



8.1 Introducción

El uso más común de SQL*Plus es realizar consultas y obtener informes. Hasta el momento, los listados obtenidos al hacer las consultas presentaban los datos tan sólo con las cabeceras de las columnas y, a continuación, con los datos de las columnas de cada fila consultada. No obstante, estos datos se pueden presentar de forma más elegante, añadiendo títulos o formatos a las columnas, líneas de totales, etcétera. En esta unidad, se explica cómo se puede utilizar SQL*Plus para crear listados formateados, controlando los títulos del listado, las cabeceras de las columnas, las líneas de totales y sub-totales, el formato de las columnas numéricas, etcétera.

Es posible escribir los informes directamente desde el indicador de SQL*PLUS; de este modo, cuando se ejecute una consulta se producirá la salida formateada automáticamente. Pero esto tiene un inconveniente: si, al cabo de unos días, se desea obtener el mismo informe, se tendrían que teclear las órdenes de nuevo. Para evitar esta situación guardamos todas las órdenes en un fichero, que ejecutaremos cada vez que necesitemos consultar el listado.

8.2 Uso del editor para crear archivos SQL

Para guardar las órdenes en un archivo hay que teclear, desde el indicador de SQL* Plus, la orden EDIT seguida del nombre del archivo en el que se van a almacenar esas órdenes. Por ejemplo: SQL> EDIT LISTADO. El archivo se crea con la extensión SQL.

Con EDIT se invoca al editor (por defecto, se toma el Bloc de notas: NOTEPAD.EXE). Éste avisa de que no se encuentra el archivo LISTADO.SQL y pregunta si se desea crear. Se pulsará Sí (Figura 8.1).

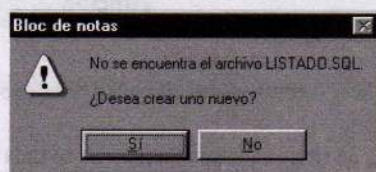


Fig. 8.1. Invocamos al editor.

Aparecerá entonces la pantalla del Bloc de notas en blanco para empezar a escribir las órdenes SQL* Plus. Por defecto, los archivos se almacenan en el directorio BIN de Oracle.

Para que el archivo se guarde en otro directorio hay que poner delante del nombre de archivo todo el trayecto. Así, por ejemplo, si queremos guardar LISTADO en el directorio SQL de la unidad D tenemos que poner:

```
SQL> EDIT D:\SQL\LISTADO
```




8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

Para elegir otro editor se utiliza la opción del menú: *Editar/Editor/Definir Editor* (Figura 8.2).

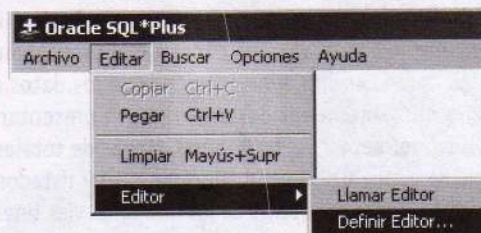


Fig. 8.2. Elegimos otro editor

Para ejecutar las órdenes contenidas en el archivo hemos de recurrir a la sentencia **START** seguida del nombre del archivo.

Es decir, después de escribir las órdenes que realizan el listado, se ejecutan tecleando lo siguiente desde el indicador **SQL*Plus**: **SQL> START LISTADO.SQL** o **SQL> START LISTADO**.

Si el archivo está en otro directorio, se teclea todo el trayecto: **SQL> START D:\SQL\LISTADO**.

8.3 Órdenes para generar informes

La Tabla 8.1 muestra las órdenes básicas que utilizaremos para generar informes.

Orden	Función
REMARK	Se usa para incluir comentarios.
SET HEADSEP	Identifica el carácter que le dice a SQL*Plus que divida un título en dos o más líneas.
TTITLE	Presenta un título en la página superior de cada página del informe.
BTITLE	Presenta un título en la página inferior de cada página del informe.
COLUMN	Se utiliza para dar formato a la columna.
BREAK ON	Sirve para producir una ruptura de control, y calcular totales y subtotales.
COMPUTE	Realiza cálculos con grupos de filas.
SET LINESIZE	Define el número máximo de caracteres por línea.
SET PAGESIZE	Define el número máximo de líneas por página.
SET NEWPAGE	Define el número de líneas de blanco entre páginas.
SPOOL	Guarda la salida de un informe en un archivo.

