

8

Bases de datos



1 La base de datos: estructura y concepto

1.1. Clasificación de bases de datos

1.2. Estructura de una base de datos

2 Gestores de bases de datos: organización, búsqueda y presentación de la información

3 Bases de datos relacionales

3.1. Tipos de datos

4 Bases de datos documentales

5 Creación de tablas con LibreOffice: BASE

5.1. La creación de la base de datos

5.2. Creación de tablas

5.3. Configurando las relaciones entre tablas

6 Selección de información y consultas

6.1. La realización de consultas sencillas

6.2. Análisis del lenguaje SQL

6.3. Consultas compuestas con operativa matemática

7 Formularios e informes

7.1. Empleo de formularios

7.2. Los informes

Lectura motivadora

El huevo del cuco.

A eso de las doce del mediodía del domingo 4 de enero, Martha y yo estábamos acostado un edredón cuando sonó la alarma de mi localizador. Me dirigí a toda prisa al ordenador, comprobé que el *hacker* estaba presente y llamé a Steve White. En menos de un minuto empezó a localizar la llamada.

En lugar de esperar a que Steve realizara su operación, y puesto que el *hacker* estaba en mi ordenador, fui en mi bici al laboratorio para observar desde allí lo que ocurría. Tardé veinte minutos en escalar la colina, pero el *hacker* se lo tomaba con calma y seguía tecleando cuando llegué a la centralita.

Junto a la impresora se había acumulado ya un montón de hojas impresas de dos centímetros de grosor. La primera línea mostraba qué se ocultaba tras el nombre de Sventek. Después de comprobar que ninguno de los técnicos de nuestros sistemas estuviera presente, se dirigió de nuevo a la base de datos Optimis del Pentágono. Pero en esta ocasión no hubo suerte. Hoy no se le autoriza a conectar, fue la respuesta del ordenador del ejército.

¡Por todos los santos, Jim Christy debía de haber tocado los resortes adecuados!

Repasando las hojas impresas, comprobé que el *hacker* había ido de pesca por Milnet. Uno por uno había intentado —en vano— conectar con quince ordenadores de las fuerzas aéreas en lugares como Eglin, Kirtland y la base aérea de Bolling. Llamaba a cada uno de dichos ordenadores, hacía girar un par de veces la manecilla y se dirigía al próximo sistema.

Hasta que llegó a los sistemas de comandancia de las Fuerzas Aéreas, división espacial.

En primer lugar probó su cuenta del sistema con la clave “director”. No hubo suerte.

A continuación lo intentó con “invitado”, pero no surtió efecto.

Acto seguido utilizó “campo”, con la clave “servicio”:

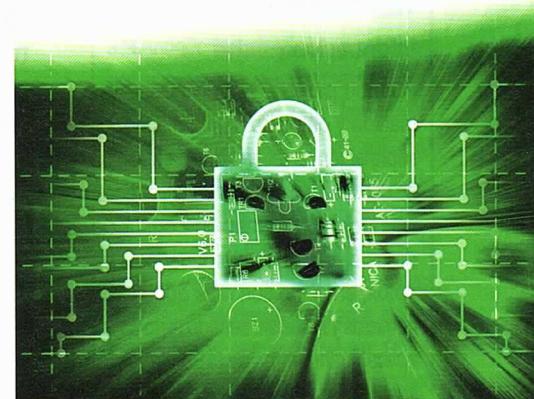
Usuario: CAMPO Clave: SERVICIO

BIENVENIDO AL SISTEMA DE COMANDANCIA DE LAS FUERZAS AÉREAS - DIVISIÓN ESPACIAL VAXA/MS 4.4. ANUNCIO IMPORTANTE. Para cualquier problema del sistema informático diríjanse a los sistemas de información de la sección de servicio al usuario, sitos en el edificio 130, habitación 2359, teléfono 643-2177/AV 833-2177.

Clifford Stoll, 1990.

Preguntas sobre el texto

1. ¿Qué crees que trataba de hacer el *hacker* de la historia?
2. ¿Conoces alguna historia real y actual de ataques a bases de datos de alguna corporación?



El control de acceso a bases de datos es fundamental para la seguridad informática.

