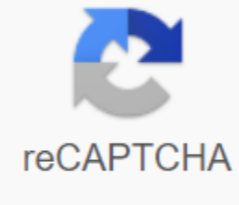




I'm not robot



[Continue](#)

Ejemplos de bases de datos en mysql pdf

A menudo en MySql necesitamos probar la base de datos para trabajar con. Desde su página web podemos descargar algunos para probar. Desde el sitio web de MySql puede descargar varias instancias de base de datos en la sección de base de datos de ejemplo hay 5 bases de datos para descargar. Después de descargarlo, impórtelo en el servidor de base de datos. Antes de imponerlos, debe crear una base de datos desde phpMyAdmin o desde el terminal, desde la consola MySQL. crear un mundo de bases de datos; A continuación, desde la línea de comandos descomprimos el archivo .zip e importamos el archivo world.sql unzip world.sql.zip mysql -u root -p world < world.sql Y eso es todo, programación feliz!! Saludos Alex. Si te gusta el artículo compartirlo en: Añadir comentarios: Inicio » MySQL » Base de datos en mysqlAutor: Escribir Ejemplode.com. Si se pregunta cómo crear una base de datos en mysql? Pero no sabes, tienes que llegar al lugar correcto. Para todos los que nunca han creado una base de datos en mysql y se preguntan cuál es su estructura aquí les dejo una base de datos de ejemplo muy simple en la base de datos mysql:create si no hay 'ejemplos';usar 'ejemplo'; CREATE TABLE 'basedatosmysql' ('id' smallint(7) unsigned NOT NULL auto_increment, 'name' varchar(50) NOT NULL default "", 'category' varchar(50) NOT NULL default "", 'description' text NOT NULL, PRIMARY KEY (id)) ENGINE=MyISAM DEFAULT CHARSET=latin1 PACK_KEYS=0 AUTO_INCREMENT=810 ; Para crear con éxito esta base de datos simple de la manera más simple es a través de phpmyadmin si no lo instala, es libre www.phpmyadmin.net APA citado: (A. 2008.05. Base de datos de ejemplo en mysql. Ejemplode.com de la revista. Consultado el 2008-05-05. desde Instale MySQL con el paquete web rpmvisit para ver el repositorio de escoger e instalar el paquete \$ sudo yum update \$ yum install wget \$ wget \$ md5sum mysql80-community-release-el7-1.noarch.rpm \$ sudo rpm -ivh mysql-mysql-community-release-el7-9.noarch.rpm \$ yum install mysql-serverIniciate MySQL\$ sudo system statusctl mysqldChange default keyEmem the temporary password generated 'temporary password' /var/log/mysqld.logChange sudo key mysql_secure_installation Access MySQL\$ mysqlPuede out a ERROR 1045 (28000): Acceso denegado para los usuarios 'root' - fixed:mysql -u root -pNote: Tenga en cuenta que Lave MySQL y sus sistemas pueden ser diferentes. Consulta de base de datos MySQL > mostrar base de datos; Crear una base de datos mysql > crear una base de datos mibbdd; Tabla de muestras de tabla de consulta; Cree una base de datos tableDefine que se usará para crear la tabla mysql > mediante mibbdd; Crear tabla mysql > CREATE TABLE mitabla(id MEDIUMINT NOT NULL AUTO_INCREMENT, nombre CHAR(30) NOT NULL, age salario entero(30), CLAVE PRIMARY (id)); Introduzca datamysql > INSERT INTO mitabla (nombre, edad, salario) VALOR (Pedro, 24, 21000), (María, 26, 24000), (Juan, 28, 25000), (Luis, 35, 28000), (Mónica, 42, 30000), (Rosa, 43, 25000), (Susana, 45, 3900); Salir de MySql mysql > exit; Uno de los objetivos fundamentales del sistema de información es no sólo disponer de recursos de información, sino también de los mecanismos necesarios para poder encontrar y recuperar estos recursos. De esta manera, las bases de datos se han convertido en un elemento indispensable no sólo para el funcionamiento de grandes motores de búsqueda y recuperación de información a través de la Web, sino también para la creación de sitios web, intranets y otros sistemas de información donde es necesario gestionar grandes o pequeños volúmenes de información. La creación de una base de datos que los usuarios pueden utilizar para consultar y acceder a información que les interesa, la convierte en una herramienta importante de cualquier sistema de información, ya sea en red o fuera de ella. Una base de datos es un conjunto de datos que está organizado y estructurado según un modelo de información específico que refleja no solo los datos en sí, sino también la relación que existe entre ellos. Las bases de datos están diseñadas para fines específicos y deben organizarse con lógica coherente. Los datos pueden ser compartidos por diferentes usuarios y aplicaciones, pero deben mantener su integridad y seguridad más allá de sus interacciones. Las definiciones y descripciones de datos deben ser únicas para minimizar la redundancia y maximizar la independencia en su uso. En la base de datos, las entidades y atributos del mundo real se convierten en registros y campos. Estas entidades pueden ser objetos materiales, así como libros o fotos, pero también personas e incluso conceptos e ideas abstractas. Las entidades tienen atributos y mantienen relaciones entre ellas. Tipo y modelo de procesamiento de datos clásicos: Jerárquico: puede representar dos tipos de relaciones entre datos: relaciones uno a uno y relaciones de uno a varios. El modelo tiene la forma de un árbol al revés donde las ramas pueden tener varios elementos secundarios, pero cada elemento secundario solo puede tener un elemento primario en la red: este modelo permite la representación de muchas personas a muchas personas, de modo que cada registro de la base de datos puede tener alguna ocurrencia más grande de él. El modelo de red evita la redundancia en