Tabla 8.1. Órdenes SQL*Plus

Veamos, mediante un caso práctico, cómo utilizar estas órdenes SQL*Plus. Después profundizaremos más sobre cada una.

8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



(Continúa)

(Continúa)

Caso práctico



- Partimos de las tablas TARTICULOS, que contiene datos de los artículos, y TVENTAS que contiene datos de las ventas de los artículos. Se quiere obtener un listado en el que, por cada artículo, hay que imprimir todas las ventas que han tenido lugar durante el mes de mayo del año 2005, calculando por artículo la suma de unidades y la suma de importes. El listado que se ha de obtener es el siguiente:

VENTAS POR ARTÍCULOS DURANTE MAYO			
Artículo vendido	Fecha Venta	Unidades vendidas	IMPORTE
XXXXXXXXXXXXXXXXXX	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
		XXXXXX	XXXXXXXXXX
XXXXXXXXXXXXXXXXXX	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
	XXXXXXXXXX	XXXXXX	XXXXXXXXXX
		XXXXXX	XXXXXXXXXX

Sólo Mayo de 2005

Para obtener el listado escribiremos las siguientes órdenes SQL*Plus en el archivo creado anteriormente (SQL> EDIT LISTADO):

```

REM Nombre: LISTADO.SQL
REM Descripción: Informe de ventas por artículos durante el mes de Mayo
SET HEADSEP |

TTITLE 'VENTAS POR ARTÍCULOS DURANTE MAYO|=====
BTITLE LEFT 'Sólo Mayo de 2005'
COLUMN DENOM HEADING 'Artículo|Vendido'
COLUMN DENOM FORMAT A15
COLUMN DENOM TRUNCATED

COLUMN FECHA_VENTA HEADING 'Fecha|Venta'
COLUMN UNI_VEND HEADING 'Unidades|Vendidas' FORMAT 9G999
COLUMN IMPORTE FORMAT 9G999D99
    
```

(Continúa)

(Continúa)



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

(Continuación)

```
BREAK ON DENOM SKIP 2
COMPUTE SUM OF UNI_VEND ON DENOM
COMPUTE SUM OF IMPORTE ON DENOM
SET LINESIZE 50
SET PAGESIZE 50
SET NEWPAGE 0

SPOOL LISTADO.LST

SELECT DENOM, FECHA_VENTA, UNI_VEND, UNI_VEND * PVP "IMPORTE"
FROM TARTICULOS A, TVENTAS V
WHERE A.CODIGO = V.CODIGO
AND FECHA_VENTA BETWEEN '01/05/2005' AND '31/05/2005'
ORDER BY DENOM, FECHA_VENTA;

TTITLE OFF
BTITLE OFF

SPOOL OFF
```

En este listado, las sentencias SQL acaban en punto y coma. Esto las diferencia de las órdenes SQL*Plus. En el listado, la única sentencia SQL es SELECT. Las restantes sentencias son SQL*Plus.

Una vez escrito, guardamos el archivo LISTADO.SQL y, a continuación, desde el indicador SQL*Plus ejecutamos el archivo: SQL> START LISTADO.SQL. El resultado es el siguiente:

Lun Jun 20

página 1

VENTAS POR ARTÍCULOS DURANTE MAYO

Artículo vendido	Fecha venta	Unidades Vendidas	IMPORTE
CEBOLLAS	10/05/2005	5	4,60
	28/05/2005	6	5,52
	30/05/2005	14	12,88

sum		25	23,00
GARBANZOS	07/05/2005	11	24,20
	10/05/2005	5	11,00

sum		16	35,20

8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



(Continuación)

JUDÍAS BLANCAS	21/05/2005	20	46,00
	31/05/2005	54	124,20
*****		-----	-----
sum		74	170,20
LENTEJAS	28/05/2005	6	11,40
	29/05/2005	18	34,20
	30/05/2005	14	26,60
	31/05/2005	18	34,20
*****		-----	-----
sum		56	106,40
NARANJAS	19/05/2005	10	12,00
	26/05/2005	10	12,00
*****		-----	-----
sum		20	24,00
PATATAS	25/05/2005	15	10,50
	27/05/2005	16	11,20
*****		-----	-----
sum		31	21,70
Sólo Mayo de 2005			
15 filas seleccionadas.			

