

Material de Algebra y Sql

- SQL (Comandos)

SELECT

Recupera datos de una o más tablas.

Sintaxis

```
SELECT [ALL | DISTINCT]
[Alias.] Elemento_Selección [AS Nombre_Columna]
[, [Alias.] Elemento_Selección [AS Nombre_Columna] ...]
FROM [FORCE]
[NombreBaseDatos!]Tabla [[AS] Local_Alias]
[[INNER | LEFT [OUTER] | RIGHT [OUTER] | FULL [OUTER] JOIN
NombreBaseDatos!]Tabla [[AS] Local_Alias]
[ON CondiciónCombinación ...]
[[INTO Destino]
| [TO FILE NombreArchivo [ADDITIVE] | TO PRINTER [PROMPT]
| TO SCREEN]]
[PREFERENCE NombrePreferencia]
[NOCONSOLE]
[PLAIN]
[NOWAIT]
[WHERE CondiciónCombinación [AND CondiciónCombinación ...]
[AND | OR CondiciónFiltro [AND | OR CondiciónFiltro ...]]]
[GROUP BY ColumnaGrupo [, ColumnaGrupo ...]]
[HAVING CondiciónFiltro]
[UNION [ALL] SELECTCommand]
[ORDER BY Elemento_Orden [ASC | DESC] [, Elemento_Orden [ASC | DESC]
...]]
```

Argumentos

SELECT

Especifica los campos, constantes y expresiones que se mostrarán en el resultado de la consulta.

ALL

De forma predeterminada, se muestran todas las filas del resultado de la consulta.

DISTINCT

Excluye duplicados de cualquier fila del resultado de la consulta.

Nota Puede utilizar DISTINCT únicamente una vez por cláusula SELECT. estén permitidos para nombres de campos de tablas.

FROM

Enumera las tablas que contienen los datos que ha obtenido la consulta.

FORCE

Especifica que las tablas se combinarán en el orden de aparición en la cláusula FROM.

INNER JOIN

Especifica que el resultado de la consulta contenga sólo filas para una tabla con la que coincidan una o varias filas en otra tabla.

LEFT [OUTER] JOIN

Especifica que el resultado de la consulta contenga todas las filas de la tabla a la izquierda de la palabra clave JOIN y sólo las filas que concuerden procedentes de la tabla a la derecha de la palabra clave JOIN. La palabra clave OUTER es opcional; se puede incluir para resaltar que se ha creado una combinación externa.

RIGHT [OUTER] JOIN

Especifica que el resultado de la consulta contenga todas las filas desde la tabla hasta la derecha de la palabra clave JOIN y sólo las filas que concuerden desde la tabla hasta la izquierda de la palabra clave JOIN. La palabra clave OUTER es opcional; puede incluirse para resaltar la creación de una combinación externa.

FULL [OUTER] JOIN

Especifica que el resultado de la consulta contenga todas las filas, concuerden o no, de ambas tablas. La palabra clave OUTER es opcional; se puede incluir para resaltar que se ha creado una combinación externa.

ON *CondiciónCombinación*

Especifica las condiciones según las cuales se combinan las tablas.

ORDER BY *Elemento_Orden*

Ordena el resultado de la consulta basándose en los datos de una o varias columnas.

Cada *Elemento_Orden* debe corresponder a una columna del resultado de la consulta, y puede ser uno de los siguientes:

- Un campo de una tabla FROM que también es un elemento de selección en la cláusula principal SELECT (no en una subconsulta).
- Una expresión numérica que indica la ubicación de la columna en la tabla resultante. (La columna de la izquierda es la número 1.)
ASC especifica un orden ascendente para los resultados de la consulta, de acuerdo con el elemento o los elementos de orden, y es el valor predeterminado para ORDER BY.
DESC especifica un orden descendente para los resultados de la consulta.
Los resultados de la consulta aparecerán desordenados si no especifica un orden con ORDER BY.

Otras operaciones

- $AVG(Elemento_Selección)$, realiza una media de una columna de datos numéricos.
- $COUNT(Elemento_Selección)$, cuenta el número de elementos seleccionados en una columna.
- $COUNT(*)$ cuenta el número de filas en el resultado de la consulta.
- $MIN(Elemento_Selección)$ determina el menor valor de $Elemento_Selección$ en una columna.
- $MAX(Elemento_Selección)$ determina el mayor valor de $Elemento_Selección$ en una columna.
- $SUM(Elemento_Selección)$ que proporciona el total de la suma de una columna de datos numéricos.

