8 de septiembre de 2016

Cartago valle del cauca

Taller base de Datos I

Lorena Sánchez Bernal

Taller 1 base de datos.

1. Definir término Base de datos.

Una **base de datos** es una colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos es un sistema de archivos electrónico.

Las bases de datos tradicionales se organizan por campos, registros y archivos. Un **campo** es una pieza única de información; un **registro** es un sistema completo de campos; y un **archivo** es una colección de registros. Por ejemplo, una guía de teléfono es análoga a un archivo. Contiene una lista de registros, cada uno de los cuales consiste en tres campos: nombre, dirección, y número de teléfono.

Masadelante.com

1. Haciendo uso de las herramientas text2minmap crear un mapa mental donde se evidencia
2. **Historia de bases de datos**

El término bases de datos fue escuchado por primera vez  en un simposio celebrado en California en 1963.

En una primera aproximación, se puede decir que una base de datos es un conjunto de información relacionada que se encuentra agrupada o estructurada.

Desde el punto de vista informático, una base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos.

Por su parte, un sistema de Gestión de Bases de datos es un tipo de software muy específico dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan; o lo que es lo mismo, una agrupación de programas que sirven para definir, construir y manipular una base de datos, permitiendo así almacenar y posteriormente acceder a los datos de forma rápida y estructurada

Actualmente, las bases de datos están teniendo un impacto decisivo sobre el creciente uso de las computadoras.

Pero para poder entender más profundamente una base de datos cabe entender su historia

**B) Orígenes de las bases de datos.**

Los orígenes de las bases de datos se remontan a la Antigüedad donde ya existían bibliotecas y toda clase de registros. Además también se utilizaban para recoger información sobre las cosechas y censos. Sin embargo, su búsqueda era lenta y poco eficaz y no se contaba con la ayuda de máquinas que pudiesen reemplazar el trabajo manual.

Posteriormente, el uso de las bases de datos se desarrolló a partir de las necesidades de almacenar grandes cantidades de información o datos. Sobre todo, desde la aparición de las primeras computadoras, el concepto de bases de datos ha estado siempre ligado a la informática.

En 1884 Herman Hollerith creó la máquina automática de tarjetas perforadas, siendo nombrado así el primer ingeniero estadístico de la historia. En esta época, los censos se realizaban de forma manual.

Posteriormente, en la década de los cincuenta se da origen a las cintas magnéticas, para automatizar la información y hacer respaldos. Esto sirvió para suplir las necesidades de información de las nuevas industrias. Y a través de este mecanismo se empezaron a automatizar información, con la desventaja de que solo se podía hacer de forma secuencial.

**Década de los 60**

Posteriormente en la época de los sesenta, las computadoras bajaron los precios para que las compañías privadas las pudiesen adquirir; dando paso a que se popularizara el uso de los discos, cosa que fue un adelanto muy efectivo en la época, debido a que a partir de este soporte se podía consultar la información directamente, sin tener que saber la ubicación exacta de los datos.

En esta misma época se dio inicio a las primeras generaciones de bases de datos de red y las bases de datos jerárquicas, ya que era posible guardar estructuras de datos en listas y árboles.

Otro de los principales logros de los años sesenta fue la alianza de IBM y American Airlines para desarrollar SABRE, un sistema operativo que manejaba las reservas de vuelos, transacciones e informaciones sobre los pasajeros de la compañía American Airlines.

Y, posteriormente, en esta misma década, se llevó a cabo el desarrollo del IDS desarrollado por Charles Bachman ( que formaba parte de la CODASYL) supuso la creación de un nuevo tipo de sistema de bases de datos conocido como modelo en red que permitió la creación de un standard en los sistemas de bases de datos gracias a la creación de nuevos lenguajes de sistemas de información.

CODASYL (Conferencie on Data Systems Languages) era un consorcio de industrias informáticas que tenían como objetivo la regularización de un lenguaje de programación estándar que pudiera ser utilizado en multitud de ordenadores.

Los miembros de este consorcio pertenecían a industrias e instituciones gubernamentales relacionadas con el proceso de datos, cuya principal meta era promover un análisis, diseño e implementación de los sistemas de datos más efectivos; y aunque trabajaron en varios lenguajes de programación como COBOL, nunca llegaron a establecer un estándar fijo, proceso que se llevó a cabo por ANSI.

**Década de los 70**

Por lo que respecta a la década de los setenta, Edgar Frank Codd, científico informático ingles conocido por sus aportaciones a la teoría de bases de datos relacionales, definió el modelo relacional a la par que publicó una serie de reglas para los sistemas de datos relacionales a través de su artículo “Un modelo relacional de datos para grandes bancos de datos compartidos”.

Este hecho dio paso al nacimiento de la segunda generación de los Sistemas Gestores de Bases de Datos.

Como consecuencia de esto, durante la década de 1970, Lawrence J. Ellison, más conocido como Larry Ellison, a partir del trabajo de Edgar F. Codd sobre los sistemas de bases de datos relacionales, desarrolló el Relational Software System, o lo que es lo mismo, lo que actualmente se conoce como Oracle Corporation, desarrollando así un sistema de gestión de bases de datos relacional con el mismo nombre que dicha compañía.

Posteriormente en la época de los ochenta también se desarrollará el SQL (Structured Query Language) o lo que es lo mismo un lenguaje de consultas o lenguaje declarativo de acceso a bases de datos relacionales que permite efectuar consultas con el fin de recuperar información de interés de una base de datos y hacer cambios sobre la base de datos de forma sencilla; además de analiza grandes cantidades de información y permitir especificar diversos tipos de operaciones frente a la misma información, a diferencia de las bases de datos de los años ochenta que se diseñaron para aplicaciones de procesamiento de transacciones.

Pero cabe destacar que ORACLE es considerado como uno de los sistemas de bases de datos más completos que existen en el mundo, y aunque su dominio en el mercado de servidores empresariales ha sido casi total hasta hace relativamente poco, actualmente sufre la competencia del SQL Server de la compañía Microsoft y de la oferta de otros Sistemas Administradores de Bases de Datos Relacionales con licencia libre como es el caso de PostgreSQL, MySQL o Firebird que aparecerían posteriormente en la década de 1990.