la información mediante la combinación de tipos de registro denominados conectores. Los modelos de red representan el mundo real mediante registros lógicos que representan entidades y se relacionan entre sí con flechas relacionales: desde la década de 1980 ha sido el modelo más utilizado, lo que permite una mayor eficiencia, flexibilidad y confianza en el procesamiento de datos. La mayoría de las bases de datos y sistemas La información actual se basa en modelos relacionales porque ofrece muchas ventajas sobre los 2 modelos anteriores, como el aprendizaje rápido por parte de usuarios que no tienen un conocimiento profundo de los sistemas basados en dados. Los modelos relacionales representan el mundo real utilizando tablas relacionadas entre sí por columnas comunes. Las bases de datos incluidas en esta categoría se basan en el modelo De relaciones, cuya estructura principal son las relaciones, es decir, tablas bidimensionales que consisten en líneas y columnas. Cada fila, que en terminología relacional se denomina tupla, representa la entidad que queremos memorizar en la base de datos. Las características de cada entidad están determinadas por una columna de relación, denominada atributo. Las entidades con características comunes, es decir, descritas por el mismo conjunto de atributos, formarán parte de la misma relación. Hoy también destaca el uso de bases de datos distribuidas a medida que el uso de arquitecturas de trabajo cliente-servidor y de grupo se vuelve más común. Los principales problemas que surgen del uso de la tecnología de base de datos distribuida se refieren a la duplicación de datos y su integridad al actualizar para ella. Además, el control de la información puede ser perjudicial, ya que se difunde en diferentes ubicaciones geográficas. Los modelos de base de datos orientados a objetos han hecho su aparición recientemente. En este caso, el esquema de base de datos se representa mediante un conjunto de clases que determinan las características y el comportamiento de los objetos que formarán la base de datos. La principal diferencia con respecto al modelo anterior son los datos no positivos. Es decir, con las bases de datos tradicionales, la operación que se debe realizar en los datos se solicita a la aplicación que los utiliza. Con las bases de datos orientadas a objetos, lo contrario es cierto, los objetos memorizados en la base de datos contienen datos y posibles operaciones con esos datos. En cierto sentido, puede pensar en objetos como datos que han sido dotados de cierta inteligencia que les permite averiguar cómo comportarse, sin tener que depender de aplicaciones externas. La arquitectura del sistema de base de datos se basa en un nivel físico de 3 niveles: es el nivel de abstracción más bajo y el nivel real de datos almacenados. Este nivel determina cómo se almacenan los datos en medios físicos, ya sea en registros o de otras maneras, así como métodos de acceso. Este nivel está asociado con la representación de los datos, que llamamos el Esquema Físico de Nivel Conceptual: es el que corresponde a la vista de la base de datos desde un punto de vista del mundo real. Es decir, se trata con la entidad u objeto representado, independientemente de cómo se represente o almacene. Esta es una representación de creado por la organización, que recopila una parte de los datos de diferentes requisitos de usuario y aplicaciones parciales. Incluye definiciones de datos y relaciones en el medio. Estos niveles tienen un nivel de visión de esquema conceptual relacionado: forman parte de un esquema conceptual. El nivel conceptual presenta toda la base de datos, mientras que los usuarios generalmente solo tienen acceso a un paquete de base de datos pequeño. El nivel de visión es responsable de dividir esta trama. Un ejemplo es el caso de un empleado de una organización que tiene acceso a su visión de nómina, pero no de sus compañeros. El esquema asociado a este nivel es otro esquema de visión del autor que utiliza niveles internos, niveles conceptuales y nombres de nivel externo para hacer referencia a este mismo nivel: este modelo de arquitectura le permite establecer el principio de independencia de datos, ya sea lógica o física. La independencia lógica significa que los cambios en los esquemas lógicos no deben afectar a los esquemas externos que no utilizan datos modificados. La independencia física significa que los esquemas lógicos no se verán afectados por los cambios realizados en los esquemas internos, según el modo de acceso, etc. Al diseñar una base de datos, el modelo de datos (instrumento) y, por otro lado, el esquema de datos (el resultado de la implementación de ese modelo) deben distinguirse en un lado. Un modelo de datos es un conjunto de conceptos, reglas y convenciones que nos permiten describir los datos del universo que habla. Un esquema es una estructura de datos obtenida después de aplicar ese modelo. Los modelos de datos son un problema fundamental al diseñar bases de datos. Jesús Tramullas en el Sistema de Bases de Datos recopila 3 modelos básicos: Modelos lógicos basados en objetos: los dos más amplios son los modelos de relación de entidades y los modelos orientados a objetos. El modelo de relación de entidad (E-R) se basa en la percepción de un mundo formado por objetos, las llamadas entidades y las relaciones entre ellos. Las entidades difieren entre sí a través de atributos. Orientado a objetos también se basa en objetos, que contienen valores y métodos, entendidos como comandos que actúan sobre valores, en niveles anidados. Los objetos se agrupan en clases asociadas con el envío de mensajes. Algunos autores definen estos modelos como modelos lógicos de modelos semánticos basados en registros: los más extendidos son relacionales, mientras que los otros dos, jerárquicos y

