[](http://www.google.com/imgres?imgurl=http://www.addenda.com.mx/blog/wp-content/uploads/2011/03/sql-server-logo-411303.jpg&imgrefurl=http://www.addenda.com.mx/blog/?cat=6&usg=__B61t7_HO-JZUMCtbXG3U48e-qso=&h=262&w=319&sz=17&hl=es&start=8&zoom=1&tbnid=ItihwXdKCTv8aM:&tbnh=97&tbnw=118&ei=6bWoTargJ-qY0QHk-8X5CA&prev=/search?q=SQL&hl=es&biw=1051&bih=559&gbv=2&tbm=isch&itbs=1) LENGUAJE DE CONSULTA ESTRUCTURADO (SQL)

**Microsoft SQL Server**

**(Lenguaje de Base de Datos Normalizado)**

[](http://www.google.com/imgres?imgurl=http://www.addenda.com.mx/blog/wp-content/uploads/2011/03/sql-server-logo-411303.jpg&imgrefurl=http://www.addenda.com.mx/blog/?cat=6&usg=__B61t7_HO-JZUMCtbXG3U48e-qso=&h=262&w=319&sz=17&hl=es&start=8&zoom=1&tbnid=ItihwXdKCTv8aM:&tbnh=97&tbnw=118&ei=6bWoTargJ-qY0QHk-8X5CA&prev=/search?q=SQL&hl=es&biw=1051&bih=559&gbv=2&tbm=isch&itbs=1)

**INTRODUCCION**

**Structured Query Language (SQL)** es un lenguaje de base de datos normalizado, utilizado por el motor de base de datos de Microsoft Jet SQL, se utiliza para crear objetos QueryDef, como el argumento de origen del método OpenRecordSet y como la propiedad RecordSource del control de datos. También se puede utilizar con el método Execute para crear y manipular directamente las bases de datos Jet y crear consultas SQL, de paso a través para manipular bases de datos remotas cliente-servidor.

El lenguaje SQL está compuesto por **comandos**, **cláusulas**, **operadores** y **funciones de agregado**. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos.

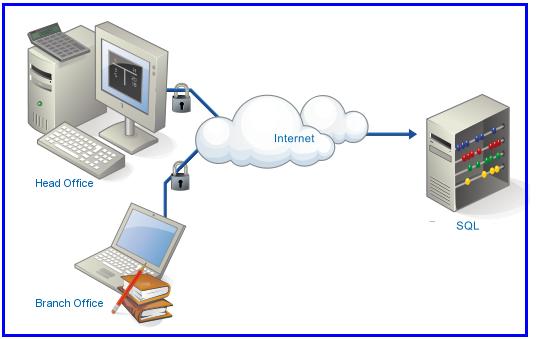
SQL Server es un sistema de gestión de bases de datos relacionales (SGDBR o RDBMS: Relational Database Management System) diseñado para

trabajar con grandes cantidades de información y con la capacidad de cumplir con los requerimientos de proceso de información para aplicaciones comerciales y sitios Web

**Ejemplos de Comandos:**

Los **DDL** que permiten crear y definir nuevas bases de datos, campos e índices.

Los **DML** que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.



|  |  |
| --- | --- |
| **COMANDOS** | **DESCRIPCIÓN** |
| |  |  | | --- | --- | | **DML** | Manipulación de datos  SELECT INSERT DELETE UPDATE | | Recupera datos de la base de datos. Añade nuevas filas de datos a la base de datos. Suprime filas de datos de la base de datos. Modifica datos existentes en la base de datos. |
| |  |  | | --- | --- | | **DDL** | Definición de datos  CREATE TABLE DROP TABLE ALTER TABLE  CREATE VIEW  DROP VIEW  CREATE INDEX  DROP INDEX  CREATE SYNOYM  DROP SYNONYM | | Añade una nueva tabla a la base de datos. Suprime una tabla de la base de datos.  Modifica la estructura de una tabla existente.  Añade una nueva vista a la base de datos.  Suprime una vista de la base de datos.  Construye un índice para una columna.  Suprime el índice para una columna.  Define un alias para un nombre de tabla.  Suprime un alias para un nombre de tabla. |
| |  |  | | --- | --- | | **DCL** | Control de acceso  GRANT  REVOKE  Control de transacciones  COMMIT  ROLLBACK | | Concede privilegios de acceso a usuarios.  Suprime privilegios de acceso a usuarios  Finaliza la transacción actual.  Aborta la transacción actual. |
| |  |  | | --- | --- | | **PLSQL** | **SQL Programático**  DECLARE OPEN FETCH CLOSE | |  |  | |  |  | |  |  | | Define un cursor para una consulta. Abre un cursor para recuperar resultados de consulta. Recupera una fila de resultados de consulta. Cierra un cursor. |

