

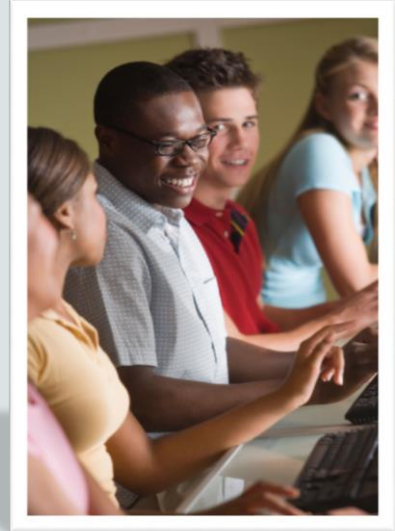


 **ACADEMY**

# Programación de bases de datos con SQL

15-3

Gestión de Vistas



**ORACLE** ACADEMY

Copyright © 2017, Oracle y/o sus filiales. Todos los derechos reservados.

# Objetivos

En esta lección se abordan los siguientes objetivos:

- Crear y ejecutar una sentencia SQL que elimina una vista
- Crear y ejecutar una consulta mediante una vista en línea
- Crear y ejecutar una consulta para el análisis de N principales

# Objetivo

- Aprender a crear y sustituir vistas no sería completo a menos que también supiera cómo eliminarlas.
- Las vistas se crean con fines específicos.
- Cuando la vista ya no es necesaria o se debe modificar, existen medios para realizar los cambios necesarios.
- Si un empleado que ha tenido acceso a información financiera deja la compañía, es probable que no desee que su vista siga estando accesible.
- En esta lección, aprenderá a suprimir una vista, crear una vista en línea y construir una sentencia SELECT para producir una lista ordenada de datos.

# Supresión de una Vista

- Puesto que una vista no contiene datos propios, su eliminación no afecta a los datos de las tablas subyacentes.
- Si la vista se ha utilizado para INSERTAR, ACTUALIZAR o SUPRIMIR datos en el pasado, esos cambios en las tablas base se mantienen.
- La supresión de una vista simplemente elimina la definición de vista de la base de datos.



# Supresión de una Vista

- Recuerde que las vistas se almacenan como una sentencia SELECT en el diccionario de datos.
- Solo el creador o usuarios con el privilegio DROP ANY VIEW pueden eliminar una vista.
- La sintaxis SQL para eliminar una vista es:

```
DROP VIEW viewname;
```

# Vistas en Línea

- Las vistas en línea también se denominan subconsultas en la cláusula FROM.
- Inserte una subconsulta en la cláusula FROM como si la subconsulta fuera un nombre de tabla.
- Las vistas en línea se utilizan normalmente para simplificar las complejas consultas mediante la eliminación de operaciones de unión y la condensación de varias consultas en una sola.

# Vistas en Línea

- Como se muestra en el siguiente ejemplo, la cláusula FROM contiene una sentencia SELECT que recupera datos como cualquier sentencia SELECT.
- A los datos devueltos por la subconsulta se les asigna un alias (d), que, a continuación, se utiliza junto con la consulta principal para devolver columnas seleccionadas de ambos orígenes de consulta.

```
SELECT e.last_name, e.salary, e.department_id, d.maxsal
FROM employees e,
     ( SELECT department_id, max(salary) maxsal
       FROM employees
       GROUP BY department_id ) d
WHERE e.department_id = d.department_id
AND e.salary = d.maxsal;
```

Esta vista en línea busca el salario más alto para cada departamento y, a continuación, la consulta muestra el nombre del empleado con ese salario.

Una vista en línea debe tener un alias (en este ejemplo, "d") ya que funciona como un nombre de tabla en la cláusula FROM y "SELECT department\_id, max(salary) ..." no es un nombre de tabla válido.



# ANÁLISIS DE N PRINCIPALES

- El análisis de N principales es una operación SQL utilizada para clasificar resultados.
- El uso del análisis de N principales resulta útil cuando desea recuperar los 5 registros principales o los n registros principales de un juego de resultados devuelto por una consulta.

```
SELECT ROWNUM AS "Longest employed", last_name, hire_date  
FROM employees  
WHERE ROWNUM <=5  
ORDER BY hire_date;
```

Longest employed	LAST_NAME	HIRE_DATE
1	King	17-Jun-1987
4	Whalen	17-Sep-1987
2	Kochhar	21-Sep-1989
3	De Haan	13-Jan-1993
5	Higgins	07-Jun-1994

Sin embargo, los resultados de esta consulta no son lo que espera. El motivo es que la cláusula ORDER BY ejecute siempre se ejecuta la última, por lo que las filas se ordenan después de que se les asigne un número. La siguiente diapositiva muestra cómo resolver este problema.

# ANÁLISIS DE N PRINCIPALES

- El análisis de N principales utiliza una vista en línea (subconsulta) para devolver un juego de resultados.
- Puede utilizar ROWNUM en su consulta para asignar un número de fila al juego de resultados.
- A continuación, la consulta principal utiliza ROWNUM para ordenar los datos y devolver los cinco principales.

```
SELECT ROWNUM AS "Longest employed", last_name, hire_date
FROM (SELECT last_name, hire_date
      FROM employees
      ORDER BY hire_date)
WHERE ROWNUM <=5;
```

Mediante una vista en línea, la cláusula ORDER BY se realiza antes de que se agregue ROWNUM.

# ANÁLISIS DE N PRINCIPALES

Longest employed	LAST_NAME	HIRE_DATE
1	King	17-Jun-1987
2	Whalen	17-Sep-1987
3	Kochhar	21-Sep-1989
4	Hunold	03-Jan-1990
5	Ernst	21-May-1991

- En el ejemplo anterior, la vista en línea selecciona primero la lista de last\_names y hire\_dates de los empleados:

```
(SELECT last_name, hire_date FROM employees
```

- A continuación, la vista en línea ordena los años del más antiguo al más reciente.

```
(ORDER BY hire_date
```

# ANÁLISIS DE N PRINCIPALES

- La cláusula WHERE de la consulta externa se utiliza para restringir el número de filas devueltas y debe utilizar un operador < o <=.

```
SELECT ROWNUM AS "Longest employed", last_name, hire_date
FROM (SELECT last_name, hire_date
      FROM employees
      ORDER BY hire_date)
WHERE ROWNUM <=5;
```

Longest employed	LAST_NAME	HIRE_DATE
1	King	17-Jun-1987
2	Whalen	17-Sep-1987
3	Kochhar	21-Sep-1989
4	Hunold	03-Jan-1990
5	Ernst	21-May-1991

# Terminología

Entre los términos clave utilizados en esta lección se incluyen:

- DROP VIEW
- VISTA EN LÍNEA
- ANÁLISIS DE N PRINCIPALES
- ROWNUM

# Resumen

En esta lección, debe haber aprendido lo siguiente:

- Crear y ejecutar una sentencia SQL que elimina una vista
- Crear y ejecutar una consulta mediante una vista en línea
- Crear y ejecutar una consulta para el análisis de N principales



 **ACADEMY**