No se pueden probar las funciones de campo.

Ejemplos de SELECT - SQL (Comando)

Los siguientes ejemplos ilustran la utilización de las funciones definidas por el usuario con SELECT - SQL:

Ejemplo 1

En el Ejemplo 1 se muestran los nombres de todas las Empresas (un campo de una tabla).

```
SELECT Empresas.nombre_emp ;  
FROM Empresa
```

Ejemplo 2

El ejemplo 2 muestra el contenido de tres campos de dos tablas y combina las dos tablas basándose en el campo emp_id. Utiliza alias locales para ambas tablas.

```
SELECT Empresa.nombre_emp,  
funcionario.nombre_func,Funcionario.Fecha_nac;  
FROM empresa,funcionario ;  
WHERE empresa.Emp_ID=funcionario.Emp_ID
```

Ejemplo 3

El ejemplo 3 muestra únicamente registros con datos únicos en los campos especificados.

```
SELECT DISTINCT Empresa.nombre_emp, funcionarios.nombre_func  
FROM empresa,funcionario ;  
WHERE empresa.Emp_ID=funcionario.Emp_ID
```

Ejemplo 4

El ejemplo 4 muestra los campos País, postalcode y nombre_emp en orden ascendente.

```
SELECT País, postalcod, nombre_emp ;  
FROM Empresa ;  
ORDER BY País, postalcod, nombre_emp
```

Ejemplo 5

El ejemplo 6 muestra únicamente los registros con una fecha nacimiento anterior a 02/16/1964.

```
SELECT Empresa.nombre_emp,  
funcionarios.nombre_func,Funcionario.Fecha_nac;  
FROM empresa,funcionario ;  
WHERE empresa.Emp_ID=funcionario.Emp_ID  
And funcionario.fecha_nac< 02/16/1964
```

Ejemplo 6

El ejemplo 6 muestra los nombres de todas las empresas de empresa con un código postal que coincida con el código postal de la tabla zonas.

```
SELECT nombre_emp FROM empresa a WHERE ;  
      EXISTS (SELECT * FROM zonas b WHERE a.postalcode = b.postalcode)
```

Ejemplo 7

El ejemplo 7 muestra todos los registros de Empresa que tengan un nombre de empresa que comience por una C mayúscula y tenga cualquier longitud.

```
SELECT * FROM Empresa a WHERE a.nombre_emp LIKE "C%"
```

Ejemplo 8

El ejemplo 8 muestra todos los registros de Empresa que tengan un nombre de país que comience por U mayúscula seguido de un carácter desconocido.

```
SELECT * FROM empresa a WHERE a.pais LIKE "U_"
```

Ejemplo 9

El ejemplo 9 muestra los nombres de todas las ciudades de empresa en mayúsculas y llama CityList a la columna de los resultados.

```
SELECT UPPER(city) AS CityList FROM empresa
```

Otras Consultas.

Dado el siguiente Esquema relacional.

```
Peliculas(#pel,titulo,Año,#prod)
Productora(#prod,nombre)
Actores(#Actor,nombre,Fec-nac)
Premios(#pre,descripción)
Obtuvo(#pel,#pre,año)
Pel-act(#Pel,#Actor)
Programación(#pel,fech,hora,cant-ent)
```

Consultas resueltas

1) dada la siguiente consulta sql representarla en algebra relacional.

```
select #pel from pelicula
where #prod in (select #prod from productora)
```

Su representación en algebra relacional.

$$\pi_{\text{#pel}}(\text{peliculas})$$

2) listar nombre de las peliculas y detalle de actores que actuaron en cada una de ellas ordenado por pelicula en forma descendente.

```
Select pelicula.titulo, actores.nombre
from pelicula,Actores,Pel-Act
Where pelicula.#pel=pel-Act.#pel and
      actores.#actor=pel-act.#actor
group by 1
order by 1 desc
```

3)listar titulo de peliculas y cantidad de entradas para las peliculas posteriores al año 1995 y que vendieron mas de 2000 entradas en todas sus presentaciones.

```
Select pelicula.titulo, sum (cant-ent)
From  pelicula,programación
Where  pelicula.#pel= programación.#pel and
pelicula.año>1995
Group by 1
Having sum(cant-ent)> 2000
```

