

Recuperación de datos

Contenido

Introducción	1
Recuperar datos de tablas mediante la instrucción SELECT	2
Filtrar los datos	8
Dar formato a los conjuntos de resultados	21
Cómo se procesan las consultas	29
Consideraciones acerca del rendimiento	30

Notas para el instructor

Este módulo proporciona a los alumnos los conocimientos y técnicas necesarios para ejecutar consultas básicas con la instrucción `SELECT`, lo que incluye ordenar datos, eliminar duplicados y cambiar el formato del conjunto de resultados. El módulo finaliza con una descripción de la forma en que se procesan las consultas.

Al final de este módulo, los alumnos serán capaces de:

- Recuperar datos de tablas mediante la instrucción `SELECT`.
- Filtrar los datos con distintas condiciones de búsqueda y la cláusula `WHERE`.
- Dar formato a los conjuntos de resultados.
- Describir cómo se procesan las consultas.
- Describir los aspectos relacionados con el rendimiento que afectan a la recuperación de datos.

Introducción

Objetivos de la diapositiva

Proporcionar una introducción a los temas y objetivos del módulo.

Explicación previa

En este módulo aprenderá a recuperar datos mediante consultas básicas.

- Recuperar datos de tablas mediante la instrucción **SELECT**
- Filtrar los datos
- Dar formato a los conjuntos de resultados
- Cómo se procesan las consultas
- Consideraciones acerca del rendimiento

Este módulo proporciona a los alumnos los conocimientos y técnicas necesarios para ejecutar consultas básicas con la instrucción **SELECT**, lo que incluye ordenar datos, eliminar duplicados y cambiar el formato del conjunto de resultados. El módulo finaliza con una descripción de la forma en que se procesan las consultas.

Al final de este módulo, los alumnos serán capaces de:

- Recuperar datos de tablas mediante la instrucción **SELECT**.
- Filtrar los datos con distintas condiciones de búsqueda en la cláusula **WHERE**.
- Dar formato a los conjuntos de resultados.
- Describir cómo se procesan las consultas.
- Describir los aspectos relacionados con el rendimiento que afectan a la recuperación de datos.

◆ Recuperación de datos mediante la instrucción SELECT

Objetivos de la diapositiva

Enumerar los temas tratados en las secciones siguientes.

Explicación previa

Para obtener datos de las tablas puede utilizarse la instrucción SELECT, en la que se especifican filas y columnas.

- Uso de la instrucción SELECT
- Especificación de columnas
- Uso de la cláusula WHERE para especificar filas

Para trabajar con datos debe seleccionar cuáles desea extraer de las tablas. Puede usar la instrucción SELECT con el objeto de especificar las columnas y filas de datos que desea recuperar de las tablas.

Uso de la instrucción SELECT

Objetivos de la diapositiva

Explicar el uso de la instrucción SELECT para obtener datos de una tabla.

Explicación previa

La instrucción SELECT permite recuperar datos.

- La lista de selección especifica las columnas
- La cláusula WHERE especifica las filas
- La cláusula FROM especifica la tabla

Sintaxis parcial

```
SELECT [ALL | DISTINCT] <listaSelección>  
FROM <tablaOrigen> [,...n]  
WHERE <condiciónBúsqueda>
```

La instrucción SELECT permite recuperar datos.

Sintaxis parcial

```
SELECT [ ALL | DISTINCT ] <listaSelección>  
FROM {<tablaOrigen>} [, ...n]  
[ WHERE <condiciónBúsqueda> ]
```

Puede utilizar la instrucción SELECT para especificar las filas y columnas que desea obtener de una tabla:

- La lista de selección especifica las columnas que se deben obtener.
- La cláusula WHERE especifica las filas que se deben obtener. Cuando se utilizan condiciones de búsqueda en la cláusula WHERE, se puede restringir el número de filas devueltas mediante operadores de comparación, cadenas de caracteres y operadores lógicos como condiciones de búsqueda.
- La cláusula FROM especifica la tabla de la que se obtienen las filas y columnas.

Especificación de columnas


Objetivos de la diapositiva

Mostrar cómo seleccionar columnas de una tabla.

Explicación previa

Para obtener columnas específicas de una tabla, puede incluirlas en la lista de selección.

```
USE northwind
SELECT employeeid, lastname, firstname, title
FROM employees
GO
```



<i>employeeid</i>	<i>lastname</i>	<i>firstname</i>	<i>title</i>
1	Davolio	Nancy	Sales Representative
2	Fuller	Andrew	Vice President, Sales
3	Leverling	Janet	Sales Representative
4	Peacock	Margaret	Sales Representative
5	Buchanan	Steven	Sales Manager
6	Suyama	Michael	Sales Representative
7	King	Robert	Sales Representative
8	Callahan	Laura	Inside Sales Coordinator
9	Dodsworth	Anne	Sales Representative

Para obtener columnas específicas de una tabla puede incluirlas en la lista de selección.

La lista de selección contiene las columnas, expresiones o palabras clave que se seleccionan, o la variable local que se asigna. Las opciones que pueden utilizarse en la lista de selección son:

Sintaxis parcial

<listaSelección> ::=

```
{
  *
  | { nombreTabla | nombreVista | aliasTabla }. *
  | { nombreColumna | expresión | IDENTITYCOL | ROWGUIDCOL }
    [ [AS] aliasColumna ]
    | aliasColumna = expresión
}
```

Al especificar las columnas que desea obtener, tenga en cuenta los siguientes hechos e instrucciones:

- La lista de selección obtiene y muestra las columnas en el orden especificado.
- Separe los nombres de columna con comas, excepto después del último.
- Evite o reduzca al mínimo el uso del asterisco (*) en la lista de selección. El asterisco se emplea para obtener todas las columnas de una tabla.