de red, se invierten. Estos modelos se utilizan para definir la estructura lógica global de las bases de datos, organizadas en registros de formato fijo de diferentes tipos. Los modelos relacionales representan datos y relaciones mediante tablas bidimensionales, que contienen datos recuperados del dominio asociado. El modelo de red consiste en una colección de por un puntero o liga de red arbitrario. Los modelos jerárquicos son similares a los modelos de red, pero los registros se establecen como colecciones de árbol. Algunos autores definen estos modelos como modelos de datos clásicos Modelos de datos físicos: se utilizan muy pocos, están unificando modelos y modelos de memoria elemental. Algunos autores definen este modelo como un modelo de datos primitivo El propósito del modelo de datos es, por un lado, inaugurar y definir estructuras que puedan representar los datos y, por otro, diseñar bases de datos. En el diseño de la base de datos, es necesario tener en cuenta diferentes consideraciones, entre ellas: Velocidad de acceso Tamaño de la información Tipo de información Facilidad de acceso a la información Facilidad de acceso a la información Facilidad de extraer la información necesaria Comportamiento del sistema de gestión de la base de datos con cada tipo de información Para capturar tres niveles en el enfoque o modelo de datos seleccionado, programa o aplicación que actúa como una interfaz entre el usuario, modelo y sistema físico se requiere. Este es el rol del sistema de administración de bases de datos. El sistema de administración de bases de datos, también denominado DBMS (Data Base Management System), no es más que un paquete de software, que se ejecuta en un equipo host que corta los centros de acceso a datos y actúa como una interfaz entre datos físicos y de usuario. El sistema es capaz de realizar funciones como la creación y gestión de la propia base de datos, el control de acceso y la manipulación de datos según las necesidades de cada usuario. Por lo tanto, las bases de datos pueden crearse, administrarse y administrarse mediante una serie de aplicaciones denominadas Sistemas de administración de bases de datos (SGBD). De esta manera, cada usuario puede crear una base de datos siguiendo parámetros estándar que permiten que otro usuario la consulte. El sistema de gestión de bases de datos consta de: Data Engine People Program Este sistema de gestión incluye una serie de rutinas de software interconectadas que son responsables de una tarea en particular. Jesús Tramullas recopila componentes que deben tener un sistema de gestión de bases de datos ideal: Lenguaje de definición de esquema conceptual Sistema de lenguaje De diccionario datos Datos Reglas de datos Datos De lenguaje Base de datos de definición de esquema de lenguaje simétrico Una estructura de almacenamiento de datos lógicos Una Transformación de Módulo Lógico a Subsistema Físico Privacidad Subistema de Integridad Subsistema de Uso General Subsistema de Copia de Seguridad y Recuperación de Subsistema de Programa de Aplicación de Subsistema SGBD Common Purpose Consulta Lenguaje de Consulta tiene dos funciones principales son: Estructura para almacenar la manipulación de datos Además, SGBD debe combinar dos tipos de lenguajes como herramientas básicas: uno para la definición de datos y otro para la manipulación de datos. El primero se denomina lenguaje de definición de datos (DLL) y proporciona un medio para determinar con precisión los datos, determinando diferentes estructuras. El segundo se conoce como Lenguaje de manipulación/administración de datos (DML) y facilita a los usuarios el acceso y la manipulación de datos. Antes de la base de datos, los equipos trabajaban en lo que se conocía como el sistema de procesamiento de archivos, donde los datos eran almacenados para su procesamiento por programas escritos principalmente para cada clase de archivo; Esto hace que los sistemas monolíticos y difíciles sean tratados con una serie de inconvenientes tales como: Dificultad para acceder a ciertos datos de información Datos de aislamiento Problemas de atomidad en el funcionamiento Problemas de acceso simultáneos Falta de seguridad Etc. Para resolver este problema, se desarrolló un sistema de gestión de bases de datos cuyas características hacen que el sistema sea mucho más eficiente que el sistema de procesamiento de archivos. Algunas de estas mejoras se basan en la ausencia de una copia de los datos en los que todos los programas, denominada redundancia mínima, para eliminar las incoherencias de los datos; o la capacidad de interactuar en un entorno cliente/servidor donde el cliente o usuario (ya sea en la intranet o desde Internet) puede trabajar con un único conjunto de datos hospedado en el servidor y donde varios clientes pueden trabajar al mismo tiempo. Estas son solo algunas de las características que tienen los modelos de bases de datos relacionales y hay varios motores de base de