**Década de los 80**

Por su parte, a principios de los años ochenta comenzó el auge de la comercialización de los sistemas relacionales, y SQL comenzó a ser el estándar de la industria, ya que las bases de datos relacionales con su sistema de tablas (compuesta por filas y columnas) pudieron competir con las bases jerárquicas y de red, como consecuencia de que su nivel de programación era sencillo y su nivel de programación era relativamente bajo.

**Década de los 90**

En la década de 1990 la investigación en bases de datos giró en torno a las bases de datos orientadas a objetos. Las cuales han tenido bastante éxito a la hora de gestionar datos complejos en los campos donde las bases de datos relacionales no han podido desarrollarse de forma eficiente. Así se desarrollaron herramientas como Excel y Access del paquete de Microsoft Office que marcan el inicio de las bases de datos orientadas a objetos.

Así se creó la tercera generación de sistemas gestores de bases de datos.

Fue también en esta época cuando se empezó a modificar la primera publicación hecha por ANSI del lenguaje SQL y se empezó a agregar nuevas expresiones regulares, consultas recursivas, triggers y algunas características orientadas a objetos, que posteriormente en el siglo XXI volverá a sufrir modificaciones introduciendo características de XML, cambios en sus funciones, estandarización del objeto sequence y de las columnas auto numéricas. Y además, se creará la posibilidad de que SQL se pueda utilizar conjuntamente con XML, y se definirá las maneras de cómo importar y guardar datos XML en una base de datos SQL. Dando asi, la posibilidad de proporcionar facilidades que permiten a las aplicaciones integrar el uso de XQuery (lenguaje de consulta XML) para acceso concurrente a datos ordinarios SQL y documentos XML. Y posteriormente, se dará la posibilidad de usar la cláusula order by.

Aunque el boom de la década de los noventa será es el nacimiento del World Wide Web a finales de la década, ya que a través de este se facilitará la consulta a bases de datos.

**Siglo XXI**

En la actualidad, las tres grandes compañías que dominan el mercado de las bases de datos son IBM, Microsoft y Oracle. Por su parte, en el campo de internet, la compañía que genera gran cantidad de información es Google. Aunque existe una gran variedad de software que  permiten crear y manejar bases de datos con gran facilidad, como por ejemplo LINQ, que es un proyecto de Microsoft que agrega consultas nativas semejantes a las de SQL a los lenguajes de la plataforma .NET. El objetivo de este proyecto es permitir que todo el código hecho en Visual Studio sean también orientados a objetos; ya que antes de LINQ la manipulación de datos externos tenía un concepto más estructurado que orientado a objetos; y es por eso que trata de facilitar y estandarizar el acceso a dichos objetos.

Cabe destacar que Visual Studio es un entorno de desarrollo integrado para sistemas operativos Windows que soporta varios lenguajes de programación tales como Visual C++, Visual#, Visual J#, ASP.NET y Visual Basic.NET, aunque se están desarrollando las extensiones necesarias para otros, cuyo objetivo es permitir crear aplicaciones, sitios y aplicaciones web, así como servicios web a cualquier entorno que soporte la plataforma .Net, creando así aplicaciones que intercomuniquen entre estaciones de trabajo, páginas web y dispositivos móviles.

Cibergrafia

<http://histinf.blogs.upv.es/2011/01/04/historia-de-las-bases-de-datos/>

1. **Tipos de bases de dato**

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al contexto que se este manejando, o la utilidad de la misma.

Según la variabilidad de los datos almacenados:

**Bases De Datos Estáticas**

 Estas son bases de datos de solo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones.

**Bases De Datos Dinámicas**

Estas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualizacion, borrado y adición de datos, ademas de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de un supermercado, una farmacia, un videoclub o una empresa.

**Modelos De Bases De Datos**

Además de la clasificación por la función de las bases de datos, estas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una "descripción" de algo conocido como contenedor de datos (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general se refieren a algoritmos , y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos:

**Bases De Datos Jerárquicas**

Éstas son bases de datos que, como su nombre indica, almacenan su información en una estructura jerárquica. En este modelo los datos se organizan en una forma similar a un árbol (visto al revés), en donde un nodo padre de información puede tener varios hijos. El nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se los conoce como hojas.

Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

**Base De Datos De Red**

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de *nodo*: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales.

**Bases De Datos Transaccionales**

Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.

**Bases De Datos Relacionales**

Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970, de los laboratorios Ibm en San José , no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por registros (las filas de una tabla), que representarían las tuplas, y campos(las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, Structured Query Language o Lenguaje Estructurado de Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos

Durante los años 80 la aparición de dBASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que dBase no utilizaba SQL como lenguaje base para su gestión.

**Bases De Datos Multidimensionales**

Son bases de datos ideadas para desarrollar aplicaciones muy concretas, como creación de **cubos olap**Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos relacional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.

**Bases De Datos Orientadas a Objetos**

Este modelo, bastante reciente, y propio de los modelos informáticos enfocado a objetos, trata de almacenar en la base de datos los *objetos* completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

Hrencia- Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

**Bases De Datos Documentales**

Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. Tesaurus es un sistema de índices optimizado para este tipo de bases de datos.

**Bases De Datos Deductivas**

Un sistema de base de datos deductiva, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos. Las bases de datos deductivas son también llamadas bases de datos lógicas, a raíz de que se basa en lógica matemática.

**Gestión de bases de datos distribuida (SGBD)**

La base de datos y el software SGBD pueden estar distribuidos en múltiples sitios conectados por una red.

Hay de dos tipos:

1. Distribuidos homogéneos: utilizan el mismo SGBD en múltiples sitios.

2. Distribuidos heterogéneos: Da lugar a los SGBD federados o sistemas multibase de datos en los que los SGBD participantes tienen cierto grado de autonomía local y tienen acceso a varias bases de datos autónomas preexistentes almacenados en los SGBD, muchos de estos emplean una arquitectura cliente-servidor.

Estas surgen debido a la existencia física de organismos descentralizados. Esto les da la capacidad de unir las bases de datos de cada localidad y acceder así a distintas universidades, sucursales de tiendas, etcétera.