4)Listar los nombres de las productoras de peliculas que no obtuvieron premios en el año 1994, ordenados en forma ascendente.

```
select productoras.nombre
from productoras,peliculas
where productoras.#prod=peliculas.#prod and
peliculas.#pel not in (select #pel from obtuvo
                        where año <> 1994)
order by 1
```

Dado el siguiente esquema:

```
Biblioteca(Cod_Bib,Dirección,Teléfono,Ciudad)
Libro(Cod_Libro,Editorial,NroEdición,Idioma)
Autor(Nom_autor,Nacionalidad)
Escribe(cod_libro,Nom_autor)
Ejemplar(Cod_ej,Cod_Libro,Cod_Bib)
Préstamo(Cod_ej,Fecha_Prest,Fecha_Dev)
```


1) Seleccionar los libros que solamente están disponibles en una biblioteca.

```
Select Cod_Llibro
From Ejemplar
Group by Cod_libro
Having count(Distinct cod_bib)=1
```

Otra forma de resolver

```
Select cod_libro
From ejemplar a
Where not exist (select *
                  From ejemplar b
                  Where a.cod_libro = b.cod_libro
                  And a.cod_bib <> b.cod_bib)
```

2) seleccionar todos los datos de las bibliotecas que tengan la mayor cantidad de ejemplares.

```
Select * From Biblioteca
Where Cod_Bib in (select cod_bib from Ejemplar
                  group by cod_bib
                  having count(*) >=all
                  (Select count(*) from ejemplar
                   Group by cod_bib))
```

3) seleccionar los libros que fueron escritos por mas de 3 autores

```
select cod_libro from Escribe
group by cod_libro
having count(*) > 3
```

4) seleccionar los libros que nunca fueron solicitados en préstamo.

```
Select cod_libro, count(*) from Ejemplar E, prestamo P
Where E.cod_ej = P.cod_ej
Group by Cod_Libro
```

5) Seleccionar los libros que nunca fueron solicitados en préstamo.

```
Select Cod_libro from Libro L
Where not exist (select * from Ejemplar E, Prestamo P
                  Where E.cod_ej=P.cod_ej
                  And l.cod_libro=E.cod_libro)
```

Algebra

Obtener datos completos de peliculas que nunca lograron vender mas de 100 entradas

$$A = \pi_{\#pel} \sigma_{\text{cant-ent} > 100} \text{Programacion}$$

$$B = \pi_{\#pel} \text{peliculas} - A$$

$$B \times \text{peliculas}$$

HAVING y WHERE en la misma consulta

En algunos casos, se puede querer excluir filas individuales de grupos (mediante una cláusula WHERE) antes de aplicar una condición a grupos en conjunto (mediante una cláusula HAVING).

Una cláusula HAVING es como una cláusula WHERE, pero sólo es aplicable a los grupos en conjunto (es decir, a las filas en el [conjunto de resultados](#) que representan a grupos), mientras que la cláusula WHERE es aplicable a las filas individuales. Una consulta puede contener tanto una cláusula WHERE como una cláusula HAVING. En ese caso:

- La cláusula WHERE es aplicable primero a las filas individuales en los [orígenes de entrada](#). Sólo se agrupan las filas que cumplen las condiciones en la cláusula WHERE.
- La cláusula HAVING entonces se aplica a las filas en el conjunto de resultados que se producen al agrupar. Sólo los grupos que cumplen las condiciones HAVING aparecen en el resultado de la consulta. Puede aplicar una cláusula HAVING sólo a las columnas que aparecen también en la cláusula GROUP BY o en una [función agregada](#).

Por ejemplo, si unimos las tablas `títulos` y `editores` para crear una consulta que muestre el precio medio de un libro para una serie de editores. Se pretende ver el precio medio para una serie específica de editores, quizás sólo los editores en Brasil. E incluso entonces, se quiere ver el precio medio sólo si está por encima de 10.00\$.