Ejemplo

En este ejemplo se obtienen las columnas **employeeid**, **lastname**, **firstname** y **title** de todos los empleados de la tabla **employees**.

```
USE northwind
SELECT employeeid, lastname, firstname, title
FROM employees
GO
```

Resultado

employeeid	lastname	firstname	title
1	Davolio	Nancy	Sales Representative
2	Fuller	Andrew	Vice President, Sales
3	Leverling	Janet	Sales Representative
4	Peacock	Margaret	Sales Representative
5	Buchanan	Steven	Sales Manager
6	Suyama	Michael	Sales Representative
7	King	Robert	Sales Representative
8	Callahan	Laura	Inside Sales Coordinator
9	Dodsworth	Anne	Sales Representative

(9 filas afectadas)

Uso de la cláusula WHERE para especificar filas


Objetivos de la diapositiva

Indicar cómo obtener filas con la cláusula WHERE.

Explicación previa

El uso de la cláusula WHERE permite obtener filas en función de las condiciones de búsqueda que se especifiquen.

```
USE northwind
SELECT employeeid, lastname, firstname, title
FROM employees
WHERE employeeid = 5
GO
```



employeeid	lastname	firstname	title
5	Buchanan	Steven	Sales Manager

Sugerencia

Compare el conjunto de resultados de la diapositiva anterior con el resultado de ésta.

Señale que el uso de la cláusula WHERE restringe el número de filas que se devuelven.

El uso de la cláusula WHERE permite obtener filas en función de las condiciones de búsqueda que se especifiquen. Las condiciones de búsqueda de la cláusula WHERE pueden contener una lista ilimitada de predicados.

```
<condiciónBúsqueda> ::=
{ [ NOT ] <predicado> | ( <condiciónBúsqueda> ) }
[ { AND | OR } [ NOT ] { <predicado> | ( <condiciónBúsqueda> ) } ]
[ , ...n ]
```

El marcador de posición *predicado* indica las expresiones que pueden incluirse en la cláusula WHERE. Un predicado puede contener las opciones siguientes:

La sintaxis incluida aquí se encuentra en el tema "Condición de búsqueda (T-SQL)" de los Libros en pantalla de SQL Server, en lugar de en el tema "Instrucción SELECT".

```
<predicado> ::=
{
  expresión { = | < | > | >= | <= } expresión
  | expresiónCadena [NOT] LIKE expresiónCadena
  [ESCAPE 'carácterEscape']
  | expresión [NOT] BETWEEN expresión AND expresión
  | expresión IS [NOT] NULL
  | CONTAINS
  ( {columna | * }, '<contieneCondiciónBúsqueda>' )
  | FREETEXT ( {columna | * }, 'cadenaTextoLibre' )
  | expresión [NOT] IN (subconsulta | expresión [, ...n])
  | expresión { = | < | > | >= | <= }
  { ALL | SOME | ANY } (subconsulta)
  | EXISTS (subconsulta)
}
```


Al especificar filas con la cláusula **WHERE**, tenga en cuenta los siguientes hechos e instrucciones:

- Utilice comillas simples para todos los datos de tipo **char**, **nchar**, **varchar**, **nvarchar**, **text**, **datetime** y **smalldatetime**.
- Utilice una cláusula **WHERE** para limitar el número de filas que se devuelven al utilizar la instrucción **SELECT**.

Ejemplo

En este ejemplo se obtienen las columnas **employeeid**, **lastname**, **firstname** y **title** de la tabla **employees** para el empleado que tiene un **employeeid** de 5.

```
USE northwind
SELECT employeeid, lastname, firstname, title
FROM employees
WHERE employeeid = 5
GO
```

Resultado

employeeid	lastname	firstname	title
5	Buchanan	Steven	Sales Manager

(1 fila afectada)

◆ Filtros de datos

Objetivos de la diapositiva

Describir los distintos tipos de condiciones de búsqueda que se utilizan en la cláusula WHERE.

Explicación previa

A veces se desea limitar el conjunto de resultados que devuelve una consulta.

- Uso de los operadores de comparación
- Uso de comparaciones de cadenas
- Uso de operadores lógicos
- Obtención de un intervalo de valores
- Uso de una lista de valores como criterio de búsqueda
- Obtención de valores desconocidos

A veces se desea limitar el conjunto de resultados que devuelve una consulta. Para limitar los resultados se puede especificar condiciones de búsqueda en una cláusula WHERE con las que filtrar los datos. No hay límite en el número de condiciones de búsqueda que se pueden incluir en una instrucción SELECT. La tabla siguiente describe el tipo de filtro y la correspondiente condición de búsqueda que se puede usar para filtrar los datos.