Cibergrafia

<http://basededatos.over-blog.net/article-tipos-de-bases-de-datos-68319538.html>

1. Definir los siguientes conceptos y realizar un mapa conceptual. donde se evidencia como se relacionan todos los términos.
2. Dato
3. Tipo de dato
4. Campo
5. Registro
6. Campo clave
7. Tabla
8. Consulta
9. Índice
10. Vista
11. Informe
12. Guiones
13. Procedimiento
14. Dato.

Un dato es una representación [simbólica](https://es.wikipedia.org/wiki/C%C3%B3digo_(comunicaci%C3%B3n)) (numérica, alfabética, algorítmica, espacial, etc.) de un atributo o variable cuantitativa o cualitativa. Los datos describen hechos empíricos, sucesos y entidades. Es un valor o referente que recibe el computador por diferentes medios, los datos representan la información que el programador manipula en la construcción de una solución o en el desarrollo de un algoritmo.

En [estructura de datos](https://es.wikipedia.org/wiki/Estructura_de_datos), es la parte mínima de la información.

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

1. Tipo de dato

En [ciencias de la computación](https://es.wikipedia.org/wiki/Ciencias_de_la_computaci%C3%B3n) un tipo de dato informático o simplemente tipo es un atributo de los datos que indica al ordenador (y/o al programador) sobre la clase de datos que se va a trabajar. Esto incluye imponer restricciones en los datos, como qué valores pueden tomar y qué operaciones se pueden realizar.

Los tipos de datos comunes son: números enteros, números con signo (negativos), números de [coma flotante](https://es.wikipedia.org/wiki/Coma_flotante) (decimales), cadenas alfanuméricas (y unicodes), estados, etc.

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

1. Que es un campo informático

Es el lugar de almacenamiento de un dato en particular.

1. Registro

Representa un [objeto](https://es.wikipedia.org/wiki/Objeto) único de [datos](https://es.wikipedia.org/wiki/Datos) implícitamente estructurados en una [tabla](https://es.wikipedia.org/wiki/Tabla_(base_de_datos)).

Un registro es un conjunto de [campos](https://es.wikipedia.org/wiki/Campo_(base_de_datos)) que contienen los datos que pertenecen a una misma repetición de entidad.

1. Campo clave.

Es un espacio único que permite identificar y diferenciar a una cosa, empresa o persona de otra.

[http://www.danaconnect.com/wiki/index.php/Glosario\_de\_términos](http://www.danaconnect.com/wiki/index.php/Glosario_de_t%C3%A9rminos).

1. Tabla

Tabla en las [bases de datos](https://es.wikipedia.org/wiki/Bases_de_datos), se refiere al tipo de modelado de datos, donde se guardan los datos recogidos por un programa. Su estructura general se asemeja a la vista general de un programa de [hoja de cálculo](https://es.wikipedia.org/wiki/Hoja_de_c%C3%A1lculo).

Una tabla es utilizada para organizar y presentar información. Las tablas se componen de filas y columnas de celdas que se pueden rellenar con textos y gráficos.

Las tablas se componen de dos estructuras: filas y columnas.

1. Que es una consulta.

En bases de datos, una consulta es el método para acceder a los datos en las bese de datos. Con las consultas se puede modificar, borrar, mostrar y agregar los datos en una base de datos. Para esto se utiliza un lenguaje de consultas. Y el más utilizado es SQL

Alega.com.ar

1. Un índice es una estructura que permite recuperar las filas de una tabla de forma más rápidas además de proporcionar una ordenación distinta a la natural de la tabla. Un índice se define sobre una columna sobré un grupo de columnas, y las filas se ordenaran según sus valores contenidos en esas columnas.
2. Vista

En teoría de bases de datos, una vista es una consulta que se presenta como una tabla (virtual) a partir de un conjunto de tablas en una base de datos relacional. Las vistas tienen la misma estructura que una tabla: filas y columnas. La única diferencia es que sólo se almacena de ellas la definición, no los datos.

1. Informe

Los informes sirven para presentar los datos de una tabla o consulta generalmente para imprimirlos. La diferencia básica con los formularios es que los datos que parecen ene el informe solo se pueden visualizar o imprimir) no se pueden modificar) y en los informes se pueden agrupar mas fácilmente la información y sacar totales por grupos.

1. Guiones

En informática, un *script*, archivo de órdenes, archivo de procesamiento por lotes o, cada vez más aceptado en círculos profesionales

=dicha extensión para ejecutar el guion, por lo tanto, esta suele ser añadida por tradición, o más bien, es útil para que el usuario pueda identificar estos archivos a través de una interfaz de [línea de comandos](https://es.wikipedia.org/wiki/L%C3%ADnea_de_comandos) sin necesidad de abrirlo.

1. Procedimientos

En [informática](http://deconceptos.com/general/informatica) se denomina procedimiento a las instrucciones que otorga un programa para ejecutar una tarea determinada, por ejemplo incorporar una imagen en un texto, implementadas según cada lenguaje de programación.

Lee todo en: [Concepto de procedimiento - Definición en DeConceptos.com](http://deconceptos.com/general/procedimiento#ixzz4JEo5clWz) <http://deconceptos.com/general/procedimiento#ixzz4JEo5clWz>

1. Nombre y explique 5 usos de las bases de datos.
2. Busca en interne 10 bases de datos más grandes del mundo anota su nombre y su tamaño en una hoja de cálculo. Genera un gráfico donde muestra el comparativo de estas bases de datos.
3. Busca en internet la biografía de los siguientes personajes y comente su principal contribución con la evolución de las bases de datos
4. Edgar Frank codd
5. Larry elison
6. Roger kent Summit
7. Bill Gates
8. Michael montey widenius
9. Con el termino Sistemas Gestores de bases de datos, realizar las siguientes opciones.
10. **Definición**

Un **sistema gestor de base de datos** (**SGBD**) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una [base de datos](https://es.wikipedia.org/wiki/Base_de_datos), además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos. Los usuarios pueden acceder a la información usando herramientas específicas de interrogación y de generación de informes, o bien mediante aplicaciones al efecto.