Se puede establecer la primera condición si incluye una cláusula WHERE, que descarte a los editores que no están en Brasil, antes de calcular los precios medios. La segunda condición requiere una cláusula HAVING, porque la condición se basa en los resultados del agrupamiento y resumen de los datos. La instrucción SQL resultante podría ser:

```
SELECT títulos.id_editor, AVG(títulos.precio)
FROM títulos INNER JOIN editores
  ON títulos.id_editor = editores.id_editor
WHERE editores.pais = 'Brasil'
GROUP BY títulos.id_editor
HAVING AVG(precio) > 10
```

Puede crear una cláusula WHERE y una cláusula HAVING que involucren a la misma columna. Para hacerlo, debe agregar la columna dos veces al panel **Cuadrícula**, después especificar un ejemplo como parte de la cláusula HAVING y el otro ejemplo como parte de la cláusula WHERE.

conjunto de resultados

Una tabla virtual que se produce por una consulta, representando todas las columnas y filas devueltas por la consulta en el orden especificado.

consulta

Una solicitud especial o un conjunto de instrucciones para recuperar, modificar, insertar o eliminar datos en una base de datos.

consulta agregada

Una consulta (instrucción SQL) que totaliza información de múltiples filas incluyendo una función agregada como Sum o Avg. Por ejemplo, puede crear una consulta que calcule el promedio del contenido de una columna `precio`. Las consultas agregadas también pueden mostrar información de subtotal creando grupos de filas que tienen datos comunes. Un ejemplo podría ser una consulta que muestra el precio medio de un libro por cada editor.

consulta Update

Una consulta (instrucción SQL) que cambia los valores en las columnas de una o más filas de una tabla. Por ejemplo, la siguiente consulta de actualización para la tabla `títulos` aumenta en un 10% el precio de todos los libros de un editor particular:

```
UPDATE títulos
SET precio = precio * 1.1
WHERE id_pub = '1389'
```

consulta de creación de tabla

Una consulta (instrucción SQL) que crea una tabla nueva y después crea filas en ella copiando filas de la tabla existente. Por ejemplo, la siguiente consulta `Crear tabla copia` copia filas de la tabla `títulos` a la tabla `títuloscocina`:

```
SELECT id_título, título, tipo, id_pub, precio
INTO títuloscocina
FROM títulos
WHERE tipo = 'cocina'
```

consulta de definición de datos

Una consulta específica de SQL que contiene instrucciones del Lenguaje de definición de datos (DDL). Estas instrucciones le permiten crear o modificar objetos (tablas, índices, vista, etc.) en la base de datos.

consulta de eliminación

Una consulta (instrucción SQL) que quita filas de una o más tablas. Por ejemplo, la siguiente consulta de eliminación quita filas de la tabla `títulos` si la columna `id_pub` contiene el valor "0766":

```
DELETE FROM títulos
WHERE id_pub = '0766'
```

consulta de inserción

Una consulta que copia columnas y filas específicas de una tabla a otra o a la misma tabla. Por ejemplo, la siguiente consulta de inserción copia el contenido de las columnas `id_título`, `título`, `tipo` e `id_pub` de la tabla `títulos` a la tabla `archivartítulos`:

```
INSERT INTO archivartítulos
SELECT id_título, título, tipo, id_pub
FROM títulos
```

consulta de inserción de valores

Una consulta (instrucción SQL) que crea una fila nueva e inserta valores en columnas especificadas. Por ejemplo, la siguiente consulta de inserción de valores inserta una fila nueva en la tabla `títulos`:

```
INSERT INTO títulos
id_título, título, tipo, id_pub, precio
VALUES ('BU0219', 'Web Page Design', 'business', '1389', 29.99)
```

consulta de paso a través

Una consulta que puede utilizar para trabajar directamente con tablas en un servidor que entiende un dialecto de SQL que es distinto del SQL procesado por la aplicación cliente. Puede utilizar una consulta de paso a través para devolver un snapshot de filas o ejecutar comandos que no devuelven filas.

consulta de selección

Una consulta que devuelve filas en un conjunto de resultados de una o más tablas. Una consulta de selección puede contener especificaciones para las columnas a devolver, las filas a seleccionar, el orden en el que colocar las filas y cómo agrupar (totalizar) la información.

Comparar con **consulta de eliminación**, **consulta de inserción** y **consulta de actualización**, todas las que cambian datos en una tabla.

consulta de referencias cruzadas

Muestra datos de valores totalizados de un campo de una tabla y después los agrupa en dos conjuntos; uno hacia abajo a la izquierda y el otro a lo largo de la parte superior de la hoja de datos.

consulta de unión

Una consulta que combina dos tablas realizando el equivalente de anexar una tabla a la otra.