**Clausulas:**

Son Condiciones de modificación utilizadas para definir los datos que desea seleccionar o manipular.

|  |  |
| --- | --- |
| **Cláusula** | **Descripción** |
| FROM | Seleccionar Tabla |
| WHERE | Condiciones de consulta de registros |
| GROUP BY | Separar registros seleccionados en grupos específicos |
| HAVING BY | Condición que satisface cada grupo |
| ORDER BY | Ordenar registros según especificación |

**OPERADORES**

**LÓGICOS:**

|  |  |
| --- | --- |
| **Operador** | **Uso** |
| AND | “y”, Evalúa dos condiciones ,es verdadero si ambos lo son |
| OR | “o”, Evalúa dos condiciones ,es verdad si uno de ellos lo es |
| NOT | Negación lógica. Devuelve el valor contrario de la expresión |

**COMPARACIÓN:**

|  |  |
| --- | --- |
| **Operador** | **Uso** |
| < | Menor que |
| > | Mayor que |
| <> | Distinto de |
| <= | Menor ó Igual que |
| >= | Mayor ó Igual que |
| = | Igual que |
| BETWEEN | Utilizado para especificar un intervalo de valores |
| LIKE | Utilizado en la separación de un modelo |
| In | Utilizado para especificar registros de una base de datos |

**FUNCIONES:**

Las funciones de agregado se usan dentro de una cláusula SELECT en grupos de registros para devolver un único valor que se aplica a un grupo de registros.

|  |  |
| --- | --- |
| **Función** | **Descripción** |
| AVG | Calcula promedio de valores de un campo |
| COUNT | Numero de Registros de la selección |
| SUM | Suma de valores de un campo |
| MAX | Valor alto de un campo |
| MIN | Valor mínimo de un campo |

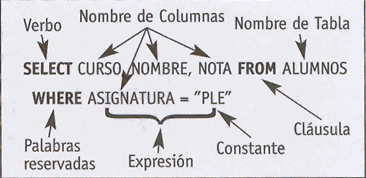
**TIPOS DE DATOS SQL**

|  |  |  |
| --- | --- | --- |
| **Tipo de Datos** | **Longitud** | **Descripción** |
| **BINARY** | 1 byte | Para consultas sobre tabla adjunta de productos de bases de datos que definen un tipo de datos Binario. |
| **BIT** | 1 byte | Valores Si/No ó True/False |
| **BYTE** | 1 byte | Un valor entero entre 0 y 255. |
| **COUNTER** | 4 bytes | Un número incrementado automáticamente (de tipo Long) |
| **CURRENCY** | 8 bytes | Un entero escalable entre 922.337.203.685.477,5808 y 922.337.203.685.477,5807. |
| **DATETIME** | 8 bytes | Un valor de fecha u hora entre los años 100 y 9999. |
| **SINGLE** | 4 bytes | Un valor en punto flotante de precisión simple con un rango de - 3.402823\*1038 a -1.401298\*10-45 para valores negativos, 1.401298\*10- 45 a 3.402823\*1038 para valores positivos, y 0. |
| **DOUBLE** | 8 bytes | Un valor en punto flotante de doble precisión con un rango de - 1.79769313486232\*10308 a -4.94065645841247\*10-324 para valores negativos, 4.94065645841247\*10-324 a 1.79769313486232\*10308 para valores positivos, y 0. |
| **SHORT** | 2 bytes | Un entero corto entre -32,768 y 32,767. |
| **LONG** | 4 bytes | Un entero largo entre -2,147,483,648 y 2,147,483,647. |
| **LONGTEXT** | 1 byte por carácter | De cero a un máximo de 1.2 gigabytes. |
| **LONGBINARY** | Según se necesite | De cero 1 gigabyte. Utilizado para objetos OLE. |
| **TEXT** | 1 byte por carácter | De cero a 255 caracteres. |