Estos sistemas también proporcionan métodos para mantener la integridad de los datos, para administrar el acceso de usuarios a los datos y para recuperar la información si el sistema se corrompe. Permiten presentar la información de la base de datos en variados formatos. La mayoría incluyen un generador de informes. También pueden incluir un módulo gráfico que permita presentar la información con gráficos y tablas.

Generalmente se accede a los datos mediante lenguajes de interrogación, lenguajes de alto nivel que simplifican la tarea de construir las aplicaciones. También simplifican la interrogación y la presentación de la información. Un SGBD permite controlar el acceso a los datos, asegurar su integridad, gestionar el acceso concurrente a ellos, recuperar los datos tras un fallo del sistema y hacer copias de seguridad. Las bases de datos y los sistemas para su gestión son esenciales para cualquier área de negocio, y deben ser gestionados con esmero.

1. **Enumerar las principales funciones**

Anadir

Borrare

Modificar

Analizar

Generación de informes

Interrogación de informes

Presentación de gráfico y tablas.

1. **Nombres de tipos de SGBD**

**PostgreSQL**  
Es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa y/o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre y/o apoyados por organizaciones comerciales.   
**Características:**  
Alta concurrencia  
Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos.  
Amplia variedad de tipos nativos  
PostgreSQL provee nativamente soporte para:  
  \* Números de precisión arbitraria.   
  \* Texto de largo ilimitado.   
  \* Figuras geométricas (con una variedad de funciones asociadas).   
  \* Direcciones IP (IPv4 e IPv6).   
  \* Bloques de direcciones estilo CIDR.   
  \* Direcciones MAC.   
  \* Arrays.   
Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GiST de PostgreSQL.  
Claves ajenas también denominadas Llaves ajenas o Claves Foráneas (foreign keys).   
Disparadores (triggers): Un disparador o trigger se define como una acción específica que se realiza de acuerdo a un evento, cuando éste ocurra dentro de la base de datos.   
Vistas.   
Integridad transaccional.  
Herencia de tablas.

1. Tipos de datos y operaciones geométricas.   
   Soporte para transacciones distribuidas. Permite a PostgreSQL integrase en un sistema distribuido formado por varios recursos (p.ej, una base de datos PostgreSQL, otra Oracle, una cola de mensajes IBM MQ JMS y un ERP SAP) gestionado por un servidor de aplicaciones donde el éxito ("commit") de la transacción global es el resultado del éxito de las transacciones locales. Nombrar los SGBD más populares

**DB2 express-C**   
  
Es un motor de base de datos perteneciente a la empresa IBM, conteniendo parte de las funcionalidades de la versión de pago DB2 propiedad también de la empresa IBM, este motor de base de datos es gratuita pero no libre (como aquellas que tienen licencia GPL) y cuenta con la ventaja de no tener limite de crecimiento en sus archivos de base de datos.  
  
   
  
**Apache Derby**   
  
Apache Derby es un sistema gestor de base de datos relacional escrito en Java que puede ser empotrado en aplicaciones Java y utilizado para procesos de transacciones online. Tiene un tamaño de 2 MB de espacio en disco. Inicialmente distribuido como IBM Cloudscape, Apache Derby es un proyecto open source licenciado bajo la Apache 2.0 License. Actualmente se distribuye como Sun Java DB.  
  
**Características:**

  \* APIs para JDBC y SQL. Soporta todas las características de SQL92 y la mayoría de SQL99.   
  \* Su código mide alrededor de 2000KB comprimido.   
  \* Soporta cifrado completo, roles y permisos. Además posee SQL SCHEMAS para separar la información en un única base de datos y control completo de

Usuarios.

  \* Soporta internamente procederes, cifrado y compresión.

Trae soporte multilenguaje y localizaciones específicas.   
  \* A partir de la versión 10.4 trae un sistema simple de replicación maestro-esclavo.   
  \* Transacciones y recuperación ante errores ACID.   
  \* Posee tres productos asociados a la marca:   
      \* Derby Embedded Database Engine: El motor propiamente dicho.   
      \* Derby Network Server: Permite convertir Derby en una base de datos que sigue el modelo cliente-servidor tradicional.   
      \* Database Utilities: Un paquete de utilidades.

MySQL  
  
MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.  
Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.  
  
Características

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad.  
Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software ibre. Entre las características disponibles en las últimas versiones se puede destacar:

  \* Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.   
  \* Disponibilidad en gran cantidad de plataformas y sistemas.   
  \* Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferente velocidad de operación, soporte físico, capacidad, distribución geográfica, transacciones...   
  \* Transacciones y claves foráneas.   
  \* Conectividad segura.   
  \* Replicación.   
  \* Búsqueda e indexación de campos de texto.

Cibergrafia.

<https://sites.google.com/site/rebecagonzalezfbd4sc4a/tareas/1-9-sistemas-gestores-de-base-de-datos>

1. Los gestores de bases de datos mas populares son.

Mysql

Postgres

Apache

1. Busca en internet las leyes de codd para el funcionamiento se SGBD relacionales y establece una relación entra cada una de las leyes ce codd

Las **12 reglas de Codd** son un sistema de reglas (numeradas del 0 al 12) propuestas por [Edgar F. Codd](https://es.wikipedia.org/wiki/Edgar_F._Codd), del [modelo relacional](https://es.wikipedia.org/wiki/Modelo_relacional) para las [bases de datos](https://es.wikipedia.org/wiki/Bases_de_datos), diseñado para definir qué requiere un sistema de administración de base de datos.[1](https://es.wikipedia.org/wiki/12_reglas_de_Codd#cite_note-1)

Codd se percató de que existían bases de datos en el mercado las cuales decían ser relacionales, pero lo único que hacían era guardar la información en las [tablas](https://es.wikipedia.org/wiki/Tabla_(base_de_datos)), sin estar estas tablas literalmente [normalizadas](https://es.wikipedia.org/wiki/Normalizaci%C3%B3n_de_bases_de_datos); entonces éste publicó 12 reglas que un verdadero sistema relacional debería tener aunque en la práctica algunas de ellas son difíciles de realizar. Un sistema podrá considerarse "más relacional" cuanto más siga estas reglas.