DBCS (conjunto de caracteres de doble byte)

Un conjunto de caracteres que utiliza uno o dos bytes para representar un carácter, permitiendo representar más de 256 caracteres. Los conjuntos de caracteres DBCS se utilizan normalmente en entornos que utilizan sistemas de escritura ideográfica, como el japonés, coreano y chino.

desencadenante

Un tipo especial de un procedimiento almacenado que se exporta automáticamente cuando se modifican los datos en una tabla especificada. Los desencadenantes se crean normalmente para forzar la **integridad referencial** o la consistencia entre datos relacionados lógicamente en diferentes tablas.

desencadenante cifrado

Un desencadenante que se crea con un parámetro de cifrado opcional que cifra el texto de definición y que no se puede descifrar. El cifrado hace indescifrable la información para protegerla de un uso o vista no autorizados.

Comparar con **desencadenante**.

diagrama de base de datos

Una representación gráfica de cualquier parte de un **esquema** de base de datos. Un diagrama de base de datos puede ser una imagen completa o parcial de la estructura de la base de datos; incluye objetos para tablas, las columnas que contienen y las relaciones entre ellos.

esquema

Una descripción de una base de datos al sistema de administración de base de datos (DBMS), que se genera utilizando el lenguaje de definición de datos proporcionado por el DBMS. Un esquema define los atributos de la base de datos, como las tablas, columnas, propiedades.

Comparar con **diagrama de base de datos**.

equipo DSN

Almacena la información de cambios para una base de datos en el registro del sistema. La información de conexión consta de parámetros y sus valores correspondientes que el Administrador del controlador de ODBC utiliza para establecer una conexión.

Comparar con el **nombre del origen de datos**.

Comparar con **archivo DSN**.

expresión

Una combinación de operadores, constantes, valores literales, funciones, nombres de columnas, controles y propiedades que dan como resultado un valor único. En una consulta, puede utilizar expresiones en cualquier lugar donde pueda utilizar un nombre de columna. En el Diseñador de consultas, puede utilizar expresiones como valores para muchas acciones y argumentos de propiedades, para establecer criterios o definir campos calculados en consultas y para establecer condiciones en macros. En el Diseñador de base de datos, puede utilizar expresiones para definir restricciones de comprobación.

factor de relleno

Una opción que especifica cómo se debe realizar cada página de índices al rellenar la base de datos. La cantidad de espacio vacío en una página de índices es importante debido a que cuando se completa una página de índices, el sistema debe tener tiempo para dividirla para hacer espacio para nuevas filas. Utilice un factor de relleno de cero (0) para aceptar el valor predeterminado de base de datos.

fila

En una tabla, un conjunto de columnas de información relacionadas que se tratan como una unidad y que describen una entidad específica. Una fila es el equivalente lógico de un **registro**.

filtro

Un conjunto de criterios aplicados a registros para mostrar un subconjunto de registros o para ordenar los registros.

función agregada

Una función que genera un valor único de un grupo de valores, frecuentemente utilizado con cláusulas Grupo By y Having. Las funciones agregadas incluyen Avg, Count, Max, Min y Sum. También se conoce como una **función de conjunto**.

función de conjunto

Una función que genera un valor único de un grupo de valores, frecuentemente utilizado con cláusulas Grupo By y Having. Las funciones agregadas incluyen Avg, Count, Max, Min y Sum. También se conoce como una **función agregada**.

índice

Un objeto de base de datos que proporciona acceso a los datos en las filas de una tabla, basándose en valores de clave. Los índices proporcionan acceso rápido a los datos y pueden forzar unicidad en las filas de una tabla.

integridad referencial (RI)

Asegura que para cada fila en una tabla de **clave externa**, existe una fila correspondiente en la tabla de **clave principal**. También impide que se elimine una fila en una tabla de clave principal cuando existe una relación con una tabla de clave externa. En diagramas de base de datos, se debe eliminar la relación entre tablas antes de poder eliminar columnas de clave principal.

lenguaje de definición de datos (DDL)

Un lenguaje, normalmente una parte de un sistema de administración de base de datos, que se utiliza para definir todos los atributos y las propiedades de una base de datos, especialmente diseños de fila, definiciones de columna, columnas de clave (y, a veces, metodología de claves), ubicaciones de archivos y estrategia de almacenamiento.

memo

Un tipo de columna que contiene cadenas largas de texto no estructurado (típicamente más de 255 caracteres).

nombre del origen de datos (DSN)

Un término ODBC para la colección de información que se utiliza para conectar su aplicación a una base de datos ODBC particular. El Administrador del controlador de ODBC utiliza esta información para crear una conexión a la base de datos. Un DSN se puede almacenar en un archivo (un archivo DSN) o en el Registro de Windows® (un equipo DSN).