Ideas previas

1. ¿Sabes qué es una base de datos?
2. Expón algún ejemplo de acceso a información que hayas realizado recientemente.
3. ¿Consideras que para acceder a los datos de una cuenta bancaria es necesario conectar con una base de datos?
4. ¿Qué entiendes por los términos campo y registro?
5. ¿Para qué sirve una consulta en una base de datos?
6. ¿Qué significa filtrar datos?
7. ¿Crees que se pueden relacionar datos de fuentes diferentes dentro de una misma base de datos? ¿Qué condición crees que deberían tener en común?

1 La base de datos: estructura y concepto

TEN EN CUENTA

Información y datos

Aunque parezcan sinónimos, información y datos no son lo mismo. La información se refiere a la acumulación de datos según determinados patrones. Es decir, la información consta de datos clasificados y ordenados. Los datos, en sí, no son más que meros números y letras sin sentido si no se les aplican parámetros de clasificación. Al proceso de ordenación y clasificación de datos se le denomina **indexación**.

Una de las frases más acuñadas en las últimas décadas es que **vivimos en una sociedad de la información**. Este término, más allá de un calificativo, se refiere a que la información nos inunda por todas partes y, en mayor medida, desde la aparición de Internet. Sin duda, todo ese cúmulo de datos tiene que ser almacenado para su tratamiento, y es ahí donde surge la necesidad de las bases de datos.

Podemos definir una base de datos como el contenedor de aquella información que se desea almacenar con el propósito de ser consultada.

Mientras que un sistema gestor de bases de datos es todo el conjunto de programas capaz de procesar, gestionar y consultar esos datos.

Los ejemplos de aplicación y uso de bases de datos son múltiples, desde los datos almacenados de cada uno de nosotros en la Seguridad Social, hasta los clientes registrados de un operador de telefonía móvil. Pero incluso, dentro del **mundo doméstico**, nosotros mismos podemos construirnos una base de datos para almacenar, por ejemplo, toda la música que tenemos en distintos soportes.



La mayor base de datos es Internet, donde podemos localizar cualquier tipo de información, como por ejemplo obras de arte, literatura, etc.

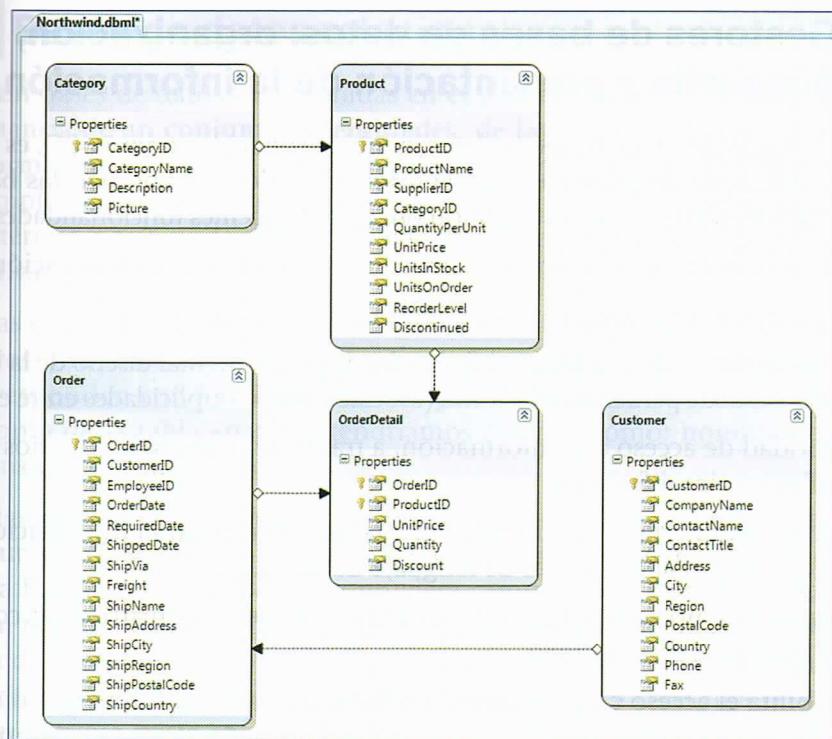
1.1. Clasificación de bases de datos

Existen diversos criterios a la hora de clasificar las bases de datos, pero el más extendido es la forma en que se modela, esquematiza o representa la información. De esta forma podemos citar cuatro modelos: el jerárquico, el relacional, el orientado a objetos y el declarativo:

