

## Parte 3

# Condicionando y Ordenando datos

# Objetivos

- Al completar esta lección, deberá ser capaz de hacer lo siguiente :
  - Limitar los renglones recuperados por una consulta
  - Ordenar los renglones recuperados por una consulta

# Limitando los Renglones utilizando una Selección

## 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

## Limitando Renglones

- Para restringir los renglones utilice la cláusula **WHERE**.

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

- La cláusula **WHERE** se coloca después de la cláusula **FROM**.

## Utilizando la cláusula WHERE

```
SELECT ename, job, deptno  
FROM emp  
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 y fechas se encierran entre comillas simples
- Las cadenas de caracteres son sensibles a las mayúsculas

```
SELECT    ename, job, deptno
FROM      emp
WHERE     ename = 'JAMES';
```

# Operadores de Comparación

Operador	Descripción
=	Igual
>	Mayor que
>=	Mayor o igual que
<	Menor que
<=	Menor o igual que
<>	Diferente

## Utilizando los Operadores de Comparación

```
SELECT ename, sal, comm  
FROM emp  
WHERE sal<=comm;
```

ename	sal	comm
MARTIN	1250	1400



## Otros Operadores de Comparación

Operador	Descripción
<b>BETWEEN ...AND...</b>	Entre dos valores (inclusive)
<b>IN(list)</b>	Coincida con uno de la lista
<b>LIKE</b>	Igual al patrón de caracteres
<b>IS NULL</b>	Es un valor null

## Utilizando el Operador BETWEEN

- Utilice el operador **BETWEEN** para condicionar renglones basados en rangos de valores.

```
SELECT  ename, sal
FROM    emp
WHERE   sal BETWEEN 1000 AND 1500;
```

ename	sal	Límite inferior	Límite superior
MARTIN	1250		
TURNER	1500		
WARD	1250		
ADAMS	1100		
MILLER	1300		

## Utilizando el Operador IN

- Utilice el operador IN para igualar valores dentro de la lista.

```
SELECT empno, ename, sal, mgr
FROM emp
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

## Utilizando el Operador LIKE

- Utilice el operador LIKE para realizar búsquedas con caracteres comodines.
- Las condiciones de búsqueda pueden contener tanto caracteres como números.
  - % indica cero o más caracteres
  - \_ indica un caracter

```
SELECT  ename  
FROM    emp  
WHERE   ename LIKE 'S%';
```

## Utilizando el Operador LIKE

- Puede combinar el patrón de caracteres como se muestra en el ejemplo :

```
SELECT  ename  
FROM    emp  
WHERE   ename LIKE '_A%';
```

```
ENAME  
-----  
JAMES  
WARD  
MARTIN
```

## Utilizando el Operador IS NULL

- **Verifique los valores nulos con el operador IS NULL**

```
SELECT  ename, mgr
FROM    emp
WHERE   mgr IS NULL;
```

ename	mgr
-----	-----
KING	NULL

# Operadores Lógicos

Operador	Descripción
AND	Devuelve TRUE si <i>ambos</i> componentes son TRUE
OR	Devuelve TRUE si <i>al menos uno</i> de los componentes es TRUE
NOT	Devuelve el valor <i>negado</i> según el valor de la expresión evaluada

## Utilizando el operador AND

**AND requiere que ambas condiciones sean TRUE**

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

empno	ename	job	sal
7876	ADAMS	CLERK	1100
7934	MILLER	CLERK	1300



## Utilizando el operador OR

**OR requiere que al menos una condición sea TRUE.**

```
SELECT empno, ename, job, sal
FROM emp
WHERE sal >= 2000
OR      job = 'CLERK';
```

empno	ename	job	sal
7839	KING	PRESIDENT	5000
7698	BLAKE	MANAGER	2850
7782	CLARK	MANAGER	2450
7566	JONES	MANAGER	2975
...			

(10 row(s) affected)

## Utilizando el operador NOT

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

ename	job
-----	-----
KING	PRESIDENT
MARTIN	SALESMAN
ALLEN	SALESMAN
TURNER	SALESMAN
WARD	SALESMAN


## Reglas de Precedencia

Orden	Operador
1	Todos los operadores de comparación
2	NOT
3	AND
4	OR

➤ **Utilice paréntesis para modificar el orden de evaluación**

## Reglas de Precedencia

```
SELECT ename, job, sal
FROM emp
WHERE job='SALESMAN'
OR job='PRESIDENT'
AND sal>1500;
```




ename	job	sal
KING	PRESIDENT	5000
MARTIN	SALESMAN	1250
ALLEN	SALESMAN	1600
TURNER	SALESMAN	1500
WARD	SALESMAN	1250

# Reglas de Precedencia

Usando paréntesis para forzar la prioridad.

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



ename	job	sal
-----	-----	-----
KING	PRESIDENT	5000
ALLEN	SALESMAN	1600

## Cláusula ORDER BY

- Ordene la salida con la cláusula ORDER BY
  - ASC: orden ascendente, default
  - DESC: orden descendente
- La cláusula ORDER BY siempre debe ser la última en una instrucción SELECT.

```
SELECT    ename, job, deptno, hiredate
FROM      emp
ORDER BY  hiredate;
```

ename	job	deptno	hiredate
SMITH	CLERK	20	1980-12-17 00:00:00.000
ALLEN	SALESMAN	30	1981-02-20 00:00:00.000
...			

(14 row(s) affected)

## Ordenando en forma descendente

```
SELECT      ename, job, deptno, hiredate
FROM        emp
ORDER BY    hiredate DESC;
```

ename	job	deptno	hiredate
ADAMS	CLERK	20	1983-01-12 00:00:00.000
SCOTT	ANALYST	20	1982-12-09 00:00:00.000
...			
(14 row(s) affected)			

## Ordenando con Alias

```
SELECT empno, ename, sal*12 ann_sal  
FROM emp  
ORDER BY ann_sal;
```

empno	ename	ann_sal
7369	SMITH	9600
7900	JAMES	11400
7876	ADAMS	13200
7654	MARTIN	15000
7521	WARD	15000
7934	MILLER	15600
7844	TURNER	18000
...		

(14 row(s) affected)



## Ordenando con Múltiples Columnas

- Utilice varias columnas para establecer diferentes criterios de ordenamiento.

```
SELECT  ename, deptno, sal
FROM    emp
ORDER BY deptno, sal DESC;
```

ename	deptno	sal
-----	-----	-----
KING	10	5000
CLARK	10	2450
MILLER	10	1300
FORD	20	3000
...		
(14 row(s) affected)		

- Puede ordenar por columnas que no aparezcan en la consulta

# Práctica

- **Seleccionar datos y cambiar el orden en que son mostrados**
- **Elimine renglones utilizando WHERE**
- **Utilice diferentes criterios de ordenamiento**
- **Utilice alias en ORDER BY**