Tipo de filtro	Condición de búsqueda
Operadores de comparación	=, >, <, >=, <= y <>
Comparaciones de cadenas	LIKE y NOT LIKE
Operadores lógicos: combinación de condiciones	AND, OR
Operador lógico: negación	NOT
Intervalo de valores	BETWEEN y NOT BETWEEN
Listas de valores	IN y NOT IN
Valores desconocidos	IS NULL e IS NOT NULL

Uso de los operadores de comparación

Objetivos de la diapositiva


Mostrar cómo obtener subconjuntos de filas mediante operadores de comparación.

Explicación previa

Puede utilizar operadores de comparación, como mayor que (>), menor que (<) e igual a (=) para seleccionar filas en función de comparaciones.

Ejemplo 1

```
USE northwind
SELECT lastname, city
FROM employees
WHERE country = 'USA'
GO
```



<i>lastname</i>	<i>city</i>
Davolio	Seattle
Fuller	Tacoma
Leverling	Kirkland
Peacock	Redmond
Callahan	Seattle

Los operadores de comparación permiten comparar los valores de una tabla con un valor o expresión especificados. También puede utilizarlos para comprobar si se cumple una condición. Los operadores de comparación comparan columnas o variables de tipos de datos compatibles. En la tabla siguiente se enumeran estos operadores.

Operador	Descripción
=	Igual a
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
<>	No igual a

Nota Evite el uso de NOT en las condiciones de búsqueda. Al usarlo se evalúan todas las filas de la tabla y la obtención de datos puede ralentizarse.

Ejemplo 1

En este ejemplo se obtiene el apellido y la ciudad de los empleados de la tabla **employees** residentes en Estados Unidos.

```
USE northwind
SELECT lastname, city
FROM employees
WHERE country = 'USA'
GO
```

Resultado

lastname	city
Davolio	Seattle
Fuller	Tacoma
Leverling	Kirkland
Peacock	Redmond
Callahan	Seattle

(5 filas afectadas)

Ejemplo 2

En este ejemplo se obtienen las columnas **orderid** y **customerid** de la tabla **orders** con fecha de pedido anterior al 1/8/96.

```
USE northwind
SELECT orderid, customerid
FROM orders
WHERE orderdate < '1/8/96'
GO
```

Resultado

orderid	customerid
10248	VINET
10249	TOMSP
10250	HANAR
10251	VICTE
10252	SUPRD
10253	HANAR

.

.

.

.

.

.

(22 filas afectadas)

Uso de comparaciones de cadenas

Objetivos de la diapositiva

Mostrar cómo obtener filas con la condición de búsqueda LIKE en combinación con caracteres comodín.

Explicación previa

La condición de búsqueda LIKE permite seleccionar filas por comparación entre cadenas de caracteres.

```
USE northwind
SELECT companyname
FROM customers
WHERE companyname LIKE '%Restaurant%'
GO
```

companyname

GROSELLA-Restaurante

Lonesome Pine Restaurant

Tortuga Restaurante

Puede utilizar la condición de búsqueda LIKE en combinación con caracteres comodín para seleccionar filas por comparación entre cadenas de caracteres. Al utilizar la condición de búsqueda LIKE, tenga en cuenta los hechos siguientes:

- Todos los caracteres de la cadena modelo son significativos, incluidos los espacios en blanco al comienzo o al final.
- LIKE sólo puede utilizarse con datos de tipo **char**, **nchar**, **varchar**, **nvarchar** o **datetime**.

Tipos de caracteres comodín

Utilice los siguientes caracteres comodín para formar el criterio de búsqueda de la cadena de caracteres:

Comodín	Descripción
%	Cualquier cadena de cero o más caracteres.
_	Cualquier carácter individual.
[]	Cualquier carácter individual contenido en el intervalo o conjunto especificado.
[^]	Cualquier carácter individual <i>no</i> contenido en el intervalo o conjunto especificado.

Ejemplos de uso de caracteres comodín

La tabla siguiente contiene ejemplos de uso de comodines con la condición de búsqueda LIKE.

Expresión	Devuelve
LIKE 'BR%'	Todos los nombres que comiencen por las letras BR.
LIKE 'Br%'	Todos los nombres que comiencen por las letras Br.
LIKE '%een'	Todos los nombres que terminen con las letras een.
LIKE '%en%'	Todos los nombres que contengan las letras en.
LIKE '_en'	Todos los nombres de tres letras que terminen con las letras en.
LIKE '[CK]%'	Todos los nombres que comiencen por C o por K.
LIKE '[S-V]ing'	Todos los nombres de cuatro letras que terminen con las letras ing y comiencen por cualquier letra comprendida entre S y V.
LIKE 'M[^c]%'	Todos los nombres que comiencen por la letra M y cuya segunda letra no sea c.

Ejemplo

En este ejemplo se obtienen las compañías de la tabla **customers** que contengan la palabra "restaurant" en su nombre.

```
USE northwind
SELECT companyname
FROM customers
WHERE companyname LIKE '%Restaurant%'
GO
```

Resultado

companyname
GROSELLA-Restaurante
Lonesome Pine Restaurant
Tortuga Restaurante

(3 filas afectadas)

Uso de operadores lógicos

Objetivos de la diapositiva


Mostrar cómo combinar varias expresiones mediante operadores lógicos.

Explicación previa

Al ejecutar una consulta, puede ser conveniente limitar el número de filas que SQL Server devuelve. Para ello, puede utilizar operadores lógicos que combinen dos o más expresiones.