- **Modelo jerárquico.** Es el más antiguo y modela la información de forma dependiente, a modo de árbol, es decir, un **dato padre se relaciona con un dato hijo y así sucesivamente**. Este tipo de modelo es muy utilizado dentro de dominios DNS y servicios de directorio.
- **Modelo relacional.** Es el modelo más extendido. Una base de datos relacional trata la información como **entidades** y establece unas **relaciones entre dichas entidades**. Por ejemplo, una empresa está formada por un cierto número de empleados, tendríamos una entidad llamada Empresa y otra entidad llamada Empleado, estableciéndose una relación de un empleado que pertenece a una empresa.
- **Modelo orientado a objetos.** Surge gracias a la **programación orientada a objetos** (POO). En él se definen **patrones de almacenamiento llamados clases**, y cada elemento almacenado es un **objeto** con unos datos determinados y unas operaciones particulares para esos datos.
- **Modelo declarativo.** Es utilizado en bases de conocimiento. En este tipo de modelo **ciertos datos son deducidos en función de los datos resultantes de las consultas realizadas**.

VOCABULARIO

Consulta: conocida también por las siglas en inglés DBQ o *data base query*, es una instrucción que permite acceder a aquellos datos contenidos en una base de datos mediante una acción de selección, filtrado, ordenación y clasificación.



CURIOSIDADES

Primeras grabaciones de datos

Aunque el manejo de datos se remonta a la Antigüedad, donde se registraban, por ejemplo, datos sobre meteorología o cosechas; habría que destacar el uso que se les daba en las antiguas bibliotecas, en las que se disponía de un archivo de fichas para localizar los volúmenes. En el mundo moderno, la primera vez que se utilizó una estructura física de datos, diferente de las anotaciones escritas, fue en 1884, cuando Hermann Hollerith empleó tarjetas perforadas para codificar los datos del censo de los Estados Unidos.

Ejemplo de la estructura de tablas de una base de datos.

Actividades

1. Investiga qué tres características otorga la programación orientada a objetos al modelo orientado a objetos.
2. ¿Existe algún otro modelo de base de datos que no haya sido incluido en la clasificación?

1.2. Estructura de una base de datos

Para lograr su objetivo, una base de datos ha de tener una estructura bien formada. Debe estar compuesta por elementos menores que permitan clasificar los datos de forma útil y accesible.

La **estructura lógica de una base de datos** es muy variable en función de los distintos tipos que hemos visto anteriormente. De forma general, es la **representación de la información y de los datos según el esquema de la base**.

En el caso de **bases de datos relacionales**, el primer elemento de la estructura lógica es **una tabla**, donde se acumulan datos bajo un mismo concepto. Una base de datos puede disponer de un número indefinido de tablas. Así, por ejemplo, si se quieren grabar datos sobre la población de un país, existirá una tabla en la que se guarden los datos y características geográficas como las de localización de las ciudades y núcleos de población. Otra tabla contendrá, a su vez, los datos demográficos, indicando el número de varones y mujeres, el índice de natalidad, el índice de mortalidad, etc.

En cuanto a la **estructura física**, el ejemplo más utilizado es la **creación de ficheros de tamaño fijo y lo suficientemente grandes para ir almacenando internamente los datos**. Cuando se quiere utilizar su información el sistema gestor de base de datos carga en distintas estructuras de memoria los datos, manteniendo toda la integridad entre ellos.

VOCABULARIO

Base de conocimiento: es un tipo de base de datos que almacena información con propósitos de aprendizaje o de soluciones empresariales. La información puede ser desde datos relacionales a artículos, documentos, etc.



TICO++

Estructura física y estructura lógica

Existen dos tipos de estructura en una base de datos, la física y la lógica. La estructura física hace referencia a los elementos físicos que se utilizan para registrar datos, es decir, cómo se almacenan los datos en ficheros y directorios. Mientras que la lógica se refiere al esquema desarrollado para clasificar dichos datos.