* Regla 0: El sistema debe ser [relacional](https://es.wikipedia.org/wiki/Base_de_datos_relacional), base de datos y [administrador de sistema](https://es.wikipedia.org/wiki/Administrador_de_sistema). Ese sistema debe utilizar sus facilidades relacionales (exclusivamente) para manejar la base de datos.
* Regla 1:La regla de la información, toda la información en la base de datos es representada unidireccionalmente, por valores en posiciones de las columnas dentro de filas de tablas. Toda la información en una base de datos relacional se representa explícitamente en el nivel Lógico exactamente de una manera: con valores en tablas.
* Regla 2: la regla del acceso garantizado, todos los datos deben ser accesibles sin ambigüedad. Esta regla es esencialmente una nueva exposición del requisito fundamental para las llaves primarias. Dice que cada valor escalar individual en la base de datos debe ser lógicamente direccionable especificando el nombre de la tabla, la columna que lo contiene y la llave primaria.
* Regla 3: Tratamiento sistemático de valores nulos, el [sistema de gestión de base de datos](https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_de_datos) debe permitir que haya campos [nulos](https://es.wikipedia.org/wiki/Null). Debe tener una representación de la "información que falta y de la información inaplicable" que es sistemática, distinto de todos los valores regulares.
* Regla 4: catálogo dinámico en línea basado en el modelo relacional, el sistema debe soportar un catálogo en línea, el catálogo relacional debe ser accesible a los usuarios autorizados. Es decir, los usuarios autorizados deben poder tener acceso a la estructura de la base de datos (catálogo).
* Regla 5: la regla comprensiva del sublenguaje de los [datos](https://es.wikipedia.org/wiki/Dato), el sistema debe soportar por lo menos un lenguaje relacional que:
  1. Tenga una [sintaxis](https://es.wikipedia.org/wiki/Sintaxis) lineal.
  2. Puede ser utilizado de manera interactiva.
  3. Soporte operaciones de definición de datos, operaciones de manipulación de datos (actualización así como la recuperación), seguridad e integridad y operaciones de [administración](https://es.wikipedia.org/wiki/Administraci%C3%B3n) de transacciones.
* Regla 6: regla de [actualización](https://es.wikipedia.org/wiki/Actualizaci%C3%B3n), todas las vistas que son teóricamente actualizables deben ser actualizables por el sistema.
* Regla 7: alto nivel de inserción, actualización y borrado, permitiendo el sistema realizar manipulación de datos de alto nivel, es decir, sobre conjuntos de tuplas. Esto significa que los datos no solo se pueden recuperar de una base de datos relacional de filas múltiples y/o de tablas múltiples, sino también pueden realizarse inserciones, actualización y borrados sobre varias tuplas y/o tablas al mismo tiempo (no sólo sobre registros individuales).
* Regla 8: independencia física de los datos, los programas de aplicación y actividades del terminal permanecen inalterados a nivel lógico cuando quiera que se realicen cambios en las representaciones de almacenamiento o métodos de acceso.
* Regla 9: independencia lógica de los datos, los cambios al [nivel lógico](https://es.wikipedia.org/w/index.php?title=Nivel_l%C3%B3gico&action=edit&redlink=1) (tablas, columnas, filas, etc.) no deben requerir un cambio a una solicitud basada en la estructura. La independencia de datos lógica es más difícil de lograr que la independencia física de datos.
* Regla 10: independencia de la integridad, las limitaciones de la integridad se deben especificar por separado de los programas de la aplicación y se almacenan en la base de datos. Debe ser posible cambiar esas limitaciones sin afectar innecesariamente las aplicaciones existentes.
* Regla 11: independencia de la distribución, la distribución de las porciones de la base de datos a las varias localizaciones debe ser invisible a los usuarios de la base de datos. Los usos existentes deben continuar funcionando con éxito:
  1. cuando una versión distribuida del SGBD se introdujo por primera vez
  2. cuando se distribuyen los datos existentes se redistribuyen en todo el sistema.
* Regla 12: la regla del [orden](https://es.wikipedia.org/wiki/Orden), si el sistema proporciona una interfaz de bajo nivel de registro, aparte de una interfaz relacional, que esa interfaz de bajo nivel no se pueda utilizar para subvertir el sistema, por ejemplo: sin pasar por [seguridad](https://es.wikipedia.org/wiki/Seguridad) relacional o limitación de [integridad](https://es.wikipedia.org/wiki/Integridad_del_mensaje). Esto es debido a que existen sistemas anteriormente no relacionales que añadieron una interfaz relacional, pero con la interfaz nativa existe la posibilidad de trabajar no relacionalmente.

Cibergrafia.

<https://es.wikipedia.org/wiki/12_reglas_de_Codd>

Vamos a realizar un ataque a una aplicación web vulnerable, la forma más común de encontrar errores es a través del uso de una comilla (caracter de escape) en un input, para este caso una variable que se envía por GET:

1. Que es SQL. Indica las revisiones que ha sufrido atreves del tiempo, busca el significado de SQL injection e indica porque un administrador debe protegerse frente a él

SQL o lenguaje de consultas estructuradas

SQL (Structured Query Language) es un lenguaje de programación estándar e interactivo para la obtención de información desde una [base de datos](http://searchdatacenter.techtarget.com/es/definicion/Base-de-datos) y para actualizarla. Aunque SQL es a la vez un ANSI y una norma ISO, muchos productos de bases de datos soportan SQL con extensiones propietarias al lenguaje estándar. Las consultas toman la forma de un lenguaje de comandos que permite seleccionar, insertar, actualizar, averiguar la ubicación de los datos, y más. También hay una interfaz de programación.

Cibergrafia

<http://searchdatacenter.techtarget.com/es/definicion/SQL-o-lenguaje-de-consultas-estructuradas>

**SQL Injection es una vulnerabilidad que permite a un atacante realizar consultas a una base de datos,** se vale de un incorrecto filtrado de la información que se pasa a través de los campos y/o variables que usa un sitio web, es por lo general usada para extraer credenciales y realizar accesos ilegítimos, práctica un tanto neófita, ya que un fallo de este tipo puede llegar a permitir ejecución de comandos en el servidor, subida y lectura de archivos, o peor aún, la alteración total de los datos almacenados.  
  