Ejemplo 1

```
USE northwind
SELECT productid, productname, supplierid, unitprice
FROM products
WHERE (productname LIKE 'T%' OR productid = 46)
AND (unitprice > 16.00)
GO
```



productid	productname	supplierid	unitprice
14	Tofu	6	23.25
29	Thüringer Rostbratwurst	12	123.79
62	Tarte au sucre	29	49.3

Puede utilizar los operadores lógicos AND, OR y NOT para combinar un conjunto de expresiones y afinar el proceso de las consultas. Los resultados de una consulta pueden variar según el agrupamiento de las expresiones y el orden de las condiciones de búsqueda.

Al utilizar operadores lógicos, tenga en cuenta las instrucciones siguientes:

- Utilice el operador AND para obtener las filas que cumplan todos los criterios de búsqueda.
- Utilice el operador OR para obtener las filas que cumplan alguno de los criterios de búsqueda.
- Utilice el operador NOT para negar la expresión especificada a continuación.

Uso de los paréntesis

Utilice paréntesis cuando especifique dos o más expresiones como criterio de búsqueda. El uso de paréntesis permite:

- Agrupar expresiones.
- Cambiar el orden de evaluación.
- Hacer más legibles las expresiones.

Orden de las condiciones de búsqueda

Al utilizar más de un operador lógico en una instrucción, tenga en cuenta los hechos siguientes:

- Microsoft® SQL Server™ 2000 evalúa en primer lugar el operador NOT, después el operador AND y, por último, el operador OR.
- Si todos los operadores de una expresión son del mismo nivel, el orden de precedencia es de izquierda a derecha.

Ejemplo 1

En el ejemplo siguiente se obtienen todos los productos cuyo nombre comienza por la letra T o tienen el número de identificación 46, y cuyo precio es superior a 16,00 dólares.

Sugerencia

Compare la consulta del ejemplo 1 con la del 2. Señale qué expresiones están agrupadas de forma distinta y sus conjuntos de resultados diferentes.

```
USE northwind
SELECT productid, productname, supplierid, unitprice
FROM products
WHERE (productname LIKE 'T%' OR productid = 46)
AND (unitprice > 16.00)
GO
```

Resultado

productid	productname	supplierid	unitprice
14	Tofu	6	23.25
29	Thüringer Rostbratwurst	12	123.79
62	Tarte au sucre	29	49.3

(3 filas afectadas)

Ejemplo 2

En el ejemplo siguiente se obtienen los productos cuyo nombre comienza por la letra T o que tienen el número de identificación 46 y un precio superior a 16,00 dólares. Compare la consulta del ejemplo 1 con la del ejemplo 2. Observe que, al ser distinto el agrupamiento de las expresiones, el procesamiento de las consultas es diferente y los conjuntos de resultados difieren.

```
USE northwind
SELECT productid, productname, supplierid, unitprice
FROM products
WHERE (productname LIKE 'T%')
OR (productid = 46 AND unitprice > 16.00)
GO
```

Resultado

productid	productname	supplierid	unitprice
54	Tourtière	25	7.45
62	Tarte au sucre	29	49.3
23	Tunnbröd	9	9
19	Teatime Chocolate Biscuits	8	9.2
14	Tofu	6	23.25
29	Thüringer Rostbratwurst	12	123.79

(6 filas afectadas)

Obtención de un intervalo de valores

Objetivos de la diapositiva

Mostrar cómo obtener datos con la condición de búsqueda BETWEEN.

Explicación previa

Para obtener filas contenidas en un intervalo de valores, puede utilizar la condición de búsqueda BETWEEN.

Ejemplo 1

```
USE northwind
SELECT productname, unitprice
FROM products
WHERE unitprice BETWEEN 10 AND 20
GO
```



productname	unitprice
Chai	18
Chang	19
Aniseed Syrup	10
Genen Shouyu	15.5
Pavlova	17.45
Sir Rodney's Scones	10
...	...

Puede utilizar la condición de búsqueda BETWEEN en la cláusula WHERE para obtener las filas contenidas en un intervalo de valores específico. Al utilizar la condición de búsqueda BETWEEN, tenga en cuenta los siguientes hechos e instrucciones:

- SQL Server incluye los valores inicial y final en el conjunto de resultados.
- Es preferible el uso de BETWEEN al de una expresión que incluya el operador AND y dos operadores de comparación ($\geq x$ AND $\leq y$). Sin embargo, para buscar en un intervalo exclusivo en el que las filas devueltas no contengan los valores inicial y final, debe utilizar una expresión de este tipo ($> x$ AND $< y$).
- Puede utilizar la condición de búsqueda NOT BETWEEN para obtener las filas que estén fuera del intervalo especificado. Tenga en cuenta que el uso de condiciones NOT puede ralentizar la recuperación de los datos.

Ejemplo 1

En este ejemplo se obtiene el nombre y el precio por unidad de todos los productos cuyo precio se encuentra entre 10,00 y 20,00 dólares. Observe que el conjunto de resultados incluye los extremos del intervalo.

```
USE northwind
SELECT productname, unitprice
FROM products
WHERE unitprice BETWEEN 10 AND 20
GO
```

Resultado

productname	unitprice
Chai	18
Chang	19
Aniseed Syrup	10
Genen Shouyu	15.5
Pavlova	17.45
Sir Rodney's Scones	10
.	
.	
.	
(29 filas afectadas)	

Ejemplo 2

En este ejemplo se obtiene el nombre y el precio por unidad de todos los productos cuyo precio se encuentra entre 10 y 20 dólares. Observe que en el conjunto de resultados no se incluyen los extremos del intervalo.

```
USE northwind
SELECT productname, unitprice
FROM products
WHERE (unitprice > 10)
AND (unitprice < 20)
GO
```

Resultado

productname	unitprice
Chai	18
Chang	19
Genen Shouyu	15.5
Pavlova	17.45
.	
.	
.	
(25 filas afectadas)	

Uso de una lista de valores como criterio de búsqueda

Objetivos de la diapositiva


Mostrar cómo obtener filas con la condición de búsqueda IN.