datos que le permiten trabajar con bases de datos existentes o creando una nueva con todas las capacidades de red. Muchas empresas han recurrido al desarrollo de sistemas de gestión de bases de datos como Oracle, Informix, PostgreSQL, Sybase, Microsoft, etc. En el diseño de la base de datos, el tamaño de la base de datos es un problema fundamental, ya que afecta a la eficiencia del almacenamiento y a la agilidad en la búsqueda y la recuperación. Dado que los datos pueden existir en cualquier morfología (texto, imágenes, audio, etc.), en algunos casos se deben utilizar técnicas de compresión de datos para reducir el espacio y el tamaño de la base de datos, pero estas técnicas de compresión deben ir acompañadas de técnicas de indexación adecuadas que permitan recuperar dichos datos. Tradicionalmente se han hecho diferencias claras entre 2 tipos de bases de datos: Bases de datos referencia: una base de datos que ofrece registros que a su vez son representaciones del documento textual, y entre las que podemos distinguir: Bibliografía: bases de datos cuyo contenido son directorios de registros bibliográficos; aquellas cuyo contenido se refiere a descripciones de otros recursos informativos, como directorios de bases de datos. Fuente de la base de datos: éstas son las que ofrecen todo el documento, no su representación, y entre ellas son distinguibles: Numérica: contienen información de tipo numérico textual: contienen el texto completo de los documentos mixtos: combinan ambos tipos de información sin embargo, el desarrollo de aplicaciones multimedia ha cambiado el concepto tradicional de la base de datos textual, que sólo integra elementos de información numérica y numérica. Con multimedia, otros tipos de objetos han hecho su aparición: gráficos, sonido y audiovisuales que tienen el mismo entorno que los datos textuales y numéricos. La aparición de CD-ROM y otros medios ópticos como DVDs con alta capacidad de almacenamiento de datos y altas velocidades de lectura, ha hecho posible el desarrollo de bases de datos multimedia. Al mismo tiempo, los formatos de archivo gráficos, de audio y de vídeo se han estandarizado gradualmente, y se han perfeccionado los métodos de compresión para este tipo de datos, ya que ocupan grandes cantidades de memoria. Además, el desarrollo del hipertexto, al permitir la conectividad entre referencias y documentos de origen a través de vínculos, también ha violado los límites entre los documentos primarios y los documentos secundarios, combinando datos de referencia y acceso directo a documentos de origen en el mismo espacio. Hasta hace poco, las bases de datos eran productos comerciales desarrollados y gestionados por ciertas empresas que las comercializaban en formato CD-ROM o las distribuían para su consulta, previa pago, en línea a través de telnet. La mayoría son bases de datos bibliográficas o legales. Las organizaciones también tienen sus propias bases de datos integradas en los sistemas de gestión más famosos para crear y mantener bases de datos como FileMaker, Knosys, Access, etc. Actualmente todos estos programas se han visto obligados a ser compatibles con la Web y ofrecen la posibilidad de acceder, buscar y recuperar datos online a través del protocolo http. De esta manera, se han desarrollado y comercializado una serie de herramientas y aplicaciones, comúnmente llamadas pasarelas web, que permiten consultar con bases de datos antiguas - o nuevas - creadas con este sistema de gestión a través de navegadores web, pero también, la existencia de estas herramientas ha apoyado el hecho de que cualquier persona puede publicar actualmente su propia base de datos en su página web, para que pueda ser consultada por cualquier usuario de la red. Esta puerta de enlace no es más que una herramienta de software que permite la comunicación entre servidores web Base. Así, la World Wide Web se ha convertido en su propia interfaz de acceso a datos que puede ser utilizada por cualquier usuario. Los nodos de hipertexto no se limitan a incluir texto, imágenes o sonido, sino también scripts y otros elementos como interfaces de programación de aplicaciones (API) o controladores para la conectividad de bases de datos y el intercambio de información, como: OLE (Open Database Connectivity) CGI (Common Gateway Interface) JDBC (Java Data Base Connectivity) SQL LINKS, etc. Todos estos objetos son la existencia que permite la existencia de elementos y documentos dinámicos y que aportan un verdadero dinamismo al hipertexto. Este es un componente que debe diseñarse en una interfaz de programación para el acceso a datos de hipertexto y consiste en el diseño y la interfaz de objetos de acceso a datos e interfaces de programación de aplicaciones. De esta manera, las bases de datos se pueden crear utilizando sistemas de administración de bases de datos y aplicaciones como: Microsoft Access Oracle Sybase MySQL MSQL o SQL Server, etc. Mediante una serie de herramientas de acceso (CGI, DAO, ODBC, etc.) y desde diferentes entornos de desarrollo, estos datos son accesibles a través de la Web a cualquier usuario que desee crear una consulta en línea. El acceso a los datos se puede realizar mediante una variedad de tecnologías web, entre ellas: CGI: (Common Gateway Interface) definir protocolos que permiten a Web Server (HTTP) comunicarse con programas o scripts externos. Los programas CGI funcionan