**Una vulnerabilidad de tipo SQL Injection puede ser explotada tanto a través del método GET como del método POST.**La práctica más común es hacerlo a través del primero, sin embargo hacerlo a través del método POST puede llegar a devolver los mismos resultados. En el mundo de la seguridad informática quienes descubren o investigan vulnerabilidades suelen hacer scripts que automatizan la explotación de las mismas, para el caso de SQL Injection las más comunes son:

[**Sqlmap**](http://sqlmap.org/)**:** Tal vez la más famosa, desarrollada en python por Bernardo Damele y Miroslav Stampar.

[**Havij**](http://itsecteam.com/products/havij-advanced-sql-injection/)**:** Desarrollada por la empresa ITSecTeam.

**SqlNinja:** Desarrollada en Perl, usada para explotar aplicaciones web que usan como back-end a Microsoft SQL Server.

**Cómo es un ataque a una aplicación web vulnerable**

1. Defina los siguientes términos.
2. Lenguaje DML

Lenguaje de Manipulación de Datos (DML)

Lenguaje de Manipulación de Datos (Data Manipulation Language, DML) es un idioma proporcionado por los sistemas gestores de bases de datos que permite a los usuarios de la misma llevar a cabo las tareas de consulta o modificación de los datos contenidos en las Bases de Datos del Sistema Gestor de Bases de Datos. El lenguaje de manipulación de datos más popular hoy día es SQL, usado para recuperar y manipular datos en una base de datos relacional. Otros ejemplos de DML son los usados por bases de datos IMS/DL1, CODASYL u otras.

Elementos del lenguaje de manipulación de datos

Select, Insert, Delete y Update

Clasificación de los DML

Se clasifican en dos grandes grupos:

lenguajes de consulta procedimentales

Lenguajes procedimentales. En este tipo de lenguaje el usuario da instrucciones al sistema para que realice una serie de procedimientos u operaciones en la base de datos para calcular un resultado final.

lenguajes de consulta no procedimentales

En los lenguajes no procedimentales el usuario describe la información deseada sin un procedimiento específico para obtener esa información.

Lenguaje de manipulación de datos (DML)

Un lenguaje de manipulación de datos (Data Manipulation Language, o DML en inglés) es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios llevar a cabo las tareas de consulta o manipulación de los datos, organizados por el modelo de datos adecuado.  
  
El lenguaje de manipulación de datos más popular hoy día es SQL, usado para recuperar y manipular datos en una base de datos relacional. Otros ejemplos de DML son los usados por bases de datos IMS/DL1, CODASYL u otras.  
  
  
1- INSERT  
  
Una sentencia INSERT de SQL agrega uno o más registros a una (y sólo una) tabla en una base de datos relacional.  
  
  
Ejemplo 1 (inserto valores alumno pepe en la materia spd2 a la tabla cursada):  
  
INSERT INTO ''cursada'' (''alumno'', ''materia'') VALUES (''pepe'', ''spd2'')  
  
  
2- UPDATE  
  
Una sentencia UPDATE de SQL es utilizada para modificar los valores de un conjunto de registros existentes en una tabla.  
  
Ejemplo 1 (modifico la materia donde  el alumno sea pepe):  
  
UPDATE ''cursada'' SET ''materia''= ''spd3'' WHERE ''alumno''= ''pepe''  
  
  
3- DELETE  
  
Una sentencia DELETE de SQL borra uno o más registros existentes en una tabla.  
  
Ejemplo 1 (borro todos los valores de las columnas alumno y materia donde la materia sea spd2):  
  
DELETE FROM ''cursada'' WHERE ''materia''= ''spd2''

Cibergrafia. http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro14/53\_\_lenguaje\_de\_manipulacin\_de\_datos\_dml.html

|  |
| --- |
|  |

1. **Lenguaje DDL**

Lenguaje de Definición de Datos (DDL)

Un lenguaje de definición de datos es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios llevar a cabo las tareas de definición de las estructuras que almacenarán los datos así como de los procedimientos o funciones que permitan consultarlos.

Algunas operaciones básicas son:   
  
**CREATE**

Este comando crea un objeto dentro de la base de datos. Puede ser una tabla, vista, índice, trigger, función, procedimiento o cualquier otro objeto que el motor de la base de datos soporte.

Ejemplo 1 (creación de una  base de datos):

CREATE DATABASE prueba;

Ejemplo 2 (creación de una tabla):  
CREATE TABLE nombre tabla (nombre\_columna tipo dato);

CREATE TABLE usuario (codigo int, nombre varchar(60));

**ALTER**

Este comando permite modificar la estructura de un objeto. Se pueden agregar/quitar campos a una tabla, modificar el tipo de un campo, agregar/quitar índices a una tabla, etc.

Ejemplo (agregar columna a una tabla):  
ALTER TABLE nombre tabla (ADD NUEVO\_ CAMPO INT UNSIGNED);

**DROP**

Este comando elimina un objeto de la base de datos. Puede ser una tabla, vista, índice, trigger, función, procedimiento o cualquier otro objeto que el motor de la base de datos soporte. Se puede combinar con la sentencia ALTER.

Ejemplo 1 (eliminar una tabla);

DROP TABLE nombre tabla;  
  
Ejemplo 2 (eliminar una columna):  
ALTER TABLE nombre tabla (DROP COLUMN nombre columna);

Algunas restricciones son:  
  
NOT NULL: Indica que la columna no podrá contener un valor nulo.

CONSTRAINT: Sirve para definir una restricción que se podrá eliminar cuando queramos sin tener que borrar la columna.