2 Gestores de bases de datos: organización, búsqueda y presentación de la información

Un sistema gestor de bases de datos, también conocido como SGBD, es una aplicación capaz de gestionar toda la información que almacenan las bases de datos contenidas en él. Esta gestión ofrece las siguientes funcionalidades:

- Almacenamiento, modificación, consulta y borrado de la información.
- Permite la independencia física y lógica de los datos.
- Mantenimiento de la integridad de los datos, ya que un mal diseño de la base de datos puede perder las relaciones (asociaciones) o duplicidades entre ellos.
- Seguridad de acceso a la información, a través de cuentas de usuarios con perfiles o roles diferentes.
- Ofrece una interfaz a los usuarios con la que manipular la información o acceder a otras aplicaciones de lenguaje de datos.
- Asegura que no se pierda la información a través de la realización continua de copias de seguridad.
- Posibilita el acceso concurrente a la información, de tal forma que distintos usuarios pueden acceder a la misma información al mismo tiempo.

El funcionamiento de un sistema gestor de datos se basa en el modelo cliente/servidor, lo que conlleva el uso de los siguientes elementos:

- **Base de datos.** Volumen para el almacenamiento de la información.
- **Servidor.** También conocido con el nombre de **motor de base de datos**, es el encargado de gestionar las consultas.
- **Cliente.** Se trata de una aplicación que utilizarán los usuarios para poder acceder a sus datos.

Estos sistemas de gestión de datos utilizan **un lenguaje que nos permite el acceso, la modificación, el control y la seguridad de la información**. Este lenguaje, llamado lenguaje de datos, se clasifica en tres categorías:

- **Lenguajes de definición de datos** (DDL, *data definition language*). Permite crear, modificar y eliminar los objetos de la base de datos, su estructura, relaciones y restricciones.
- **Lenguaje de control de datos** (DCL, *data control language*). Establece la seguridad a través de permisos a dichos objetos, así como el uso que los usuarios pueden hacer de ellos.
- **Lenguaje de manipulación de datos** (DML, *data manipulation language*). Permite la consulta, inserción, actualización y borrado de los datos.

Gráfico 1. Elementos de un sistema gestor de bases de datos



3 Bases de datos relacionales

Son bases de datos estructuradas en el modelo relacional, es decir, en la existencia de **un conjunto de entidades**, de las cuales se desea almacenar **información** y entre las que se establecen **unas relaciones**. Estas entidades se implementan en la base de datos a través de tablas. Por ejemplo, a partir del término *país*, podemos crear una tabla llamada «países», que sirva para archivarlos.

Las **entidades** disponen de determinados **atributos** que aportan datos relativos a la información que se quiere almacenar de ellas; estos atributos **corresponden con los diferentes campos de datos de una tabla**. Según el ejemplo de la tabla «países», tendríamos campos como: nombre del país, extensión, población, etc.

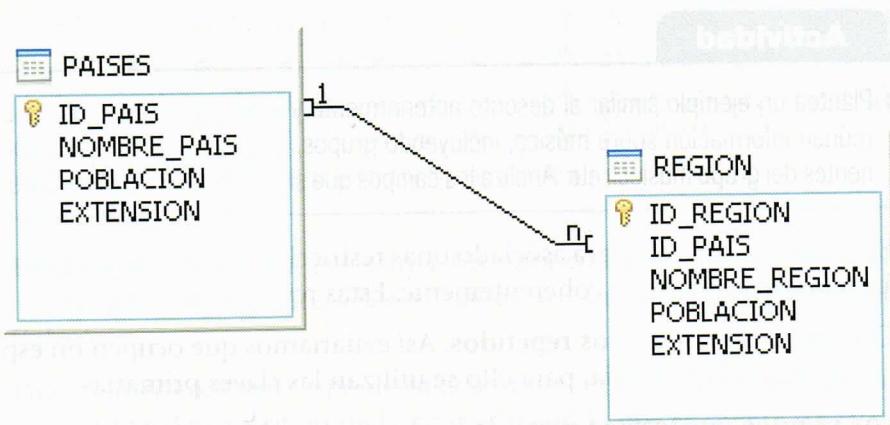
Siguiendo con el ejemplo anterior, pensemos en un rango inferior *región* (de un país); su representación en la base de datos podría ser a través de una tabla llamada «regiones» que contuviera campos con la siguiente información: nombre de la región, país en el que se encuentra, población y extensión de la región, etc. A partir de esta estructura tenemos que pensar cómo relacionar la tabla «países» y la tabla «regiones». Para crear una relación es necesario que exista en ambas tablas un campo que contenga la misma información. En este caso la respuesta es a través del campo nombre del país.