Comparar con conexión de datos.

objeto

Uno de los componentes de una base de datos: una tabla, índice, desencadenante, vista, clave, restricción, valor predeterminado, regla, tipo de datos definido por el usuario o procedimiento almacenado. También se llama un **objeto de base de datos**.

objeto de base de datos

Uno de los componentes de una base de datos: una tabla, índice, desencadenante, vista, clave, restricción, valor predeterminado, regla, tipo de datos definido por el usuario o procedimiento almacenado.

ODBC (conectividad abierta de base de datos)

Un protocolo estándar para servidores de base de datos que proporciona un lenguaje común para aplicaciones Windows® para tener acceso a una base de datos en una red. Instale controladores ODBC para varias bases de datos que le permitan conectarse a las bases de datos y tener acceso a sus datos.

operador de combinación

Un operador de comparación en una condición de combinación que determina cómo se evalúan los dos lados de la condición y qué registros se devuelven. El operador de comparación más común es la equivalencia (=).

origen de datos

Un término ODBC para la entidad que proporciona datos; similar al término **servicios de fondo**.

origen de entrada

Una tabla, vista o diagrama de base de datos que se utiliza como un origen de información para una consulta. Algunas bases de datos pueden hacer disponibles otros orígenes de entrada; por ejemplo, si utiliza un servidor de base de datos Oracle, puede utilizar sinónimos como orígenes de entrada.

página de código

Un conjunto de caracteres que utiliza un equipo para interpretar y mostrar los datos de forma adecuada. Las páginas de código normalmente corresponden a plataformas e idiomas diferentes y se utilizan en aplicaciones internacionales.

Comparar con **configuración regional**.

páginas de propiedades

Un cuadro de diálogo con fichas donde puede identificar las características de tablas, relaciones, índices, restricciones y claves. Cada objeto en un diagrama de base de datos tiene un conjunto de propiedades que determinan la definición del objeto de base de datos. Cada conjunto de fichas sólo muestra las propiedades específicas para el objeto seleccionado. Si se seleccionan múltiples objetos, las páginas de propiedades muestran las propiedades del primer objeto que seleccionó.

parámetros

Un valor dado a una variable. Un parámetro actúa como marcador de posición en una consulta o procedimiento almacenado que se puede rellenar cuando se ejecuta la consulta o el procedimiento almacenado. Los parámetros le permiten utilizar la misma consulta o procedimiento almacenado muchas veces, cada vez con valores diferentes.

Por ejemplo, puede crear una consulta con un parámetro para un apellido, después cuando ejecute la consulta, solicitarle el apellido a buscar. Los parámetros se pueden utilizar para cualquier valor literal y en algunas bases de datos, también para referencias de columnas.

persistencia

Almacén de objetos y estructuras de datos que implican la conversión de estructuras de datos complejas en un formato adecuado para el almacenamiento de archivos. Los datos persistentes se mantienen en el equipo entre sesiones.

predeterminado

Objeto de base de datos que habilita a la base de datos para insertar un valor en una columna si el usuario no introduce ninguno explícitamente.

Comparar con **predeterminado global**.

predeterminado global

Un valor predeterminado que se define para una base de datos específica y se comparte por las columnas de tablas diferentes.