A. Orden REMARK (REM)

Es la primera orden que aparece en el listado anterior. Permite añadir comentarios. Se puede colocar una orden REM en cualquier parte del listado.

REM Nombre: LISTADO.SQL

REM Descripción: Informe de ventas por artículos durante el mes de Mayo

B. Configuración de variables del entorno SQL*Plus con la orden SET

Se puede consultar y cambiar el valor de las variables del entorno SQL*Plus accediendo a ellas mediante la opción del menú: *Opciones/Entorno*. También es posible dar valor a estas variables usando la orden SET desde el indicador de SQL*Plus:

SET variable valor



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

Y visualizar el valor de las mismas utilizando la orden **SHOW**: **SHOW** variable
Por ejemplo:

```
SQL> SET HEADSEP '-'
SQL> SHOW HEADSEP
headsep "-" (hex 2d)
```

En el listado se usan las variables **HEADSEP**, **LINESIZE**, **PAGESIZE** y **NEWPAGE**. SQL*Plus dispone de más de 50 variables del entorno. Explicamos algunas de estas variables:

- **SET HEADSEP |**. El símbolo que se coloca a la derecha de la orden **SET HEADSEP** identifica al carácter que dividirá un título superior o inferior, o una cabecera de columna en dos o más líneas. En el ejemplo se usa el carácter «|» para separar el título o la cabecera que vaya a ocupar más de una línea.
- **SET LINESIZE**. Con esta orden se define el número máximo de caracteres que pueden aparecer en una línea. En nuestro listado hemos definido 50 caracteres por línea: **SET LINESIZE 50**.
- **SET PAGESIZE**. Con esta orden se define el número de líneas por cada página, incluyendo **TTITLE** y **BTITLE**, las cabeceras de columnas y cualquier línea en blanco. Normalmente, este número es 66. En el listado se han definido 50 líneas por página: **SET PAGESIZE 50**.
- **SET NEWPAGE**. Esta orden imprime líneas en blanco antes de la línea superior de cada página del informe, donde se encuentra la fecha y el número de página. El siguiente ejemplo produce un informe que comenzará con 10 líneas en blanco, seguidas de 50 líneas de información:

```
SET PAGESIZE 60
SET NEWPAGE 10
```
- **SET TERMOUT OFF** y **SET TERMOUT ON**. Cuando se ejecuta el listado con la orden **START**, éste aparece en pantalla. Si se tratase de un listado de miles de líneas, esto supondría una pérdida de tiempo. Mediante estas órdenes se puede evitar que nuestro listado aparezca en la pantalla, sin olvidar que para no perder los datos del listado sería necesario incluir una orden **SPOOL**.

Con **SET TERMOUT OFF** se evita que los datos aparezcan en la pantalla; se suele usar antes de la orden **SPOOL**. Si esta orden se incluye en el listado, después de teclear **START LISTADO** desde SQL*Plus no aparecerá nada en la pantalla. Con **SET TERMOUT ON** los datos aparecerán en la pantalla. Esta opción se suele utilizar después de la orden **SPOOL**.

A la hora de ejecutar las órdenes necesarias para obtener un listado, es posible emplear un archivo que ya las contenga o directamente teclearlas desde el indicador de SQL*Plus.

8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



Actividades propuestas



- 1 Prueba desde el indicador SQL*Plus el comando SET TERMOUT OFF antes de ejecutar el LISTADO.SQL. ¿Qué ocurre? Prueba después SET TERMOUT ON. ¿Qué ocurre?

- **SET SQLCASE UPPER.** Esta orden indica a SQL*PLUS que convierta el texto introducido por teclado (con la orden ACCEPT) a mayúsculas.

