenguaje SQL

- **Es la principal herramienta de un SGBD relacional.**
- El usuario, con este lenguaje, realiza preguntas(consultas) al servidor, contestando este a las demandas del usuario.
- Este lenguaje comúnmente se denomina SQL, Structured Query Language.
- Se divide en 4 sublenguajes, el total de todos ellos permite al SGBD cumplir con todas sus funcionalidades

# 1.3.3. El lenguaje SQL

- **Lenguaje DML:** o lenguaje de manipulación de datos (Data Manipulation Language).
  - Este lenguaje permite con 4 sentencias sencillas seleccionar determinados datos (SELECT), insertar datos (INSERT), modificarlos (UPDATE) o incluso borrarlos (DELETE).
- **Lenguaje DDL:** o lenguaje de definición de datos (Data Definition Language).
  - Este lenguaje permite crear toda la estructura de una base de datos ( desde tablas hasta usuarios). Sus cláusulas son del tipo DROP (Eliminar objetos) y CREATE (Crear objetos). En capítulos posteriores se detallará la sintaxis de cada una de estas sentencias.

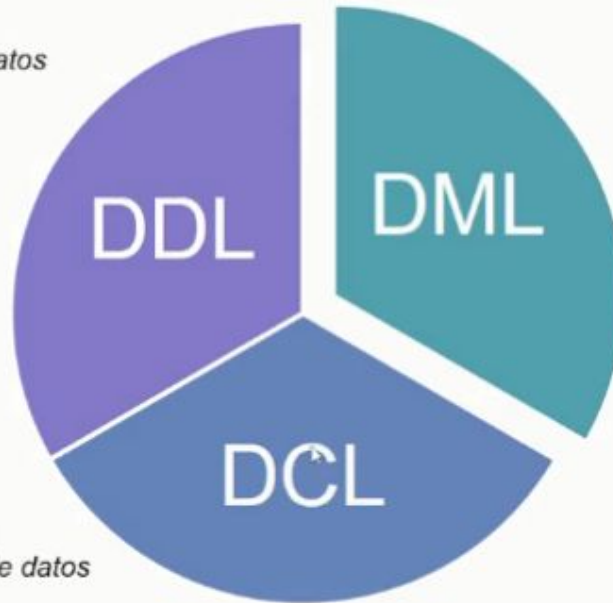
# 1.3.3. El lenguaje SQL

- **Lenguaje DCL:** o lenguaje de control de datos (Data Control Language).
  - Incluye comandos (GRANT y REVOKE) que permiten al administrador gestionar el **acceso a los datos** contenidos en la base de datos.
- **Lenguaje TCL:** o lenguaje de control de transacciones. El propósito de este lenguaje es **permitir ejecutar varios comandos de forma simultánea** como si fuera un comando atómico o indivisible.
  - Si es posible ejecutar todos los comandos, se aplica la transacción (COMMIT), y si, en algún paso de la ejecución, sucede algo inesperado, se pueden deshacer todos los pasos dados (ROLLBACK).

# Sub lenguajes SQL

Data Definition Language  
*Definición de datos*

- CREATE
- ALTER
- DROP



Data Manipulation Language  
*Manipulación de datos*

- SELECT
- INSERT
- UPDATE
- DELETE

Data Control Language  
*Control de datos*

- GRANT
- REVOKE

DDL  
DML  
DCL

TCL: Transaction Control Language

- BEGIN
- COMMIT
- ROLLBACK

## 1.3.4 ¿Qué es CRUD?

□ **El concepto CRUD** está estrechamente vinculado a la gestión de datos digitales. CRUD hace referencia a un acrónimo en el que se reúnen las primeras letras de las cuatro operaciones fundamentales que se realizan en los sistemas gestores de bases de datos:

- **Create (Crear registros)**
- **Read (Leer registros)**
- **Update (Actualizar registros)**
- **Delete (Borrar registros)**

## 1.3.4 ¿Qué es CRUD?

### □ Create (Crear registros)

**Es la operación que permite, una vez creadas las tablas, insertar datos o registros en ellas. Se lleva a cabo a través de la instrucción SQL llamada INSERT**



## 1.3.4 ¿Qué es CRUD?

### □ Read (Leer registros)

**Es la operación que permite seleccionar o leer información de las tablas. Es lo que se llama realizar consultas a las tablas empleando la instrucción `SELECT` que proporciona SQL.**

## 1.3.4 ¿Qué es CRUD?

### □ Update (Actualizar registros)

**Es la operación que permite modificar los datos que hay almacenados en las tablas y se realiza a través de la instrucción UPDATE que proporciona SQL.**

## 1.3.4 ¿Qué es CRUD?

### ❑ Delete (Borrar registros)

**Es la instrucción que permite eliminar datos de las tablas y se realiza utilizando la instrucción DELETE que proporciona SQL.**

# Resumen

- Los conceptos clave de este capítulo son los siguientes:
  - Un fichero es una estructura de información que crea el sistema operativo para almacenar información.
  - El tipo y formato del fichero determina la forma de interpretar la información que contiene. Se clasifican según su contenido, organización y utilidad.
  - Los ficheros de texto plano no requieren un formato para ser interpretado puesto que contienen únicamente texto, sin embargo, los ficheros binarios, cómo almacenan múltiples formas de datos (texto, imágenes, vídeo...) requieren una aplicación que sepa cómo está estructurada la información en ellos.

# Resumen

- Una base de datos está organizada mediante tablas. Las tablas contienen registros de información o filas. Cada registro está compuesto por múltiples campos o columnas. Las tablas se relacionan entre sí para dar cierto sentido a la información almacenada en ellas.
- Una base de datos almacena multitud de objetos como tablas, consultas, índices, vistas, funciones y procedimientos.
- Las bases de datos tienen múltiples aplicaciones, contables, administrativas, motores de búsquedas, científicas, bibliotecas, censos, virus, etc.

# Resumen

- Un SGBD es el conjunto de herramientas software que manipulan bases de datos. Ofrecen a los usuarios funciones como almacenar y acceso a los datos, garantizando la integridad y seguridad de los mismos.
- El lenguaje SQL es una interfaz de programación entre el usuario y la base de datos. Se compone de varios sublenguajes: DML, DDL, DCL y TCL.
- Los gestores de bases de datos que manipulan bases de datos pequeñas se llaman gestores de bases de datos ofimáticas, y los que manipulan bases de datos medianas o grandes se denominan gestores de bases de datos corporativos.

**Vamos a empezar estudiando las BD (Bases de Datos) relacionales.**