TEN EN CUENTA

Relación entre tablas

A la hora de diseñar una base de datos se analizan todas las posibles entidades y qué información (atributos) deben llevar asociadas. De esta forma, se crean cada una de las tablas de la base de datos. Lo más importante es determinar de qué forma se relacionan dichas entidades y, como consecuencia, cómo se relacionan las tablas.



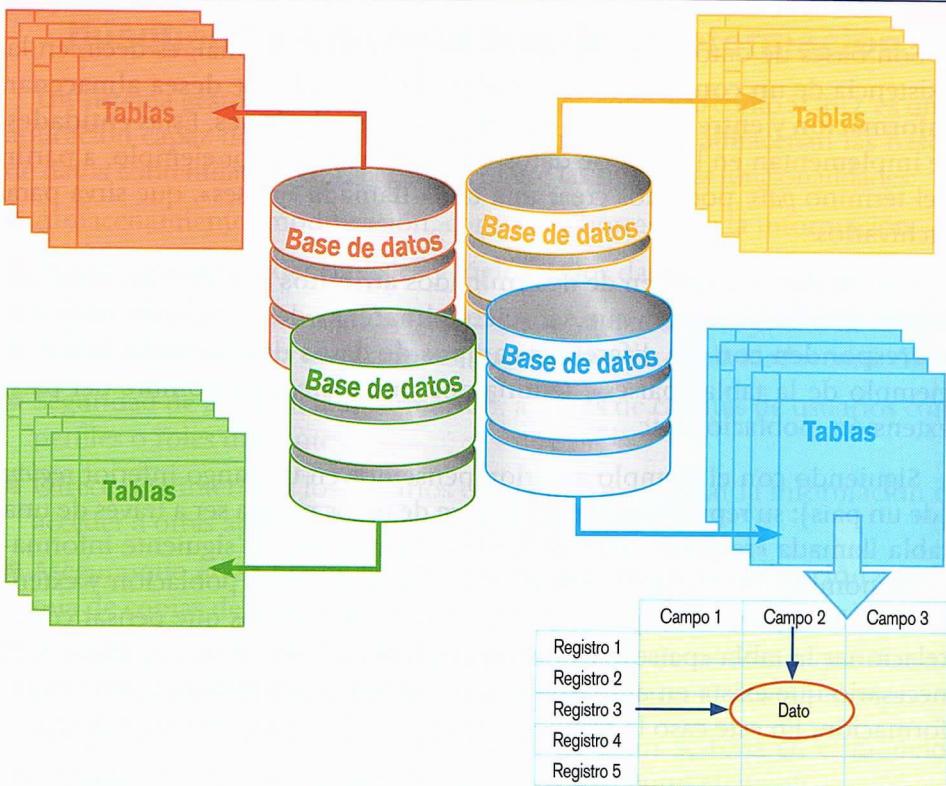
Relaciones entre tablas.

Finalmente, las **tablas** deben rellenarse con datos en forma de **registros**. Un **registro**, también denominado **tupla**, es la **información** contenida en **todos los campos** de una tabla referidos a una entidad de información. Un registro suele coincidir con una fila.

Ejemplo:

En la tabla «Países», tendríamos como registro: España, 504 000 km², 47 millones de habitantes.

Gráfico 2. División estructurada de los elementos de bases de datos



TEN EN CUENTA

Clave principal o primaria

Es el conjunto mínimo de campos que identifican de forma única cada registro de la tabla. Además, será la referencia mediante la cual se crean relaciones con otras tablas. Un campo de clave primaria nunca puede quedar vacío ni contener un dato igual al que existe en otro registro.

Clave ajena

Conjunto de campos de una tabla cuyos valores coinciden con los valores de la clave primaria de otra tabla. Al contrario que la clave primaria, una clave ajena puede contener un valor vacío (NULL).

Actividad

3. Plantea un ejemplo similar al descrito anteriormente y organiza varias tablas que reúnan información sobre música, incluyendo grupos, discos, canciones, componentes del grupo musical, etc. Analiza los campos que sirven de relación entre ellas.

