

Restricción y Clasificación de los Datos

Objetivos

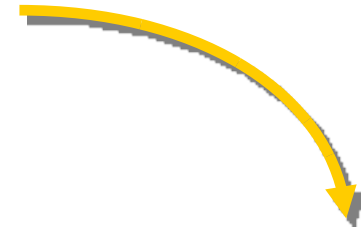
- Al completar esta lección, debería ser capaz de hacer lo siguiente:
 - Limitar los registros recuperados por una consulta
 - Clasificar los registros recuperados por una consulta

Limitar Filas por medio de Selecciones

EMP

EMPNO	ENAME	JOB	...	DEPTNO
7839	KING	PRESIDENT		10
7698	BLAKE	MANAGER		30
7782	CLARK	MANAGER		10
7566	JONES	MANAGER		20
...				

“...recupera todos los empleados del departamento 10”



EMP

EMPNO	ENAME	JOB	...	DEPTNO
7839	KING	PRESIDENT		10
7782	CLARK	MANAGER		10
7934	MILLER	CLERK		10

Limitar las filas seleccionadas

- Restringir las filas recuperadas usando la cláusula WHERE.

```
SELECT          [DISTINCT] {*, column [alias], ...}  
FROM            table  
[WHERE          condition(s)];
```

- La cláusula WHERE se escribe después de la cláusula FROM.

Uso de la Cláusula WHERE

```
SQL> SELECT ename, job, deptno  
2 FROM emp  
3 WHERE job='CLERK';
```

ENAME	JOB	DEPTNO
-----	-----	-----
JAMES	CLERK	30
SMITH	CLERK	20
ADAMS	CLERK	20
MILLER	CLERK	10

Cadenas de Caracteres y Fechas

- Las cadenas de caracteres o las fechas se encierran entre comillas simples.
- Los valores de tipo caracter son sensibles a mayúsculas/minúsculas y los valores de tipo fecha son sensibles al formato.
- El formato de fecha por defecto es 'DD-MON-YY'.

```
SQL> SELECT  ename, job, deptno  
2 FROM      emp  
3 WHERE     ename = 'JAMES' ;
```

Operadores de Comparación

Operador	Significado
=	Igual que
>	Mayor que
>=	Mayor que o igual a
<	Menor que
<=	Menor que o igual a
<> ó !=	Diferente de

Uso de Operadores de Comparación

```
SQL> SELECT ename, sal, comm  
2 FROM emp  
3 WHERE sal<=comm;
```

ENAME	SAL	COMM
MARTIN	1250	1400

Otros Operadores de Comparación


Operador	Significado
BETWEEN ...AND...	Entre dos valores (inclusive)
IN(list)	Lista de valores
LIKE	Se ajusta a un patrón
IS NULL	Es un valor nulo

Uso del Operador BETWEEN

- Use el operador BETWEEN para visualizar registros basados en un rango de valores.

```
SQL> SELECT  ename, sal
      2  FROM    emp
      3  WHERE   sal BETWEEN 1000 AND 1500;
```

ENAME	SAL		
MARTIN	1250	Lower limit	Higher limit
TURNER	1500		
WARD	1250		
ADAMS	1100		
MILLER	1300		



Using the IN Operator

- Use el operador IN para localizar valores coincidentes con una lista de valores.

```
SQL> SELECT empno, ename, sal, mgr
2 FROM emp
3 WHERE mgr IN (7902, 7566, 7788);
```

EMPNO	ENAME	SAL	MGR
7902	FORD	3000	7566
7369	SMITH	800	7902
7788	SCOTT	3000	7566
7876	ADAMS	1100	7788

Uso del Operador LIKE

- Usar el operador LIKE para ejecutar búsquedas en cadenas de caracteres que incluyan comodines.
- Las condiciones de búsqueda pueden contener caracteres o números.
 - "%" denota cero o varios caracteres.
 - "_" denota un solo caracter.

```
SQL> SELECT   ename
2  FROM      emp
3  WHERE     ename LIKE 'S%';
```

Uso del Operador LIKE

- Se puede combinar patrones de caracteres

```
SQL> SELECT      ename  
      2 FROM      emp  
      3 WHERE     ename LIKE ' _A% ' ;
```

ENAME

JAMES

WARD

- Se puede usar el identificador ESCAPE
para buscar "%" o "_"

Uso del Operador IS NULL

- Verificar la presencia de valores nulos con el operador IS NULL.

```
SQL> SELECT  ename, mgr  
2  FROM      emp  
3  WHERE     mgr IS NULL;
```

ENAME	MGR
-----	-----
KING	

Operadores Lógicos

Operador	Significado
AND	Devuelve TRUE si ambas condiciones son TRUE
OR	Devuelve TRUE si alguna de las condiciones es TRUE
NOT	Devuelve TRUE si la siguiente condición es FALSE

Uso del Operador AND

AND requiere que ambas condiciones sean TRUE

```
SQL> SELECT empno, ename, job, sal
2   FROM emp
3  WHERE sal >= 1100
4  AND   job = 'CLERK';
```

EMPNO	ENAME	JOB	SAL
7876	ADAMS	CLERK	1100
7934	MILLER	CLERK	1300

Uso del Operador OR

OR requiere que al menos una condición sea TRUE

```
SQL> SELECT empno, ename, job, sal
2 FROM emp
3 WHERE sal >= 1100
4 OR job = 'CLERK';
```

EMPNO	ENAME	JOB	SAL
7839	KING	PRESIDENT	5000
7698	BLAKE	MANAGER	2850
7782	CLARK	MANAGER	2450
7566	JONES	MANAGER	2975
7654	MARTIN	SALESMAN	1250

...