Explicación previa

Puede que desee obtener las filas que coincidan con una lista de valores determinada. Para ello, puede utilizar la condición de búsqueda IN en la cláusula WHERE.

Ejemplo 1

```
USE northwind
SELECT companyname, country
FROM suppliers
WHERE country IN ('Japan', 'Italy')
GO
```



companyname	country
Tokyo Traders	Japan
Mayumi's	Japan
Formaggi Fortini s.r.l.	Italy
Pasta Buttini s.r.l.	Italy

Sugerencia

Señale que SQL Server resuelve los ejemplos 1 y 2 de la misma forma y devuelve el mismo conjunto de resultados. En el ejemplo 1 se utiliza el operador de búsqueda IN, mientras que en el ejemplo 2 se emplean dos operadores igual a (=) conectados con el operador OR.

Puede utilizar la condición de búsqueda IN en la cláusula WHERE para obtener las filas que coincidan con una lista de valores especificada. Al utilizar la condición de búsqueda IN, tenga en cuenta las instrucciones siguientes:

- Puede utilizar la condición de búsqueda IN o un conjunto de expresiones de comparación conectadas con el operador OR. SQL Server las resuelve de la misma forma y los conjuntos de resultados devueltos son idénticos.
- No incluya un valor NULL en la condición de búsqueda. Un valor NULL en la lista de condiciones de búsqueda da como resultado la expresión = NULL. Puede producir conjuntos de resultados inesperados.
- Utilice la condición NOT IN para obtener las filas cuyos valores no se encuentren en la lista especificada. Tenga en cuenta que el uso de condiciones NOT puede hacer que los datos se recuperen más lentamente.

Ejemplo 1

En este ejemplo se obtiene la lista de compañías de la tabla **suppliers** que se encuentran en Japón o en Italia.

```
USE northwind
SELECT companyname, country
FROM suppliers
WHERE country IN ('Japan', 'Italy')
GO
```

Resultado

companyname	country
Tokyo Traders	Japan
Mayumi's	Japan
Formaggi Fortini s.r.l.	Italy
Pasta Buttini s.r.l.	Italy

(4 filas afectadas)

Ejemplo 2

En este ejemplo también se obtiene una lista con las compañías de la tabla **suppliers** que se encuentran en Japón o en Italia. Observe que en lugar de la condición de búsqueda IN se utilizan dos expresiones con un operador de comparación combinadas con el operador OR. El conjunto de resultados es idéntico al del ejemplo 1.

```
USE northwind
SELECT companyname, country
FROM suppliers
WHERE country = 'Japan' OR country = 'Italy'
GO
```

Resultado

companyname	country
Tokyo Traders	Japan
Mayumi's	Japan
Formaggi Fortini s.r.l.	Italy
Pasta Buttini s.r.l.	Italy

(4 filas afectadas)

Obtención de valores desconocidos

Objetivos de la diapositiva

Mostrar cómo obtener filas que contienen valores desconocidos.

Explicación previa

Para obtener filas que contengan valores desconocidos puede especificar IS NULL en la cláusula WHERE.

```
USE northwind
SELECT companyname, fax
FROM suppliers
WHERE fax IS NULL
GO
```



<i>companyname</i>	<i>fax</i>
Exotic Liquids	NULL
New Orleans Cajun Delights	NULL
Tokyo Traders	NULL
Cooperativa de Quesos 'Las Cabras'	NULL
...	...

Una columna tiene un valor NULL cuando no se ha especificado ningún valor para ella durante la entrada de datos y no tiene definido un valor predeterminado. Un valor NULL no es lo mismo que cero (que es un valor numérico) o blanco (que es un valor de carácter).

Puede utilizar la condición de búsqueda IS NULL para obtener las filas en las que falte información en una columna específica. Al obtener las filas con valores desconocidos, tenga en cuenta los siguientes hechos e instrucciones:

- Los valores NULL causan errores en todas las comparaciones porque no se evalúan como iguales entre sí.
- En la instrucción CREATE TABLE se define si las columnas permiten valores NULL.
- Puede utilizar la condición de búsqueda IS NOT NULL para obtener las filas con valores conocidos en las columnas especificadas.

Ejemplo

En este ejemplo se obtiene la lista de compañías de la tabla **suppliers** cuya columna **fax** contiene un valor NULL.

```
USE northwind
SELECT companyname, fax
FROM suppliers
WHERE fax IS NULL
GO
```

Resultado

companyname	fax
Exotic Liquids	NULL
New Orleans Cajun Delights	NULL
Tokyo Traders	NULL
Cooperativa de Quesos 'Las Cabras'	NULL
.	
.	
.	

(16 filas afectadas)

◆ Dar formato a los conjuntos de resultados

Objetivos de la diapositiva

Mostrar cómo dar formato a los conjuntos de resultados.