El modelo relacional lleva asociado unas restricciones que ayudan a que la información se almacene coherentemente. Estas restricciones son:

- **Que no existan registros repetidos.** Así evitaríamos que ocupen un espacio de forma innecesaria, para ello se utilizan las claves primarias.
- **Los campos que forman parte de la clave primaria no pueden estar vacíos o nulos.**
- **Debe existir integridad referencial.** Significa que las relaciones entre tablas deben realizarse entre campos que contengan la misma información y, además, sean clave primaria de una tabla con clave ajena de la otra.

3.1. Tipos de datos

Una de las principales características que se le pide a una base de datos es la consistencia de los datos. Esto significa que **debe existir coherencia en el tipo de dato que se definen para cada campo**. A modo de ejemplo, si se configura un campo para alojar datos numéricos, no será posible incluir un carácter alfabético dentro de él. En caso de ocurrir esto, la base de datos ofrecerá un error antes de introducir el valor esperado.

Así, es importante conocer los diferentes tipos de datos para poder configurar cada campo adecuadamente.

- **Tipo carácter.** Representan letras (a-z), símbolos alfanuméricos (j, ", \$, %, &, /, *, etc.) y dígitos numéricos (del 0 al 9). Concatenando caracteres se crean palabras o códigos y con ellos frases. La mayoría de los sistemas operativos emplean 8 o 16 bits para codificar un tipo carácter.

■ **Tipo nulo.** Representa un valor especial que simboliza la ausencia de dato.



El uso de 2 bytes permite codificar los caracteres de muchos alfabetos.

- **Tipo booleano.** Emplea tan solo 1 bit para codificar un 0 o un 1. Normalmente se asocia a los valores cualitativos de TRUE (verdad) o FALSE (falso).

■ **Tipo numérico.** Como tipo de carácter numérico existe un amplio rango de subtipos dependiendo de la precisión con la que se quiera trabajar. El número de bits reservados para codificar un número será proporcional a la precisión. Así, los subtipos numéricos serán:

- **Byte.** Se emplea para números enteros codificados con 1 *byte*, es decir, entre el 0 y el 255.
 - **Short integer.** Se almacenan en 2 *bytes*, es decir, 16 bits, con lo que el rango se amplía de 0 a 65 535.
 - **Integer.** Es el más común de los tipos para números enteros, permitiendo valores negativos y positivos dentro de su propio rango. Emplea 32 bits, es decir 4 octetos con lo que el rango es de -2 147 483 647 hasta el 2 147 483 648 incluyendo el 0.
 - **Long integer.** Emplea 64 bits y se utiliza para grandes números. Esto ofrece un rango enorme de números desde el negativo $9,2234 \cdot 10^{18}$ hasta su simétrico positivo. En algunas aplicaciones este tipo se denomina **entero doble** o simplemente **double**.

TEN EN CUENTA

Múltiples grafías en el mundo

Dada la multiplicidad de tipos de caracteres empleados en los cientos de idiomas del mundo, el uso de 2 bytes para codificar los caracteres alfabéticos permite dar cobertura a todos ellos. Piensa que 1 byte permite la codificación de tan solo 256 tipos de caracteres diferentes y teniendo en cuenta que el estándar de computación es el empleo de bytes, utilizando 2 bytes se podrán codificar los caracteres chinos, árabes, japoneses, cirílicos, tailandeses, coreanos, etc.



CURIOSIDADES

¿Texto o número?

Utilizaremos un campo numérico cuando queramos realizar operaciones aritméticas empleando los datos almacenados en nuestra base de datos; para el resto de casos hablamos de tipo texto. Por ejemplo, el número de teléfono de un cliente sería de tipo texto ya que no se realizarán operaciones matemáticas con él.

Dependiendo de los programas, algunas tienen predefinidas otro tipo de datos, como los subtipos de fecha y hora, los campos *memo* para incorporar largas cadenas de caracteres, etc.