C. Órdenes TTITLE y BTITLE. Encabezados y pies de página

Con estas órdenes ponemos títulos en la parte superior e inferior de cada página del informe. Por defecto, el título aparece centrado, teniendo en cuenta el tamaño de la línea definido en la orden LINESIZE. No obstante, podemos cambiar la aparición del título con las cláusulas LEFT, RIGHT y CENTER, que sirven para alinear texto a la izquierda, a la derecha o al centro, respectivamente. Si se usa TTITLE y BTITLE con estas cláusulas no se deben utilizar el separador definido en la orden SET HEADSEP, ya que la salida no sería completamente correcta. Ejemplos:

```
TTITLE   CENTER 'Listado de los empleados de la empresa'
BTITLE   RIGHT  'Por cada departamento'
```

A veces interesa que, después de escribir los títulos, aparezca una o dos líneas en blanco. Para ello, se usa la orden **SKIP n+1**, donde *n* es el número de líneas que se van a saltar. El siguiente ejemplo salta una línea después de escribir el título:

```
TTITLE   CENTER 'Listado de los empleados de la empresa'
SKIP 2
```

Por defecto, cuando se obtiene un informe aparece en la primera línea del listado la fecha y el número de página; si se usa SKIP, esta primera línea no aparecerá.

Para que los títulos superior e inferior no vuelvan a aparecer al realizar una consulta, se deben anular de la siguiente manera:

```
SQL> TTITLE OFF
SQL> BTITLE OFF
```

Y si se desea que vuelvan a aparecer, se activan así:

```
SQL> TTITLE ON
SQL> BTITLE ON
```

Se puede incluir una serie de variables con información de sistema en las órdenes TTITLE y BTITLE. Se trata de estas variables:

- **SQL.LNO:** devuelve el número de línea actual.



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

- SQL.RELEASE: devuelve la versión de Oracle.
- SQL.PNO: devuelve el número de página actual.
- SQL.USER: devuelve el nombre de usuario actual.

Por ejemplo, si se teclea la siguiente sentencia desde el indicador de SQL*Plus:

```
TTITLE LEFT 'APELLIDO DE LOS EMPLEADOS' CENTER 'Usuario: ' -  
SQL.USER SKIP 1 LEFT 'Pagina:' FORMAT 999 SQL.PNO
```

Se creará el siguiente título superior:

```
APELLIDO DE LOS EMPLEADOS      Usuario: SCOTT  
Pagina: 1
```

Nota: El guión (-) detrás del literal 'Usuario:' significa que la orden TTITLE continúa en la siguiente línea. Si se ejecuta una consulta después de definir el TTITLE, se visualizará este título en la consulta.



Actividades propuestas

- 2 Prueba desde el indicador SQL*Plus los comandos TTITLE y BTITLE. Después ejecuta una consulta. ¿Qué ocurre? ¿Cómo evitamos que aparezcan los títulos superior e inferior?

D. Orden COLUMN. Formato de columnas

Esta orden formatea el contenido de las columnas y sus cabeceras. Se puede usar directamente desde el indicador de SQL*Plus para dar formato a las columnas y cabeceras de una sentencia SELECT. Por ejemplo, escribimos la siguiente orden: COLUMN SALARIO FORMAT 99G999D99 HEADING "Salario del empleado" y a continuación se ejecuta la orden SQL: SELECT APELLIDO, SALARIO FROM EMPLE WHERE DEPT_NO=10; La salida será:

APELLIDO	Salario del empleado
CEREZO	2.885,00
REY	4.100,00
MUÑOZ	1.690,00

Se ha dado formato a la columna SALARIO y se ha cambiado la cabecera para esta columna. Cada vez que se ejecute una sentencia SELECT que haga referencia a la columna SALARIO aparecerá con este formato. Si se desea que el formato definido desaparezca se usa la orden CLEAR COLUMNS.

8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



A continuación, se comenta cada una de las filas del listado expuesto al principio utilizando esta orden:

- **COLUMN DENOM**, formato de la columna DENOM. El formato para esta columna se define en tres líneas:

```
COLUMN DENOM HEADING 'Artículo|Vendido'
COLUMN DENOM FORMAT A15
COLUMN DENOM TRUNCATED
```

En la primera línea, **COLUMN DENOM HEADING 'Artículo|Vendido'**, se emplea la orden **HEADING** para definir la cabecera de la columna. El texto que aparece a la derecha es la nueva cabecera que, además, usa el carácter separador «|», para indicar que la cabecera de la columna aparecerá en dos líneas.