Explicación previa

Para que los conjuntos de resultados sean más legibles, puede ordenar los datos, eliminar filas duplicadas, cambiar el nombre de las columnas o utilizar literales.

- Ordenación de los datos
- Eliminación de filas duplicadas
- Cambio del nombre de las columnas
- Uso de literales

Si desea mejorar la legibilidad de un conjunto de resultados, puede cambiar su orden, eliminar filas duplicadas, reemplazar el nombre de las columnas por alias o utilizar literales para sustituir valores. Estas opciones de formato no cambian los datos, sino sólo su presentación.

Ordenación de los datos

Objetivos de la diapositiva


Mostrar el uso de la cláusula ORDER BY.

Explicación previa

Después de decidir las columnas y filas que desea recuperar, puede ordenar el conjunto de resultados con la cláusula ORDER BY.

Ejemplo 1

```
USE northwind
SELECT productid, productname, categoryid, unitprice
FROM products
ORDER BY categoryid, unitprice DESC
GO
```



productid	productname	categoryid	unitprice
38	Côte de Blaye	1	263.5000
43	Ipoh Coffee	1	46.0000
2	Chang	1	19.0000
...
63	Vegie-spread	2	43.9000
8	Northwoods Cranberry Sauce	2	40.0000
61	Sirop d'érable	2	28.5000
...

Sugerencia

Compare las columnas especificadas en las cláusulas ORDER BY de los ejemplos 1 y 2. En el ejemplo 2, las columnas especificadas se reemplazan por sus posiciones relativas en la lista de selección. Los conjuntos de resultados de los ejemplos 1 y 2 son idénticos.

Puede utilizar la cláusula ORDER BY para ordenar las filas del conjunto de resultados de forma ascendente (ASC) o descendente (DESC). Cuando utilice la cláusula ORDER BY, tenga en cuenta los siguientes hechos e instrucciones:

- El tipo de orden se especifica cuando se instala SQL Server. Puede ejecutar el procedimiento almacenado del sistema **sp_helpsort** para determinar el tipo de orden definido para la base de datos durante la instalación.
- SQL Server no garantiza ningún orden en el conjunto de resultados a menos que se especifique uno con la cláusula ORDER BY.
- De forma predeterminada, SQL Server ordena los datos de forma ascendente.
- No es necesario que las columnas incluidas en la cláusula ORDER BY aparezcan en la lista de selección.
- Las columnas especificadas en la cláusula ORDER BY no pueden tener más de 8060 bytes.
- Es posible ordenar por nombres de columna, valores calculados o expresiones.
- En la cláusula ORDER BY, puede hacer referencia a las columnas por su posición en la lista de selección. Las columnas se evalúan de la misma forma y el conjunto de resultados es el mismo.
- La cláusula ORDER BY no debe utilizarse para columnas de tipo **text** o **image**.

Sugerencia El uso de índices adecuados puede hacer más eficientes las ordenaciones con ORDER BY.

Ejemplo 1

En este ejemplo se obtiene la identificación, nombre, categoría y precio por unidad de cada producto de la tabla **products**. De forma predeterminada, el conjunto de resultados se ordena por categorías en orden ascendente y, en cada categoría, las filas se ordenan por precio unitario en orden descendente.

```
USE northwind
SELECT productid, productname, categoryid, unitprice
FROM products
ORDER BY categoryid, unitprice DESC
GO
```

Resultado

productid	productname	categoryid	unitprice
38	Côte de Blaye	1	263.5000
43	Ipoh Coffee	1	46.0000
2	Chang	1	19.0000
1	Chai	1	18.0000
35	Steeleye Stout	1	18.0000
39	Chartreuse verte	1	18.0000
76	Lakkalikööri	1	18.0000
70	Outback Lager	1	15.0000
34	Sasquatch Ale	1	14.0000
67	Laughing Lumberjack Lager	1	14.0000
75	Rhönbräu Klosterbier	1	7.7500
24	Guaraná Fantástica	1	4.5000
63	Veggie-spread	2	43.9000
8	Northwoods Cranberry Sauce	2	40.0000
61	Sirop d'érable	2	28.5000
6	Grandma's Boysenberry Spread	2	25.0000
.			
.			
.			

(77 filas afectadas)

Ejemplo 2

Este ejemplo es similar al ejemplo 1. La única diferencia son los números especificados en la cláusula ORDER BY, que indican posiciones de columnas en la lista de selección. SQL Server procesa las dos consultas de la misma forma y devuelve el mismo conjunto de resultados.

```
USE northwind
SELECT productid, productname, categoryid, unitprice
FROM products
ORDER BY 3, 4 DESC
GO
```

Eliminación de filas duplicadas

Objetivos de la diapositiva

Mostrar cómo eliminar filas duplicadas con la condición de búsqueda DISTINCT.

Explicación previa

Si desea eliminar las filas duplicadas del conjunto de resultados, especifique la cláusula DISTINCT en la instrucción SELECT.

Ejemplo 1

```
USE northwind
SELECT DISTINCT country
FROM suppliers
ORDER BY country
GO
```

country
Australia
Brazil
Canada
Denmark
Finland
France
Germany
Italy
Japan
Netherlands
Norway
Singapore
Spain
Sweden
UK
USA

Sugerencia

Ejecute una consulta similar a la de la diapositiva sin utilizar la cláusula DISTINCT.