Comparar con **predeterminado**.

procedimiento almacenado

Una colección precompilada de instrucciones SQL e instrucciones de flujo de control almacenadas bajo un solo nombre y procesadas como una unidad. Los procedimientos almacenados se almacenan en una base de datos; se pueden ejecutar con una llamada desde una aplicación y permite variables declaradas por el usuario, ejecución condicional y otras características eficaces de programación.

propiedad

Un atributo con nombre de un objeto. Las propiedades definen las características del objeto, como su tamaño y nombre, o el estado de un objeto, como habilitado o deshabilitado. Se pueden establecer propiedades para tablas, columnas, índices, restricciones, claves y relaciones.

proyecto de base de datos

Una colección de una o más conexiones de datos (una base de datos y la información necesaria para tener acceso a la base de datos). Cuando crea un proyecto de base de datos, puede conectarse a una o más bases de datos mediante ODBC y ver sus componentes mediante una interfaz visual de usuario que incluye un Diseñador de base de datos para diseñar y crear bases de datos y un Diseñador de consultas para crear instrucciones SQL para cualquier base de datos compatible con ODBC.

registro

Un término que se utiliza en algunos sistemas de base de datos para querer decir "**fila**". Aunque los registros y las filas tienen significados ligeramente diferentes en la teoría formal de base de datos, en la mayoría de los casos son sinónimos.

registro de transacción

Un área de almacenamiento reservada por la base de datos para seguir las transacciones realizadas a la base de datos.

regla

Un objeto de base de datos que se limita a una columna o a un tipo de datos definido por el usuario y que especifica qué datos se pueden introducir en esa columna.

Comparar con **regla global**.

regla global

Una regla que se define para una base de datos específica y se comparte por columnas de tablas diferentes.

Comparar con **regla**.

relación

Un vínculo entre tablas en que se referencia a la clave principal de una tabla con la clave externa de otra tabla. La línea de relación se representa en un diagrama de base de datos mediante una línea sólida si se fuerza la integridad referencial entre tablas o con una línea de rayas si no se fuerza la integridad para transacciones INSERT y UPDATE. Los puntos

finales de una línea de relación muestran un símbolo de clave principal para denotar

una relación de clave principal a clave externa o un símbolo de infinito para denotar el
lado de la clave externa de una relación de uno a varios.

relación de varios a varios

Una relación entre dos tablas en la que las filas de cada tabla tienen múltiples filas coincidentes en la tabla relacionada. Por ejemplo, una relación varios a varios es posible entre la tabla `autores` y la tabla `títulos` en la base de datos de ejemplo `publicaciones` porque cada autor puede haber escrito varios títulos y cada título puede tener varios autores. Las relaciones de varios a varios se mantienen utilizando una tercera tabla llamada **tabla de unión**.

relación no forzada

Un vínculo entre tablas en que se referencia las clave principal de una tabla con una clave externa de otra tabla y no se comprueba la integridad referencial durante las transacciones INSERT y UPDATE. Una relación no forzada se representa con una línea de rayas en un diagrama de base de datos.

relación reflexiva

Una relación de una columna o una combinación de columnas de una tabla a otras columnas de la misma tabla. Las relaciones reflexivas se utilizan para comparar filas dentro de la misma tabla. Por ejemplo, puede crear una relación reflexiva para buscar todos los editores que viven en la misma ciudad y tienen el mismo código postal. En consultas, esto se llama una **autocombinación**.

relación uno a uno

Una relación entre dos tablas en la que una fila de la primera tabla se puede relacionar sólo con una fila de la segunda tabla y una fila de la segunda tabla se puede relacionar sólo con una fila de la primera tabla. Este tipo de relación no es normal. Un ejemplo es la relación entre una tabla `empleado` y una tabla `buzones` que se utiliza para almacenar información acerca de direcciones de correo electrónico en una compañía. Las tablas tienen una relación uno a uno debido a que cada empleado tiene un buzón y cada buzón está asignado a un solo empleado.

relación uno a varios

Una relación entre dos tablas en la que una fila de la primera tabla se puede relacionar con una o más filas de la segunda tabla, pero una fila de la segunda tabla se puede relacionar sólo con una fila de la primera tabla. Una relación uno a varios típica es entre la tabla `títulos` en la base de datos de ejemplo `publicaciones`, en la que cada editor puede relacionarse con varios títulos, pero cada título se puede relacionar sólo con un editor.

restricción

Una restricción situada sobre el valor que se puede introducir en una columna o una fila. Por ejemplo, edad no puede ser < 0 o > 110 .