TICO++

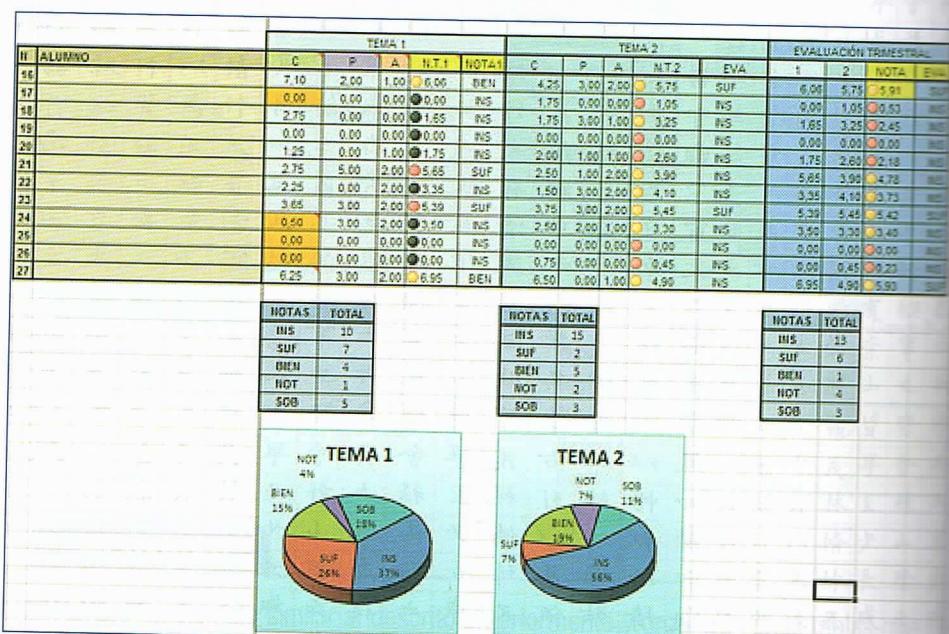
La notación científica

En el cálculo numérico decimal, el empleo de números con más de 9 dígitos se hace muy tedioso. A menudo se emplea la conocida como notación científica, que resulta más apropiada para tener un orden de magnitud del número que se esté tratando. Para indicar un número como el 12345678 de forma exacta, se utiliza el primer carácter, en este caso el número 1 como entero seguido de una coma. A la derecha de la coma se sitúan el resto de números y, dado que son 7 posiciones, se escribe $1,2345678 \times 10^7$. De forma equivalente se puede variar la posición de la coma hacia la derecha, reduciendo entonces el valor del exponente decimal. Así, todos estos números son el mismo: $12,345678 \times 10^6 = 123,45678 \times 10^5 = 1234,5678 \times 10^4 = 12345,678$.

Esta idea de poder mover la coma y jugar con el exponente decimal se emplea para codificar los números decimales, denominándose **notación en coma flotante**.

■ **Tipo fraccionario.** Para conseguir decimales se emplea la coma flotante, de forma que el exponente decimal empleado es un número negativo. Igual que para los números enteros, los números decimales tienen dos subtipos:

- **Float.** Emplea 16 bits para codificar los números.
- **Decimal.** Utiliza 32 bits con lo que resulta, junto con el *long integer*, el tipo numérico de mayor precisión.



Al seleccionar el formato de celda podemos definir el tipo de dato que vamos a introducir.

Actividades

4. ¿Qué ventajas piensas que tiene el hecho de emplear un campo específico de fecha, frente a utilizar tres campos enteros para el día, el mes y el año?
5. ¿Qué subtipo numérico emplearías para almacenar los siguientes números?

27.	89 230.
856.	1,4523*10 ⁷ .
6. Actualmente, el código UNICODE es el más utilizado entre las grandes empresas de informática para el tratamiento de datos. Busca información sobre el código y explica por qué se ha convertido en el estándar de codificación de caracteres.
7. Indica los siguientes números con notación científica y un solo entero:

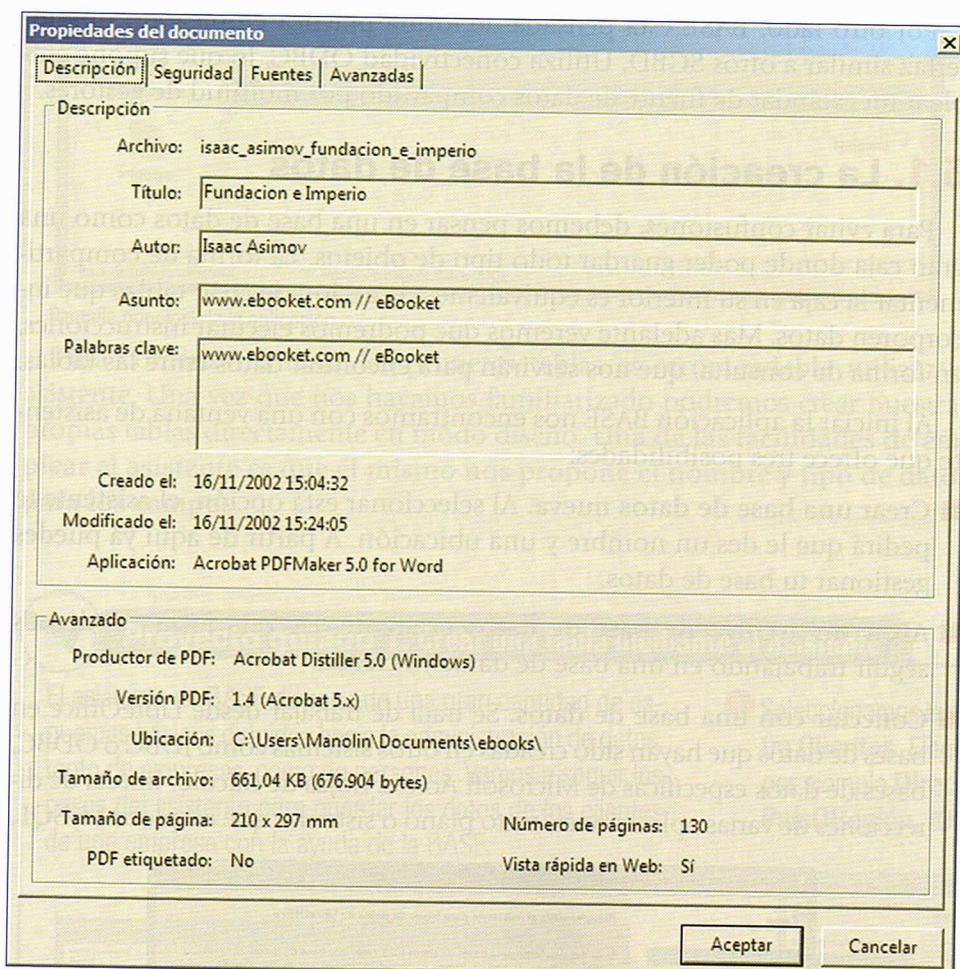
9 827 942 387.	4 954 885,3828.
3 326 737 399 743 762.	0,00383475.
8. Crea una presentación en la nube sobre la historia de las bases de datos. Puedes hacerlo en <http://prezi.com/>. Busca información en la Red desde diferentes fuentes y crea tu propia historia.

4 Bases de datos documentales

El modelo de datos relacionales no siempre cubre las necesidades de almacenamiento y gestión de ciertos tipos de datos. Pensemos, por ejemplo, lo difícil que sería hacer un modelo relacional con artículos de prensa cuya estructura no estuviese definida, debido a que no se conoce su extensión, formatos, etc.

Un sistema documental ha de ser capaz de recuperar documentos selectivamente a partir de los aspectos de su contenido.

Estos documentos se representan en la base de datos, en forma de registros. Esto supone que, si por ejemplo, creamos una base de datos de artículos de prensa, es necesario que además de los campos de autor, medio, fecha, etc., exista un campo textual que contenga el texto del artículo en sí.



Ventana de prioridades de un documento PDF. En ella se observan datos consignados en los campos de propiedades relativos al contenido del documento.

VOCABULARIO

Descriptores: para que los contenidos sean accesibles, y el gestor pueda indicar o suponer cuál es el tema que trata, se utilizan palabras clave llamadas «descriptores», que servirán como vías de acceso al registro.



CURIOSIDADES

Bases de datos documentales en el Estado

En la actualidad, las Administraciones Públicas utilizan las bases de datos documentales para ofrecer a los ciudadanos todo tipo de documentación estatal: jurídica, legislativa, etc.

Actividad

9. Consulta otras bases de datos documentales que existan, tanto con licencia de propietario, como freeware.

Además, es frecuente que la mayoría de los campos estén indexados, es decir, que tengan un determinado orden pre establecido. Este atributo permite que cualquiera de los campos pueda disponer de descriptor, creando una diferencia con el modelo relacional, dado que habrá de permitir que las claves tengan el mismo valor en diferentes registros.

Ejemplo:

Algunos ejemplos de bases de datos documentales son MongoDB, CouchDB o RavenDB.