Haga que los alumnos comparen el número de filas obtenidas.

Si necesita obtener una lista de valores únicos, puede utilizar la cláusula DISTINCT para eliminar las filas duplicadas del conjunto de resultados. Al utilizar la cláusula DISTINCT, tenga en cuenta los hechos siguientes:

- El conjunto de resultados incluirá todas las filas que cumplan la condición de búsqueda de la instrucción SELECT, a menos que se especifique una cláusula DISTINCT.
- La combinación de valores de la lista de selección determina si las filas están duplicadas o no.
- Las filas que contengan una combinación única de valores se incluyen en el conjunto de resultados.
- La cláusula DISTINCT utiliza un orden aleatorio en el conjunto de resultados, a menos que se especifique también la cláusula ORDER BY.
- Si se especifica una cláusula DISTINCT, la cláusula ORDER BY debe incluir columnas que aparezcan en el conjunto de resultados.

Ejemplo 1

En este ejemplo se obtienen todas las filas de la tabla **suppliers**, pero sólo se muestra el nombre de cada país una vez.

```
USE northwind
SELECT DISTINCT country
FROM suppliers
ORDER BY country
GO
```

Resultado

country

Australia
Brazil
Canada
Denmark
Finland
France
Germany
Italy
Japan
Netherlands
Norway
Singapore
Spain
Sweden
UK
USA

(16 filas afectadas)

Ejemplo 2

En este ejemplo no se especifica la cláusula **DISTINCT**. Se obtienen todas las filas de la tabla **suppliers** en orden descendente. Observe que se incluyen todos los casos de cada país.

```
USE northwind
SELECT country
  FROM suppliers
 ORDER BY country
GO
```

Resultado

country

Australia
Australia
Brazil
Canada
Canada
Denmark
Finland
France
France
France
Germany
Germany
Germany
Italy
Italy
Japan
Japan
Netherlands
Norway
Singapore
Spain
.
.
.

(29 filas afectadas)

Cambio del nombre de las columnas

Objetivos de la diapositiva

Mostrar cómo puede cambiar el nombre de las columnas para mejorar la legibilidad.

Explicación previa

Puede utilizar alias para cambiar el nombre de las columnas y así hacer más legibles los conjuntos de resultados.

```
USE northwind
SELECT  firstname AS First, lastname AS Last
        ,employeeid AS 'Employee ID:'
FROM employees
GO
```



First	Last	Employee ID:
Nancy	Davolio	1
Andrew	Fuller	2
Janet	Leverling	3
Margaret	Peacock	4
Steven	Buchanan	5
Michael	Suyama	6
Robert	King	7
Laura	Callahan	8
Anne	Dodsworth	9

Si desea crear nombres de columnas más legibles, puede utilizar la palabra clave AS para reemplazar los nombres predeterminados por alias en la lista de selección.

Sintaxis parcial

```
SELECT nombreColumna | expresión AS títuloColumna
FROM nombreTabla
```

Al cambiar los nombres de las columnas, tenga en cuenta los siguientes hechos e instrucciones:

- De forma predeterminada, en el conjunto de resultados se muestran los nombres de columna designados en la instrucción CREATE TABLE.
- Utilice comillas simples para los nombres de columna que contengan espacios o que no se ajusten a las convenciones de denominación de objetos de SQL Server.
- Puede crear alias de columna para columnas calculadas que contengan funciones o literales de cadena.
- Cada alias de columna puede contener hasta 128 caracteres.

Ejemplo

En este ejemplo se obtiene una lista de empleados de la tabla **employees**. Los alias de columna especificados reemplazan a los nombres de columna **firstname**, **lastname** y **employeeid**. Observe que el alias Employee ID: aparece entre comillas simples porque contiene un espacio.

```
USE northwind
SELECT  firstname AS First, lastname AS Last
        ,employeeid AS 'Employee ID:'
FROM employees
GO
```

Resultado

First	Last	Employee ID:
Nancy	Davolio	1
Andrew	Fuller	2
Janet	Leverling	3
Margaret	Peacock	4
Steven	Buchanan	5
Michael	Suyama	6
Robert	King	7
Laura	Callahan	8
Anne	Dodsworth	9

(9 filas afectadas)

Uso de literales


Objetivos de la diapositiva

Mostrar cómo el uso de literales en la instrucción SELECT hace más legibles los conjuntos de resultados.

Explicación previa

Los valores de los conjuntos de resultados pueden ser más legibles si utiliza literales.

```
USE northwind
SELECT  firstname, lastname
        , 'Número de identificación:', employeeid
FROM employees
GO
```



<i>First</i>	<i>Last</i>	<i>Employee ID:</i>
Nancy	Davolio	Número de identificación: 1
Andrew	Fuller	Número de identificación: 2
Janet	Leverling	Número de identificación: 3
Margaret	Peacock	Número de identificación: 4
Steven	Buchanan	Número de identificación: 5
Michael	Suyama	Número de identificación: 6
Robert	King	Número de identificación: 7
Laura	Callahan	Número de identificación: 8
Anne	Dodsworth	Número de identificación: 9

Los literales son letras, números o símbolos que se utilizan como valores específicos en un conjunto de resultados. Puede incluirlos en la lista de selección para hacer más legible el resultado.