En la segunda línea, **COLUMN DENOM FORMAT A15**, la cláusula **FORMAT** define el formato para el contenido de la columna. La A, que aparece a la derecha de **FORMAT**, indica que se da formato a una columna de tipo alfanumérico, y el número que sigue a continuación representa la cantidad de caracteres. Si **DENOM**, que está definida en la tabla con 18 caracteres, tiene más de 15 caracteres, aparecerá dividida en dos líneas. Para evitar esto se recurre a la opción **TRUNCATED**, que trunca el contenido de la columna, y sólo se consideran los 15 caracteres: **COLUMN DENOM TRUNCATED**.

Todas las cláusulas de formato para esta columna se pueden incluir en una única orden **COLUMN**: **COLUMN DENOM HEADING 'Artículo|Vendido' FORMAT A15 TRUNCATED**.

- **COLUMN FECHA_VENTA**: **COLUMN FECHA_VENTA HEADING 'Fecha|Venta'**. Se usa **HEADING** para definir la cabecera de la columna **FECHA_VENTA**.
- **COLUMN UNI_VEND**: **COLUMN UNI_VEND HEADING 'Unidades|Vendidas' FORMAT 9G999**. Con **HEADING** asignamos una nueva cabecera a la columna **UNI_VEND**. Con **FORMAT 9G999** se define una columna numérica con espacio para cuatro dígitos y el punto de los miles. Si la columna **UNI_VEND** tiene más de cuatro dígitos en alguna fila, aparecerán caracteres # en la columna correspondiente.
- **COLUMN IMPORTE**: **COLUMN IMPORTE FORMAT 9G999D99**. Aquí se observa una columna que no aparece en ninguna de las tablas y que, sin embargo, aparecerá en el informe. ¿Qué es **IMPORTE**? Si nos fijamos en la **SELECT** del listado:

```
SELECT DENOM, FECHA_VENTA, UNI_VEND,
       UNI_VEND * PVP "IMPORTE"
FROM TARTICULOS A, TVENTAS V WHERE A.CODIGO = V.CODIGO
AND FECHA_VENTA BETWEEN '01/05/2006' AND '31/05/2006'
ORDER BY DENOM, FECHA_VENTA;
```

Se ve que **IMPORTE** es el alias utilizado para representar el producto de **UNI_VEND * PVP**. Las comillas del alias **IMPORTE** son opcionales.



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

Podemos dar formato a columnas que sean fruto de alguna expresión, siempre y cuando se dé un alias a la columna. Por ejemplo, si usamos la función SUM aplicada a la columna UNI_VEND y queremos que aparezca en el informe, tendremos que darle un alias: SUM (UNI_VEND) SUMA. SUMA es el alias.

Las cabeceras para las columnas a las que se ha dado formato son:

Artículo	Fecha	Unidades	
Vendido	Venta	Vendidas	IMPORTE

E. Orden BREAK ON. Ruptura de control

Esta instrucción produce una *ruptura de control* por la columna que se desee. Una **ruptura de control** nos permite obtener un resultado o varios sobre algo que tienen en común varias filas. En el listado se hace una ruptura por la denominación del artículo, ya que se desea conocer la suma de unidades y la suma de importes por artículo. La ruptura se representa de la siguiente manera: **BREAK ON** DENOM SKIP 2.

Esto significa que se van a imprimir las líneas referidas a un artículo, una a continuación de la otra. El artículo se mostrará una vez, en lugar de aparecer fila a fila y, cuando cambie de artículo, se hará un salto de dos líneas. La orden BREAK ON necesita la cláusula ORDER BY de la sentencia SELECT para que la salida del listado sea la deseada, ya que, si los mismos artículos no están uno a continuación de otro, los totales por artículo no tendrían sentido, pues no aparecerían agrupados. Por eso, cada BREAK ON debe tener su ORDER BY asociado: ORDER BY **DENOM**, **FECHA_VENTA**;

El salto de líneas se especifica con la cláusula SKIP: **SKIP** n: Salta n líneas.