**Comandos del DDL y del DML**

**Comandos DDL**

|  |  |
| --- | --- |
| **Comando** | **Descripción** |
| CREATE | Utilizado para crear nuevas tablas, stored procedures e índices |
| DROP | Empleado para eliminar tablas, stored procedures e índices |
| ALTER | Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos |

**Comandos DML**

|  |  |
| --- | --- |
| **Comando** | **Descripción** |
| SELECT | Utilizado para consultar registros de la base de datos que satisfagan un criterio determinado. |
| INSERT | Utilizado para cargar lotes de datos en la base de datos en una única operación. |
| DELETE | Utilizado para eliminar los valores de los campos y registros especificados. |
| UPDATE | Utilizado para modificar registros de una tabla de una base de datos. |

Fuentes de información: <http://www.aulaclic.es/sql/t_8_1.htm>

1. Lenguaje DCL

Un **Lenguaje de Control de Datos** (**DCL** por sus siglas en inglés: **D**ata **C**ontrol **L**anguage) es un lenguaje proporcionado por el [Sistema de Gestión de Base de Datos](https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_base_de_datos) que incluye una serie de comandos SQL que permiten al administrador controlar el acceso a los datos contenidos en la [Base de Datos](https://es.wikipedia.org/wiki/Base_de_Datos)

Algunos ejemplos de comandos incluidos en el DCL son los siguientes:

**GRANT**: Permite dar permisos a uno o varios usuarios o roles para realizar tareas determinadas.

**REVOKE**: Permite eliminar permisos que previamente se han concedido con GRANT

Las tareas sobre las que se pueden conceder o denegar permisos son las siguientes:

CONNECT

SELECT

INSERT

UPDATE

DELETE

USAGE

En [Oracle](https://es.wikipedia.org/wiki/Oracle), la ejecución de un comando DCL implica un [COMMIT](https://es.wikipedia.org/wiki/Commit) de forma implícita. Sin embargo, en [PostgreSQL](https://es.wikipedia.org/wiki/PostgreSQL), la ejecución de un comando DCL forma parte de una transacción, por lo que puede ser deshecha mediante el comando [ROLLBACK](https://es.wikipedia.org/wiki/Rollback).

Cibergrafia <https://es.wikipedia.org/wiki/Lenguaje_de_Control_de_Datos>

1. Lenguaje TCL

Tcl es un [lenguaje interpretado](https://es.wikipedia.org/wiki/Lenguaje_interpretado), y su código puede ser creado y modificado dinámicamente. Sus [reglas sintácticas](https://es.wikipedia.org/wiki/Sintaxis) son extremadamente simples y posee reglas de alcance dinámico. Permite escribir código fácil de mantener. Los "scripts" Tcl son a menudo más compactos y legibles que los programas funcionalmente equivalentes en otros [lenguajes de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n). Es un lenguaje multiplataforma, con intérpretes que se ejecutan sobre [Windows](https://es.wikipedia.org/wiki/Windows), [Linux](https://es.wikipedia.org/wiki/Linux), [UNIX](https://es.wikipedia.org/wiki/Unix), [MacOS](https://es.wikipedia.org/wiki/MacOS) y [OSX](https://es.wikipedia.org/wiki/Mac_OS_X) e incluso microprocesadores PIC.

Todos los elementos de un programa son comandos, incluyendo las estructuras del lenguaje. Dichos comandos se escriben en [notación polaca](https://es.wikipedia.org/wiki/Notaci%C3%B3n_polaca) y pueden ser redefinidos o sobrescritos de manera dinámica.

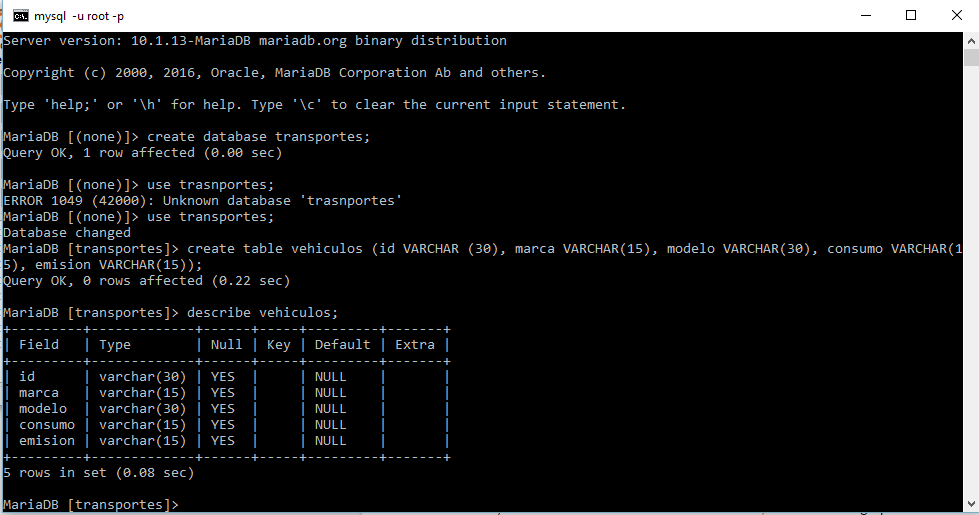
Una característica notable es que los datos son manejados como [cadenas de caracteres](https://es.wikipedia.org/wiki/Cadena_de_caracteres) [Unicode](https://es.wikipedia.org/wiki/Unicode), incluyendo el [código fuente](https://es.wikipedia.org/wiki/C%C3%B3digo_fuente), soportando [Unicode](https://es.wikipedia.org/wiki/Unicode) desde el lanzamiento de la versión 8.1, en el año [1999](https://es.wikipedia.org/wiki/1999).

Una de las características más usadas de Tcl es su extensibilidad. Por ejemplo, si una aplicación requiere algo de funcionalidad no ofrecida por el Tcl estándar, los nuevos comandos de Tcl pueden ser implementados usando el lenguaje [C](https://es.wikipedia.org/wiki/C), un integrado sumamente fácil. Tcl es "extensible" a través de [C](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_C), [C++](https://es.wikipedia.org/wiki/C%2B%2B) y [Java](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java). Mediante una extensión, permite la programación orientada a objetos. Puede extenderse también a [entornos gráficos](https://es.wikipedia.org/wiki/Entorno_gr%C3%A1fico), a través de una interfaz denominada [Tk](https://es.wikipedia.org/wiki/Tk" \o "Tk).