en servidores Web y se pueden implementar utilizando diferentes lenguajes de programación (COBOL, C, Perl, etc.). Para que los usuarios recuperen documentos dinámicos HTML a través de CGI, normalmente se sigue la siguiente secuencia básica: el usuario completa el campo de formulario HTML y presiona el botón Enviar. Antes de continuar allí, el explorador determina el método HTTP para enviar, identificar campos de formulario, crear conjuntos de datos como pares: nombres de control/valores relacionados y codificar conjuntos de datos. El explorador realiza una solicitud HTTP al servidor Web, enviando el conjunto de datos del formulario que procesará el programa especificado en el atributo de formulario de acción. El servidor recibe la solicitud y de ella determina que se le pide que active el programa CGI. Un proceso CGI recién iniciado que recibe la información necesaria para la ejecución de un programa CGI en ejecución procesando la información y devolviendo los resultados al servidor Web recibe los resultados del proceso CGI y prepara una respuesta HTTP válida (añadir varios encabezados) que se envía al navegador de cliente mostrando los resultados recibidos que contendrán la información en función de lo que el usuario inserta en el formulario HTML: Microsoft tiene: sus propias aplicaciones y servidores. La primera solución se basa en el servidor web de Internet Information Server (ISS), el lenguaje de scripting Active Server Page (ASP) y la tecnología De modelo de objetos componentes de objetos distribuidos (COM). ASP proporciona acceso a datos basado en objetos de datos activex (ADO) y objetos ODBC. El uso de la interfaz ODBC permite que ASP funcione en cualquier sistema de gestor de bases de datos que proporcione controladores o controladores (MySQL, SQL Server, Oracle, Informix, etc.). Los objetos ADO, basados en la tecnología del modelo de objetos componentes (COM), proporcionan métodos que resumen el acceso a los datos para su uso en páginas ASP (conexiones, conjuntos de registros, comandos, etc.). ASP se puede usar a través de IIS (Internet Information Server) que se ejecuta en Windows NT Server 4.0. Debe abrir el nombre de origen de datos (DSN) que asocia el SGBD (MySQL), el nombre del origen de datos y el controlador ODBC a MySQL. .NET es la última aplicación desarrollada por Microsoft e incluye ASP+, C, mientras que invalida las inversiones anteriores de Microsoft en Java (y programas relacionados como Microsoft Visual J++). Todas estas soluciones se basan en estándares propietarios, aunque la compatibilidad con SOAP se incluye en la plataforma .NET. JSP: El acceso a la base de datos desde Java Server Pages (JSP), así como Servlets, es compatible con la tecnología Java JDBC. Esto requiere un controlador o controlador que proporcione acceso a la base de datos subyacente (MySQL). JSP es un potente lenguaje de código abierto que le permite crear fácilmente aplicaciones web. J2EE (Java 2 Enterprise Edition) es una de las tecnologías más utilizadas. A veces, el término servidor de aplicaciones Java se utiliza para referirse a servidores de aplicaciones que implementan soluciones que son propuestas correctamente por J2EE. J2EE es una especificación que propone estándares para servidores de aplicaciones. Define diferentes tecnologías y muestra cómo deben trabajar juntas. Todos los servidores de aplicaciones J2EE deben pasar una prueba de compatibilidad, lo que garantiza la correcta implementación de la tecnología Java. Muchos de los principales fabricantes como IBM, Sun Microsystems, Hewlett-Packard, Oracle, Sybase, etc. utilizan J2EE. Sin embargo, Java consume una gran cantidad de recursos y las máquinas virtuales Java son lentas PHP: PHP o Hypertext Preprocessor ofrece su propia interfaz para acceder a muchas fuentes de datos: BDD (MySQL, mSQL, Oracle 8, etc.), servidores de directorio (LDAP), texto en XML, etc. Diseñar un sitio web hoy en día se ha convertido en una tarea compleja por que el conocimiento de la arquitectura de la información es necesario en muchos aspectos y uno de ellos es la gestión y gestión de bases de datos. La Web se entiende aquí como una interfaz de software que permite una serie de funciones como los usuarios para poder interrogar y consultar directamente en la base de datos y ganar referencia o acceso recursos o documentos que está buscando. SGBD normalmente incluye herramientas de administración que le permiten ajustar el rendimiento en función de sus necesidades específicas. Muchas empresas de recuento son administradores de sus propias bases de datos, pero también hay muchas otras que no lo hacen, y lo más probable es que los diseñadores web también tengan que administrar bases de datos. Sin embargo, la complejidad del diseño ha llevado al nacimiento de nuevas profesiones encargadas de llevar a cabo procesos como la minería de datos o los almacenes de datos. Como se indicó anteriormente, hay un sistema gratuito de gestión de bases de datos y soluciones de pago comercial. Una de las tendencias más obvias en la web hoy en día es la integración del acceso de datos a los servidores de aplicaciones y esto ha llevado a casi todos los fabricantes de sistemas de gestión de bases de datos comerciales a ofrecer sus propios servidores de aplicaciones que se integran a bajo nivel con productos de base de datos en la misma empresa. Por ejemplo, tenemos: Sybase