Sintaxis parcial

```
SELECT nombreColumna | 'literalCadena' [, nombreColumna |
'literalCadena'...]
FROM nombreTabla
```

Ejemplo

En este ejemplo se obtiene una lista de empleados de la tabla **employees**. Observe que la cadena Número de identificación: precede a la columna **employeeid** en el conjunto de resultados.

```
USE northwind
SELECT  firstname, lastname
        , 'Número de identificación:', employeeid
FROM employees
GO
```

Resultado

firstname	lastname	employeeid
Nancy	Davolio	Número de identificación: 1
Andrew	Fuller	Número de identificación: 2
Janet	Leverling	Número de identificación: 3
Margaret	Peacock	Número de identificación: 4
Steven	Buchanan	Número de identificación: 5
Michael	Suyama	Número de identificación: 6
Robert	King	Número de identificación: 7
Laura	Callahan	Número de identificación: 8
Anne	Dodsworth	Número de identificación: 9

(9 filas afectadas)

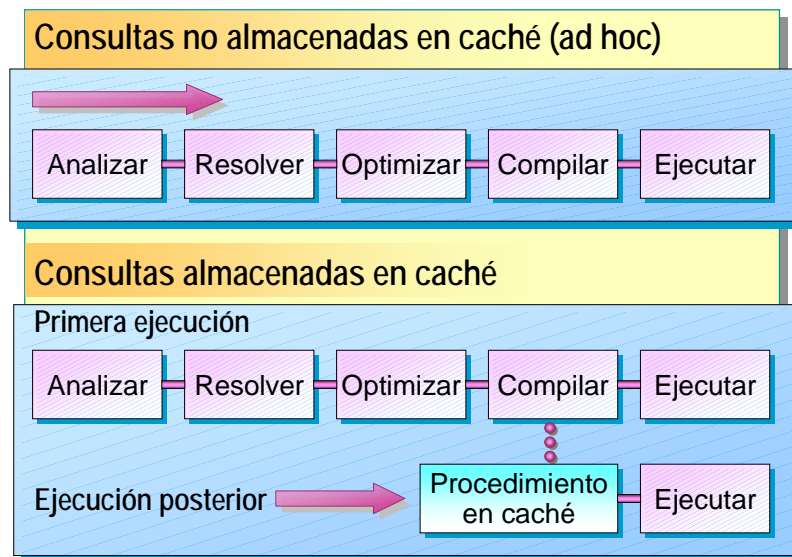
◆ Cómo se procesan las consultas

Objetivos de la diapositiva

Describir cómo se procesan las consultas.

Explicación previa

Todas las consultas siguen el mismo proceso antes de ejecutarse.



Todas las consultas siguen el mismo proceso antes de ejecutarse. SQL Server puede almacenar alguno de los procesos para una ejecución posterior de la misma consulta.

Consultas que no están almacenadas en caché (ad hoc)

Todas las consultas se analizan, resuelven, optimizan y compilan antes de ejecutarse.

Las vistas se reducen (resuelven) a una única instrucción. La instrucción que hace referencia a la vista se mezcla con la definición de la vista y crea una instrucción SELECT.

Proceso	Descripción
Analizar	Comprueba la precisión de la sintaxis.
Resolver	Comprueba que existen los nombres de los objetos; determina el permiso del propietario del objeto.
Optimizar	Determina los índices que se deben utilizar y las estrategias de combinación.
Compilar	Convierte la consulta en un formato ejecutable.
Ejecutar	Emite las solicitudes compiladas para procesarlas.

Consultas almacenadas en caché

Para mejorar el rendimiento, SQL Server puede guardar los planes de consultas compilados para reutilizarlos. Los planes de consultas son instrucciones optimizadas acerca de cómo procesar las consultas y tener acceso a los datos. Se almacenan en la *caché de procedimientos*, una ubicación de almacenamiento temporal de la versión que se esté ejecutando actualmente de una consulta específica.

Consideraciones acerca del rendimiento

Objetivos de la diapositiva

Tratar las consideraciones acerca del rendimiento al realizar consultas básicas.

Explicación previa

Al efectuar consultas básicas debe tener en cuenta algunos de los aspectos que afectan al rendimiento de SQL Server.

- Las condiciones de búsqueda negativas pueden hacer que la recuperación de datos sea más lenta
- La obtención de datos con LIKE puede resultar más lenta
- Las coincidencias exactas o intervalos hacen que la obtención de datos sea más rápida
- La cláusula ORDER BY puede ralentizar la obtención de datos

Al realizar consultas básicas debe tener en cuenta algunos de los aspectos que afectan al rendimiento de SQL Server:

- Utilice condiciones de búsqueda positivas, en lugar de negativas. Las condiciones de búsqueda negativas, como NOT BETWEEN, NOT IN e IS NOT NULL, pueden hacer que la recuperación de datos sea más lenta, ya que con ellas se evalúan todas las filas.
- Evite el uso de la condición de búsqueda LIKE si puede escribir una consulta más específica. La obtención de datos con LIKE puede resultar más lenta.
- Utilice coincidencias exactas o intervalos como condiciones de búsqueda siempre que sea posible. El rendimiento de las consultas específicas es mayor.
- La obtención de datos puede ralentizarse cuando se utiliza la cláusula ORDER BY, ya que SQL Server debe determinar y ordenar el conjunto de resultados antes de devolver la primera fila.