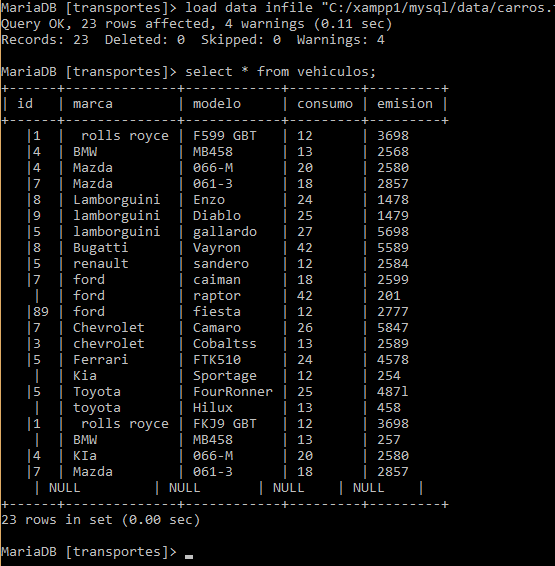
La programación orientada a eventos se realiza sobre "sockets" y archivos, además son posibles los eventos basados en tiempo y los definidos por el usuario.

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

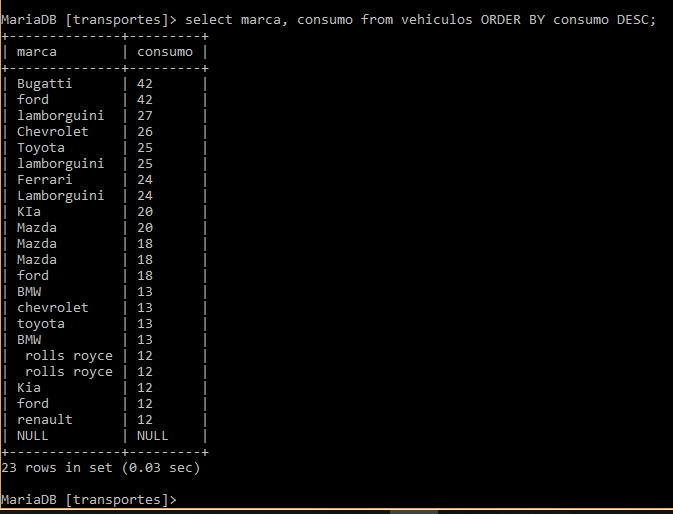
1. Haciendo uso de la herramienta MySQL , escribir el código para hacer las siguientes operaciones en la base de datos.
2. Crear la base de datos. Transportes
3. Crear una tabla llamada vehículos con los campos (id, marca, modelo, consumo, emisiones)



1. Insertar el siguiente registro( 918, Ferrari.f500 GTB, 17,5 240.



1. Insertar mediante el archivo de texto. 20 vehículos en la tabla con diferentes valores.
2. mostrar todos los registros
3. mostrar marca y consumo de la tabla vehículos ordenados de manera descendente por consumo



12) Qué es modelado de datos?

El modelado de datos es el proceso de documentar un diseño de sistema de software complejo como un diagrama de fácil comprensión, usando texto y símbolos para representar la forma en que los datos necesitan fluir. El diagrama se puede utilizar como un mapa para la construcción de un nuevo software o para la reingeniería de una aplicación antigua.

Cibergrafias

<http://searchdatacenter.techtarget.com/es/definicion/Modelado-de-datos>

13) Definir

1. Modelo conceptual:

Representa la vista lógica (o comunitaria), que es independiente del SGBD.

Es parte de las especificaciones de requisitos de usuario y su resultado es el esquema conceptual de la base de datos. Un

esquema conceptual es una descripción de alto nivel de la estructura dela base de datos, independientemente del SGBD que se vaya a utilizar para manipularla. Un modelo conceptual es un lenguaje que se utiliza para describir esquemas conceptuales. El objetivo del diseño conceptual es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar esta información. El diseño lógico parte del esquema conceptual y da como resultado un esquema lógico.

<https://es.scribd.com/doc/33287143/Disen-o-conceptual-y-logico-de-base-de-datos>

Son los orientados a la descripción de estructuras de datos y restricciones de integridad. Se usan fundamentalmente durante la etapa de Análisis de un problema dado y están orientados a representar los elementos que intervienen en ese problema y sus relaciones.

b) Modelo logico

Son orientados a las operaciones más que a la descripción de una realidad. Usualmente están implementados en algún Manejador de Base de Datos. El ejemplo más típico es el Modelo Relacional, que cuenta con la particularidad de contar también con buenas características conceptuales.

<https://es.wikipedia.org/wiki/Modelo_de_datos>

Es una descripción de la estructura de la base de datos en términos delas estructuras de datos que puede procesar un tipo de SGBD. Unmodelo lógico es un lenguaje usado para especificar esquemas lógicos(modelo relacional, modelo de red, etc.). El diseño lógico depende del tipode SGBD que se vaya a utilizar, no depende del producto concreto.

<https://es.scribd.com/doc/33287143/Disen-o-conceptual-y-logico-de-base-de-datos>

1. Modelo Físico:

Son estructuras de datos a bajo nivel implementadas dentro del propio manejador. Ejemplos típicos de estas estructuras son los Árboles B+, las estructuras de Hash, etc.

**En que consiste ?**

– Diseñar el esquema físico a través del refinamiento del esquema

lógico, para su implementación en un DBMS específico.

<https://es.wikipedia.org/wiki/Modelo_de_datos>

**Objetivo:**

– Implementar el modelo lógico empleando de forma eficiente las

estructuras físicas del DBMS, de forma de obtener optimizar la

performance del sistema.

**Problemas planteados:**

– Aplicación de información sobre transacciones y requerimientos de

performance.

– Conocimiento sobre procesamiento de operaciones.

– Elección de estructuras físicas adecuadas para el DBMS.

– Configuración de la BD.

<https://www.fing.edu.uy/inco/cursos/tagsi/TAGSI-DisBD(v2007).pdf>

**Técnicas a aplicar:**

– Pasaje de Mod. Lógicos Mod. Físico.

– Refinamiento del almacenamiento basado en:

» Información sobre volúmenes de datos

» Estructuras físicas disponibles en el DBMS

específico.

» Estrategias de procesamiento de operaciones en el

DMBS específico