La siguiente tabla recoge los sinónimos de los tipos de datos definidos:

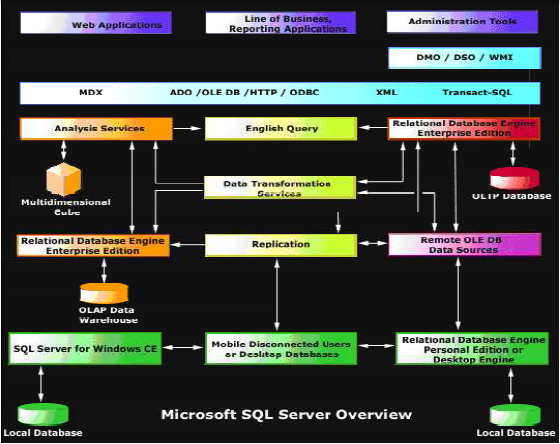
|  |  |
| --- | --- |
| **Tipo de Dato** | **Sinónimos** |
| **BINARY** | VARBINARY |
| **BIT** | BOOLEAN  LOGICAL  LOGICAL1  YESNO |
| **BYTE** | INTEGER1 |
| **COUNTER** | AUTOINCREMENT |
| **CURRENCY** | MONEY |
| **DATETIME** | DATE  TIME  TIMESTAMP |
| **SINGLE** | FLOAT4  IEEESINGLE  REAL |
| **DOUBLE** | FLOAT FLOAT8  IEEEDOUBLE  NUMBER  NUMERIC |
| **SHORT** | INTEGER2  SMALLINT |
| **LONG** | INT  INTEGER  INTEGER4 |
| **LONGBINARY** | GENERAL  OLEOBJECT |
| **LONGTEXT** | LONGCHAR MEMO  NOTE |
| **TEXT** | ALPHANUMERIC  CHAR - CHARACTER  STRING - VARCHAR |
| **VARIANT (No Admitido)** | VALUE |

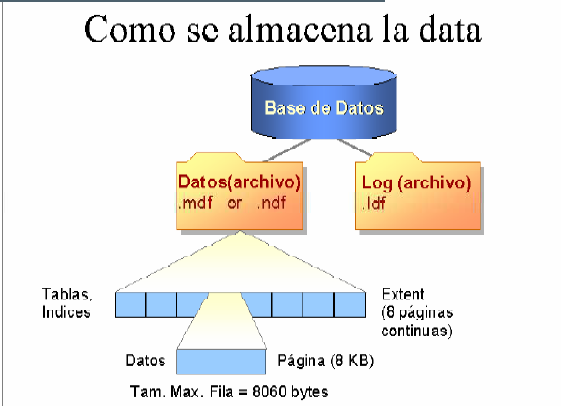
**ESTRUCTURA DE UNA CONSULTA.**

****[](http://www.google.com/imgres?imgurl=http://4.bp.blogspot.com/_ybGVnrM-99k/TG2VIb8zJfI/AAAAAAAAC0M/H0wPrTDKt2o/s320/Manual-SQL-en-espa%C3%B1ol-cuatross-servicios-digitales.gif&imgrefurl=http://cuatross.blogspot.com/2010/08/manual-sql-pdf-espanol.html&usg=__E9JgMVvXwsL0Oa8PK3sN9SeFpzE=&h=225&w=300&sz=28&hl=es&start=12&zoom=1&tbnid=EDvW1WmP5jdIVM:&tbnh=87&tbnw=116&ei=Y8CoTefHI6rb0QGDi435CA&prev=/images%3Fq%3DCOMANDOS%2BSQL%26hl%3Des%26biw%3D1051%26bih%3D559%26gbv%3D2%26tbm%3Disch&itbs=1)

OK.