14 rows selected.

Uso del Operador NOT

```
SQL> SELECT ename, job
      2 FROM emp
      3 WHERE job NOT IN ('CLERK', 'MANAGER', 'ANALYST');
```

ENAME	JOB
-----	-----
KING	PRESIDENT
MARTIN	SALESMAN
ALLEN	SALESMAN
TURNER	SALESMAN
WARD	SALESMAN

Reglas de Precedencia

Orden Evaluado	Operador
1	Todos los Operadores de Comparación
2	NOT
3	AND
4	OR

- Utilice paréntesis para modificar las reglas de precedencia.


Reglas de Precedencia

```
SQL> SELECT ename, job, sal
      2  FROM    emp
      3  WHERE   job='SALESMAN'
      4  OR      job='PRESIDENT'
      5  AND     sal>1500;
```

ENAME	JOB	SAL
KING	PRESIDENT	5000
MARTIN	SALESMAN	1250
ALLEN	SALESMAN	1600
TURNER	SALESMAN	1500
WARD	SALESMAN	1250

Reglas de Precedencia

Uso de paréntesis para forzar prioridad.

```
SQL> SELECT      ename, job, sal
  2  FROM          emp
  3  WHERE  (job='SALESMAN'
  4  OR            job='PRESIDENT')
  5  AND          sal>1500;
```

ENAME	JOB	SAL
KING	PRESIDENT	5000
ALLEN	SALESMAN	1600

La Cláusula ORDER BY

- Ordenar filas con la cláusula ORDER BY.
- ASC – orden ascendente, constituye la opción por defecto.
- DESC – orden descendente.
- La cláusula ORDER BY siempre va al final del SELECT

```
SQL> SELECT      ename, job, deptno, hiredate
  2  FROM          emp
  3  ORDER BY hiredate;
```

ENAME	JOB	DEPTNO	HIREDATE
SMITH	CLERK	20	17-DEC-80
ALLEN	SALESMAN	30	20-FEB-81
...			

14 rows selected.

Clasificación en Orden Descendente

```
SQL> SELECT      ename, job, deptno, hiredate  
2  FROM          emp  
3  ORDER BY hiredate DESC;
```

ENAME	JOB	DEPTNO	HIREDATE
-----	-----	-----	-----
ADAMS	CLERK	20	12-JAN-83
SCOTT	ANALYST	20	09-DEC-82
MILLER	CLERK	10	23-JAN-82
JAMES	CLERK	30	03-DEC-81
FORD	ANALYST	20	03-DEC-81
KING	PRESIDENT	10	17-NOV-81
MARTIN	SALESMAN	30	28-SEP-81
...			

14 rows selected.

Clasificación por alias de campo (columna)

```
SQL> SELECT    empno, ename, sal*12 annsal  
2  FROM      emp  
3  ORDER BY  annsal;
```

EMPNO	ENAME	ANNSAL
7369	SMITH	9600
7900	JAMES	11400
7876	ADAMS	13200
7654	MARTIN	15000
7521	WARD	15000
7934	MILLER	15600
7844	TURNER	18000

...

14 rows selected.

Clasificación por Múltiples Columnas

- La lista de ORDER BY es el orden de clasificación

```
SQL> SELECT  ename, deptno, sal
  2  FROM    emp
  3  ORDER BY deptno, sal DESC;
```

ENAME	DEPTNO	SAL
KING	10	5000
CLARK	10	2450
MILLER	10	1300
FORD	20	3000
...		

14 rows selected.

- Puede ordenar por una columna no seleccionada.

Clasificación por Múltiples Columnas

- La lista de ORDER BY es el orden de clasificación

```
SQL> SELECT  ename, deptno, sal  
2  FROM      emp  
3  ORDER BY  2, 3 DESC;
```

ENAME	DEPTNO	SAL
KING	10	5000
CLARK	10	2450
MILLER	10	1300
FORD	20	3000
...		

14 rows selected.

Resumen

SELECT	[DISTINCT] {*, <i>columna</i> [<i>alias</i>], ...}
FROM	<i>tabla</i>
[WHERE	<i>condicion(es)</i>
[ORDER BY	{<i>columna</i>, <i>expresion</i>, <i>alias</i>} [ASC DESC]];

Visión General de la Práctica

- Seleccionar datos y cambiar el orden de las filas mostradas.
- Restringir las filas usando la cláusula `WHERE`.
- Usar el método de alias de campo con comillas dobles.