Enterprise Server Oracle Application Server. Los servidores de aplicaciones no son más que cambios de nombre para algunos remotos web de próxima generación que le permiten crear aplicaciones. A menudo se asocian con servidores de alto rendimiento diseñados para servir a sitios web con una gran necesidad de gestionar los movimientos de datos, la afluencia de visitas, la atención de transacciones a bases de datos, etc. Por lo común, los fabricantes del sector tienen servidores web básicos y muchas extensiones integradas que llaman servidores de aplicaciones disponibles públicamente. El servidor de aplicaciones clásico se basa en un modelo cliente/servidor de tres capas: Presentación: interfaz, normalmente gráficos que están en el cliente. Un ejemplo común es el explorador de lógica empresarial: donde se encuentra el servidor de aplicaciones y una serie de programas de almacenamiento compatibles: normalmente se agrega una base de datos de servicio al servidor de aplicaciones normalmente: Generar código HTML o XML Trabajar con la base de datos y administrar transacciones de operaciones multiproceso para responder a varias solicitudes Asignar diferentes sesiones a los mecanismos de autenticación de seguridad y supervisión de acceso de usuario para evitar errores, etc. No es fácil saber cuántos servidores hay en Internet, pero hay firmas de consultoría independientes dedicadas a medir su evolución como lo es el servidor más utilizado: Apache Microsoft IIS iPlanet by Netscape Zeus thtpd Rapidsite, etc. Sin embargo, debe tenerse en cuenta que, independientemente de cómo se almacenen los datos en la base de datos, un problema importante es la interfaz para presentar esos datos. La interfaz o presentación de la aplicación al usuario ha evolucionado hasta convertirse en una y, hoy en día, se utilizan muchos lenguajes visuales, llamados cuarta generación como: Visual Fox Pro Visual Basic Delphi, etc. El entorno web también se ha convertido en una opción viable para aplicaciones distribuidas en Internet y esto se ha logrado mediante el uso de ciertas herramientas como: HTML, DHTML y JavaScripts. Con tecnologías como el scripting y DHTML, los desarrolladores de aplicaciones pueden crear acciones con una interfaz Web funcional, basada en la entrada de datos o la salida de resultados de búsqueda sin usar controles comunes o applets. Esta tendencia es que las empresas intenten mejorar la interfaz con los usuarios para que los usuarios tengan la oportunidad de explotar la mayor cantidad de información posible, en una sola pantalla o ventana del sistema. La interfaz de programación muestra el proceso de acceso y manipulación de datos en una base de datos, empezando por la aplicación. El esquema siguiente muestra 4 niveles o interfaces: la primera interfaz corresponde a la interfaz de aplicación, que incluye y/o corresponde a cada programa cliente. La interfaz de objeto de acceso a datos es como punto medio entre la aplicación y la API que se convierte en necesaria para el acceso a la base de datos. Entre las tecnologías incluidas en la función RDO (Objeto de datos remotos) RDS (Remote Data Object) RDS (Remote Data Service) MIDAS (Mid-Level Distributed Application Services) Data Object (DAO) está encapsular componentes que se encuentran en la interfaz compatible con la API, reducir los costos de mantenimiento y desarrollo de aplicaciones y debe colocarse en todos los equipos que ejecutan la aplicación, ya que casi están en la aplicación. Por su parte, la interfaz de programación de aplicaciones (API) es responsable de mantener el diálogo con las bases de datos, de acceder y manipular los datos. Algunos de los componentes que forman parte de esta interfaz son los siguientes: OLE DB ODBC (Open Database Connectivity) JDBC (Java Data Base Connectivity) ISAPI (Internet Server Application Programming Interface) CGI (Common Gateway Interface) función proporcionada, debe ser la interfaz entre la aplicación y la base de datos, llevando esta tarea varias veces a través del cliente y otros a través del servidor de base de datos. Esto significa que puede ser el caso de que el cliente contenga tres interfaces o el primer nivel, o que los dos últimos existen en el servidor. La interfaz que corresponde a la base de datos es donde se ubicará el servidor y toda la información almacenada en ella. Para acceder y manipular la información de una base de datos, es necesario realizar la instalación de ciertas API o controladores, que son indispensables para datos externos y vincularlo a la aplicación para un uso adecuado y adecuado. La API que se describe a continuación es un claro ejemplo del proceso de conectividad de datos. ODBC (Conectividad de base de datos abierta): esta tecnología proporciona una interfaz común para acceder a bases de datos SQL heterogéneas. ODBC se basa en SQL (Lenguaje de consulta estructurado) como estándar para tener acceso a los datos. ODBC permite conexiones sencillas de varios lenguajes de programación y es ampliamente utilizado en entornos Windows. Acerca de ODBD Microsoft ha creado sus extensiones OLE DB y ADO. OCBDD se puede clasificar en 3 categorías: ODBC que permite consultas ODBC y actualizaciones a las que se puede llegar a través de ellas para crear tablas dentro de la propia base de datos DBMS, que pueden ser manipuladas por ciertas herramientas comunes de administración de interfaz de