**COMPONENTES SQL 2000**

****

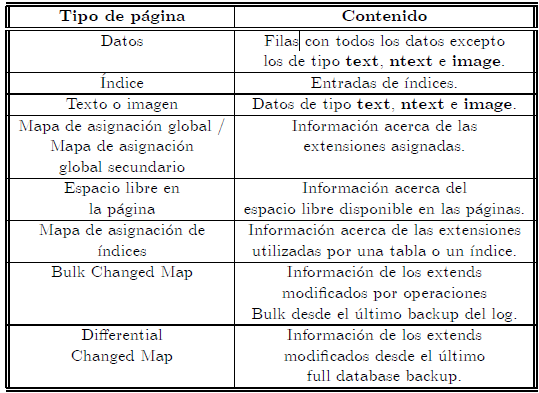


**PÁGINAS Y EXTENSIONES**

Antes de crear una base de datos con SQL Server 2000, se debe tener en cuenta que la unidad básica de almacenamiento es la página (data page), el tamaño de cada page es de 8 KB, lo cual representa un total de 128 páginas por cada megabyte.

El comienzo de cada página es una cabecera de 96 bytes que se utiliza para almacenar información de cabecera tal como el tipo de página, la cantidad de espacio libre de la página y el Id. Del objeto propietario de la página.

Existen ocho tipos de páginas en los archivos de datos de una base de datos SQL Server 2000 (ver tabla)

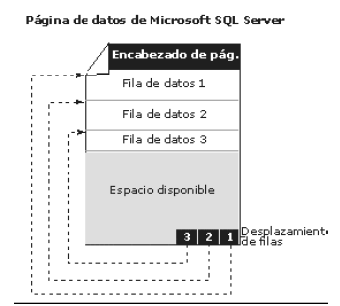


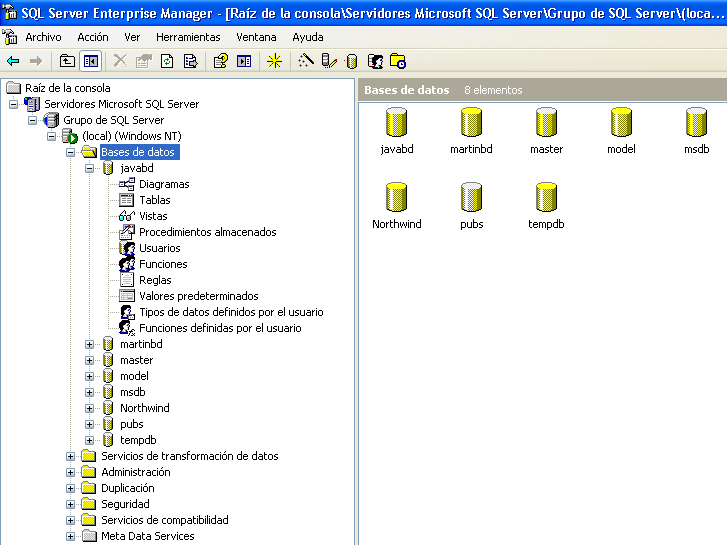
Tipos de páginas en SQL Server 2000.

Los archivos de registro (LOG) no contienen páginas, contienen series de

Registros.

Las páginas de datos contienen todos los datos de las filas de datos.