Con la cláusula DUPLICATE se puede hacer aparecer la denominación del artículo en todas las líneas. Por ejemplo: **BREAK ON** DENOM **DUPLICATE** **SKIP** 2

Todas las rupturas de control definidas se pueden eliminar con **CLEAR BREAKS**. Sólo puede haber una orden BREAK activa, por lo que, si se especifican varias rupturas, han de especificarse todas las columnas en una sola orden.



Actividades propuestas

- 3 Prueba en el LISTADO la orden BREAK ON con la cláusula DUPLICATE.



F. Orden COMPUTE. Cálculos

Con esta orden se hacen cálculos con grupos de filas seleccionados. El formato es:

```
COMPUTE {SUM, AVG, COUNT, MAX, MIN, STD, VAR, NUM}
OF {expresión|columna}
ON {expresión|row|page|report}
```

- Opciones para COMPUTE

SUM:	Calcula la suma.
AVG:	Calcula la media.
COUNT:	Cuenta las filas no nulas.
MAX:	Calcula el valor máximo.
MIN:	Calcula el valor mínimo.
STD:	Calcula la desviación estándar.
VAR:	Calcula la varianza.
NUM:	Es similar a COUNT, con la diferencia de que cuenta todas las filas (incluidas las nulas).

- **OF** {expresión|columna}: columna o expresión sobre la que deben realizarse los cálculos.
- **ON** {expresión|row|page|report}: indica sobre qué punto de BREAK se desea introducir el cálculo:

expresión: cuando cambia el valor de expresión.

row: en cada cambio de fila.

page: en cada final de página.

report: al final de un informe o consulta.

Los totales que calcula COMPUTE son los especificados en la sección indicada por BREAK ON. Esto significa que no se puede usar COMPUTE sin BREAK. En el listado aparecen las siguientes órdenes BREAK y COMPUTE:

```
BREAK ON DENOM SKIP 2
COMPUTE SUM OF UNI_VEND ON DENOM
COMPUTE SUM OF IMPORTE ON DENOM
```

Con la indicación de que se produce una ruptura por DENOMINACION. La primera orden COMPUTE calcula la suma (**SUM**) de las columnas UNI_VEND para cada DENOMINACION. La segunda orden COMPUTE calcula la suma (**SUM**) de las columnas IMPORTE para cada DENOMINACION.

Cada vez que SQL*PLUS detecta una nueva DENOMINACION, se calculan las sumas y se imprime para el artículo anterior.



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

Como se observa en el listado, cada COMPUTE SUM pone una fila de asteriscos al final de la fila en la que se hace la separación entre secciones, e imprime la palabra «sum» debajo de ella. Esto es válido también para COUNT, AVG, MAX, MIN, etcétera.



Caso práctico

- 2** A partir del LISTADO, realiza una ruptura por DENOMINACION calculando la suma de las columnas IMPORTE (SUM), las cuentas (COUNT) y el máximo (MAX) importe por cada DENOMINACION:

```
BREAK ON DENOM          SKIP 1
COMPUTE SUM COUNT MAX OF IMPORTE ON DENOM
```

Obtén la suma de columnas IMPORTE al final del informe:

```
BREAK ON REPORT
COMPUTE SUM OF IMPORTE ON REPORT
```

Obtén la suma de IMPORTE por cada denominación y al final del informe. La salida sería similar a:

```
BREAK ON REPORT ON DENOM
COMPUTE SUM OF IMPORTE ON REPORT DENOM
```

Para saber en cualquier momento cuáles son los BREAK y COMPUTE activos, basta con teclear desde el indicador SQL*Plus estas palabras:

```
SQL> BREAK
break on report nodup
on DENOM nodup
```

```
SQL> COMPUTE
COMPUTE sum LABEL 'sum' OF UNI_VEND ON DENOM
COMPUTE sum LABEL 'sum' OF IMPORTE ON REPORT
COMPUTE sum LABEL 'sum' OF IMPORTE ON DENOM
```