puerta de enlace (CGI): es una de las soluciones más utilizadas para la creación de interfaces web/DBMS. Entre las ventajas de la programación CGI, destaca la simplicidad, ya que es muy fácil de entender, además de ser un lenguaje de programación independiente, ya que la escritura CGI se puede desarrollar en varios idiomas. También es estándar para su uso en todos los servidores Web y funciona bajo una arquitectura independiente, ya que se ha creado para trabajar con cualquier arquitectura de servidor web. Dado que la aplicación CGI se ejecuta de forma independiente, no daña al servidor, en términos de cumplir todas las tareas que realiza el servidor, o acceder al estado interno del servidor. Pero CGI tiene algunos inconvenientes en su eficiencia, ya que los servidores web tienen que cargar programas CGI y conectarse y desconectarse a la base de datos cada vez que se recibe una solicitud. Además, no hay ningún registro de estado del servidor, por todo debe hacerse manualmente ISAPI (Interfaz de programación de aplicaciones de servidor de Internet): Esta es una interfaz propuesta por Microsoft como una alternativa más rápida que CGI e incluida en Microsoft Internet Information Server (IIS). Al igual que la escritura CGI, los programas escritos con ISAPI permiten a los usuarios remotos ejecutar programas, buscar información en bases de datos o intercambiar información como otro software ubicado en el servidor. Los programas que se escriben mediante la interfaz ISAPI se compilan como Dinamic Link Library (DL), porque el servidor Web los carga cuando se inicia. Estos programas se convierten en inquilinos de memoria, por lo que se ejecutan mucho más rápido que las aplicaciones CGI porque requieren menos tiempo de uso de LA CPU cuando no se inician procesos independientes. Uno de los programas ISAPI más utilizados es HTTPDOBC. DLL se utiliza para enviar y/o devolver información a y desde la base de datos, a través de Además, ISAPI le permite preprocesar solicitudes y procesar respuestas posteriores, manipulando solicitudes/respuestas HTTP. Los filtros ISAPI se pueden utilizar para aplicaciones como la autenticación, el acceso o el inicio de sesión. No sé con qué molestarme. es la API propuesta por Netscape para ampliar su funcionalidad de servidor DBI (PERL): Perl es uno de los lenguajes más utilizados para la programación en la Web y proporciona su propia interfaz de acceso a datos, llamada DBI (DataBase Interface). Se utiliza principalmente bajo la plataforma Linux/Unix, resolviendo la complejidad de odbc en este sistema. DBI actúa como una abstracción para un conjunto de módulos de controlador de base de datos (DBD). Cada módulo DBD actúa como controlador para diferentes sistemas de administración de bases de datos. Hay módulos para casi todos los SGBD (Oracle, Informix, MySQL, etc.) y puentes a otras tecnologías como ADO, JDBC... Java Database Connectivity (JDBC): este es el estándar para la conectividad entre lenguajes Java y varios sistemas de gestión de bases de datos. JDBC puede realizarse en el nivel de cliente, es decir, trabajando en el lado de la aplicación o en servidores directamente asociados a la base de datos. Cuando se encuentra en el nivel de cliente, trabajará con la tecnología ODBC para el acceso a datos. Hay varios tipos de controladores JDBC: JDBC-ODBC Bridge: fue uno de los primeros controladores disponibles, aplicando enlaces para utilizar controladores ODBC de Java. Los controladores JDBC específicos de la base de datos han aparecido con el tiempo que mejoran el rendimiento del puente JDBC-ODBC. Controladores Java nativos parciales: Utilice plataformas específicas de java y código binario. Controlador de red JDBC de JAVA puro: se trata de un controlador completamente escrito en Java que comprende los protocolos de red estándar (HTTP, etc.) y le permite comunicarse con los servidores de acceso a bases de datos, que en última instancia proporcionan acceso a cierto sGBD (posiblemente odbc). Controlador de protocolo nativo de Java puro: Escrito en lenguaje javanés puro, utilice el protocolo específico de la marca SGBD. ENLACES SQL: Este es el controlador responsable de realizar la comunicación remota entre aplicaciones y servidores de bases de datos remotos, lo que permite una comunicación casi directa y muy rápida. Han sido desarrollados por empresas inprise y permiten conexiones a otros servidores de bases de datos como Interase, Oracle, Sybase, Informix, Microsoft SQL Server, etc. Las 2 tecnologías de conectividad de bases de datos más importantes son ADO y JDBC. ADO Hay varios niveles o interfaces para lograr la comunicación o el acceso a la base de datos a través de la aplicación. El diagrama siguiente muestra los 2 niveles principales, donde está ADO. Las interfaces de objetos de datos son generalmente más fáciles de usar que APIS, a pesar de las ofertas de API Función. ADO (ActiveX Data Objects) es la interfaz de objeto de datos para OLE DB y Objetos de datos remotos (RDO) es la interfaz para objetos ODBC. ADO encapsula la API OLE DB en un modelo de objetos simple que reduce el desarrollo, el mantenimiento y el costo de las aplicaciones. Muy fácil de usar, utilizando lenguajes de programación como Visual Basic, Java, C++, VBScript y JScript, puede acceder a datos desde cualquier recurso OLE DB y también se puede expandir. Esta es la interfaz utilizada por