Comparar con **restricción de comprobación**, **restricción única**.

restricción única

Una restricción que fuerza la integridad de entidad en una clave no principal. Esto asegura que se introducen valores no duplicados en la columnas en las que se coloca la restricción. Se permiten valores NULL mientras las filas son únicas en la tabla.

restricciones de comprobación

Especifica los valores de datos que se aceptan en una columna. Puede aplicar restricciones de comprobación para múltiples columnas y puede aplicar múltiples restricciones de comprobación a una sola columna. Cuando se elimina una tabla, también se eliminan todas las restricciones de comprobación.

secuencia de comandos de cambios

Un archivo de texto que contiene instrucciones SQL para todos los cambios realizados en una base de datos, en el orden en que se realizaron durante una sesión de modificación. Cada registro de cambios se guarda en un archivo de texto separado con una extensión .sql. El registro de cambios crea un registro de los cambios realizados a la base de datos los cuales se pueden aplicar de forma inversa a la base de datos posteriormente mediante una herramienta como isql.

servicios de fondo

El servidor o el motor en un sistema de base de datos. En general, los servicios de fondo se refieren al sistema que es el repositorio de los datos, en oposición al cliente, que presenta datos al usuario. En un sistema cliente/servidor, los servicios de fondo se corresponden con el servidor. En un sistema de base de datos integrado, como Microsoft® Visual FoxPro™ o Microsoft® Excel, los servicios de fondo se corresponden con la funcionalidad del núcleo de la base de datos del producto. Como regla, los servicios de fondo contienen el código que controla no sólo el almacenamiento de datos, sino también los índices, la validación, los desencadenantes, la integridad referencial y otras funciones de base de datos.

servidor

Un equipo en una red que controla el acceso a datos.

Comparar con **cliente/servidor**.

SQL

Lenguaje de consulta estructurado, un lenguaje de consulta de base de datos y programación.

subconjunto

Una selección de tablas y líneas de relación entre ellas que es parte de un diagrama de base de datos más grande. Esta selección se puede copiar a un nuevo diagrama de base de datos. Esto se llama seccionar el diagrama.

subconsulta

Una instrucción SELECT que se define en una consulta. El conjunto de resultados de la instrucción SELECT incorporada pasa a ser parte de la condición de búsqueda para la segunda consulta. Por ejemplo, la siguiente consulta utiliza una subconsulta para buscar todos los proveedores en Suecia, después utiliza el resultado en una cláusula WHERE para buscar todos los productos de los proveedores seleccionados:

```
WHERE id_proveedor IN  
      (SELECT id_proveedor  
       FROM proveedor  
       WHERE (país= 'Suecia'))
```

T

tabla

Una estructura de datos que contiene una colección de filas (o registros) que tienen columnas asociadas (o campos). Es el equivalente lógico de un archivo de base de datos.

tabla de unión

Una tabla que tiene asociaciones con otras tablas y que se utiliza indirectamente como una asociación entre esas dos tablas. También llamada **tabla de vinculación**.

tabla de vinculación

Una tabla que tiene asociaciones con otras tablas y que se utiliza indirectamente como una asociación entre esas dos tablas. También llamada **tabla de unión**.

tipo de datos definido por el usuario

Una definición del tipo de datos que puede contener una columna. Se crea por el usuario y se define en los términos de los tipos de datos existentes del sistema. Las reglas y valores predeterminados pueden limitar los tipos de datos definidos por el usuario, pero no los tipos de datos del sistema.

UNC (código de nombre universal)

Una convención de denominación para archivos que proporcionan significados independientes del equipo para la localización del archivo. Un nombre UNC normalmente incluirá una referencia a una carpeta compartida y un nombre de archivo accesible en una red a menos que especifique una letra de unidad y la ruta de acceso. Por ejemplo, para tener acceso a una base de datos llamada Neptuno.mdb en un directorio compartido llamado Ejemplos en el equipo llamado MiEstación, podría utilizar el nombre UNC \\MiEstación\Ejemplos\Neptuno.mdb.

Las convenciones de denominación UNC son particularmente útiles en aplicaciones basadas en Web debido a que permiten hacer referencia a datos que no están almacenados necesariamente en un servidor Web particular.

vista

Una tabla virtual que se genera mediante una consulta cuya definición se almacena en la base de datos. Por ejemplo, una vista se puede definir conteniendo tres de las cinco columnas disponibles de una tabla, creada para limitar el acceso a cierta información. Las vistas se pueden tratar como tablas para la mayoría de operaciones de base de datos, incluyendo consultas de selección y bajo algunas circunstancias consultas de actualización, inserción y eliminación. Cualquier operación realizada en vistas afecta realmente a los datos de la tabla o tablas en las que se basa la vista.