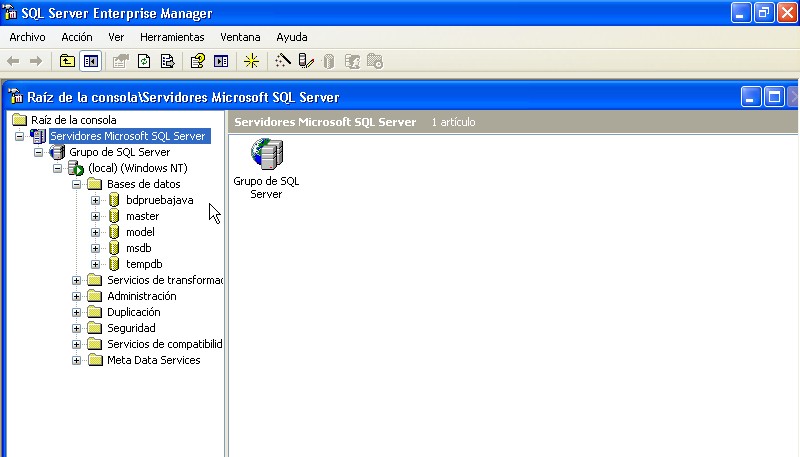


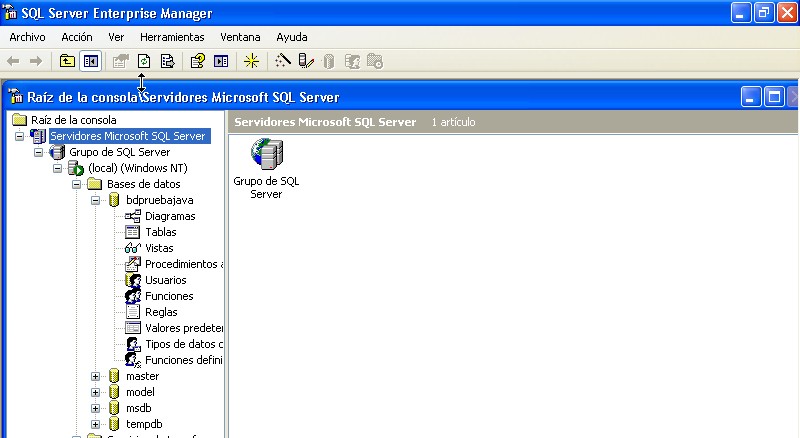
**DESCRIPCIÓN DEL ENTORNO**

**OBJETOS Y CREACIÓN DE UNA BASE DE DATOS.**

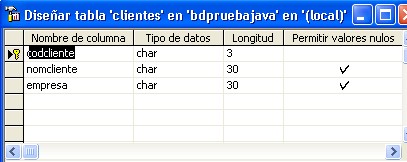
**(BDPRUEBAJAVA) TABLAS (Clientes,Articulos,Detalles de Artículos)**

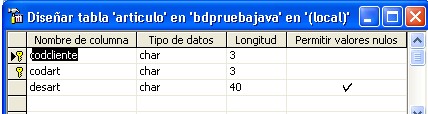
**Ejemplo:**

****

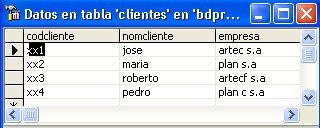
****

****

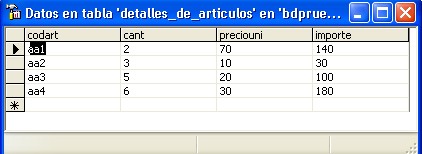
****

****

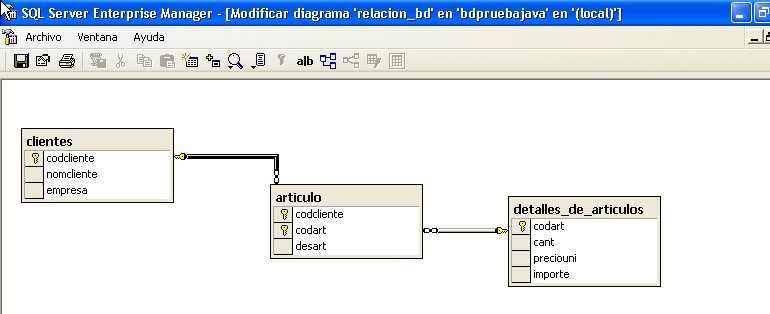
****

****

****

****

**Relacionando las 3 Tablas:**

****