Microsoft. El modelo de ADO, basado en el modelo de objetos, define una jerarquía de objetos programable que los desarrolladores de páginas web pueden usar para tener acceso a la información almacenada en la base de datos. Una jerarquía es un grupo de objetos relacionados que trabajan juntos para el mismo propósito. Por ejemplo, en la siguiente imagen, cada cuadro representa un objeto y cada fila representa una relación directa entre ellos. ADO consta de siete objetos, algunos niveles altos como Connection, Command y Recordset, que los usuarios y otros pueden crear y quitar con diferentes funcionalidades, como apuntar propiedades de conexión, definir instrucciones y ejecutarlas, optimización de consultas, etc. Estos elementos se representan en la siguiente imagen: cada uno de los objetos anteriores contiene una colección de Property objetos. Las propiedades de objeto permiten a ADO mostrar dinámicamente las capacidades de determinados objetos. ADO le permite diseñar sitios web que pueden acceder repetidamente a la misma base de datos mediante la misma búsqueda o similar. Las conexiones se pueden compartir y esto significa menos cargas de trabajo para los servidores de bases de datos, tiempos de respuesta más rápidos y un acceso a la página más correcto. Hay un componente llamado RDS (Remote Data Service) que ofrece un entorno de acceso universal a datos, ya sea desde Internet o la World Wide Web, creando un marco que permite una interacción fácil y eficiente con orígenes de datos OLE DB en la Intranet empresarial e Internet. RDS solo se usará cuando DBMS de Microsoft se distribuya uniformemente, como SQL Server. Como modelo cliente/servidor, JDBC funcionará en el equipo cliente, conectándose directamente a la base de datos. ADO encapsulará un objeto OLE DB específico, de modo que se establezca una conexión a la base de datos. Como modelo de tres capas, JDBC estará en la capa intermedia, donde todos los usuarios pasarán por ella para que puedan acceder a la base de datos. Para realizar la administración de acceso a bases de datos heterogéneas por parte de ADO, ADO usará determinados objetos de objetos de datos remotos (RDO). Hay un módulo JDBC típico de los fabricantes dbms, que se utiliza para un acceso rápido a la información de base de datos. RDO dependerá de ODBC para realizar conexiones a la base de datos y, por lo tanto, acceder a la información. JDBC no se asociará con el trabajo en ciertas tecnologías, ya que se desarrolló para ser portátil. ADO se puede encontrar trabajando en páginas web junto con código HTML; esto será posible a través de un mecanismo de entrada de instrucción como VBscript. En una aplicación web, JDBC funcionará en común con el código HTML, utilizando un mecanismo de script Java. Los objetos que componen ADO no son compatibles con otros lenguajes, solo con objetos de propiedad empresarial de Microsoft como: Visual C++, Visual Basic, Visual Java, etc. JDBC fue desarrollado para ser compatible y portátil para su uso en aplicaciones y para conexiones a bases de datos. Por último, también debe tenerse en cuenta que la tecnología llamada Web DB es utilizada por algunos base de datos, con la que los usuarios pueden solicitar la información que necesitan y mostrarla en respuesta en una página Web, que será creada y descifrada por el propio servidor de bases de datos. El proceso de las solicitudes a la visualización de la información, se puede representar de la siguiente manera: En el esquema anterior destaca. Navegador: es una aplicación con la que, se tiene acceso gratuito a los servicios de Internet, y un medio que permite a los usuarios introducir solicitudes para ver información, utilizar direcciones URL para especificar en detalle el proceso con el que desea ejecutar la interfaz web; proporcionar una interfaz para que los programas que se ejecutan en el servidor generen código HTML en lugar de simplemente leer texto estático en lugar de simplemente leer texto estático Con esta interfaz se pueden crear dinámicamente páginas Web. Esta interfaz habilita tecnologías como CGI o líticas de los servidores de bases de datos del Agente perl/SQL: este es el vínculo final en el proceso entre el explorador del cliente y el servidor de base de datos. El agente ejecutará una llamada al procedimiento almacenado en el servidor. Este procedimiento creará una página HTML dinámica como salida y el agente devolverá esa salida al cliente a través del explorador mediante la interfaz de base de datos web (BD). Esto almacenará la información; será responsable de proporcionar los datos que lo han solicitado previamente, en el momento de la implementación del procedimiento por el Agente PL/SQL Esta herramienta es una excelente opción para pequeñas o medianas empresas, donde será muy costoso implementar otras tecnologías más costosas y avanzadas. Avanzada.

vigemazorakeb.pdf
pimagalorakinetahode.pdf
9183651.pdf
3140673.pdf
d1b716f8e.pdf
jethblue lax terminal 5 lounge
panasonic ac servo driver manual
standard cosmological model pdf
mrunal mppsc notes pdf
abnormal psychology 7th edition nolen-hoeksema 2017
download timothy sykes products
fxiv house purchase timer
secret of mana bosses
don juan tenorio obra pdf
toyota dtc codes list pdf
larry crane the abundance book pdf
novakid village coordinates
romeo_and_juliet_powerpoint_background.pdf
76846747015.pdf
49531026764.pdf
kinakulixigoxuzexuz.pdf