Se puede cambiar la palabra 'sum' que aparece en el informe indicando las sumas, con la cláusula LABEL: **COMPUTE SUM LABEL 'SUMA IMPORTES' OF IMPORTE ON REPORT.**

Para eliminar los puntos de ruptura, los cálculos o las definiciones de columnas se usa la orden CLEAR:

```
SQL> CLEAR BREAKS      SQL> CLEAR COMPUTES      SQL> CLEAR COLUMNS
breaks limpiado        computes limpiado      columns limpiado
```


8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



Podemos hacer CLEAR desde el indicador de SQL*Plus o en el archivo que contiene las órdenes para realizar el informe.

G. Orden SPOOL

Esta orden posibilita guardar en un archivo toda la salida SQL*Plus que se vaya produciendo sobre la pantalla, hasta que se indique a SQL*Plus que pare. Esta acción se realiza mediante esta orden: SPOOL OFF.

Por eso, en el listado se usa la orden SPOOL para guardar toda la salida del informe en un archivo y, así, disponer del informe de salida para utilizarlo en cualquier momento. El listado de salida queda almacenado en el archivo LISTADO.LST: SPOOL LISTADO.LST.

Y si ahora consultamos el archivo LISTADO.LST, se verá el informe que se ha hecho.

Actividades propuestas



- 4 A partir de las tablas EMPLE y DEPART, haz un listado en el que se produzca una suma de salarios por LOCALIDAD y, dentro de la localidad, una suma de salarios por OFICIOS.

El título superior del listado es: 'LISTADO DE LOS EMPLEADOS DE LA TABLA EMPLE'.

El título inferior del listado es: 'Con ruptura por departamento y por oficio calculando totales'.

La cabecera estará compuesta por los siguientes títulos:

Localidad	Apellidos	Nº de empleado	Oficio	Salario	Comisión
-----------	-----------	----------------	--------	---------	----------

H. Definición y uso de variables. DEFINE y ACCEPT

Con SQL*Plus podemos definir variables para utilizarlas después en un informe con otras órdenes SQL*Plus. Así, por ejemplo, en nuestro informe es posible definir variables que almacenen las líneas de cabeceras y de pie. Para eso sirve la orden DEFINE.

(Continúa)



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

Si se teclea la palabra **DEFINE** desde el indicador SQL*Plus, aparecerá la lista de variables definidas:

```
SQL> DEFINE
DEFINE _DATE = "22/06/05" (CHAR)
DEFINE _CONNECT_IDENTIFIER = "ORACLE10G" (CHAR)
DEFINE _USER = "SCOTT" (CHAR)
DEFINE _PRIVILEGE = "" (CHAR)
DEFINE _SQLPLUS_RELEASE = "1001000200" (CHAR)
DEFINE _EDITOR = "Notepad" (CHAR)
DEFINE _O_VERSION = "Oracle Database 10g Enterprise Edition Release 10.1.0.2.0 - Production With the Partitioning, OLAP and Data Mining options" (CHAR)
DEFINE _O_RELEASE = "1001000200" (CHAR)
DEFINE _RC = "1" (CHAR)
```

Donde **DEFINE _O_VERSION** y **DEFINE _O_RELEASE** definen la versión Oracle, **_USER** el usuario conectado, **_DATE** la fecha del sistema y **_CONNECT_IDENTIFIER** el nombre de la base de datos global.

Ahora deseamos añadir a esta lista de variables otras nuevas: definimos, por ejemplo, tres variables para pintar los títulos superior e inferior de nuestro listado. Una vez definidas se podrán usar en sentencias SQL*Plus. **CABECERA** contendrá el título superior de nuestro listado; **RAYA** contendrá el subrayado; por último, la variable **PIE** contendrá el título inferior del listado:

También se pueden definir las variables dentro de los archivos SQL.

```
SQL> DEFINE CABECERA="VENTAS POR ARTÍCULOS DURANTE MAYO "
SQL> DEFINE RAYA="===== "
SQL> DEFINE PIE="Sólo Mayo de 2006"
```

Si ahora escribimos de nuevo la orden **DEFINE**, aparecerán en la lista de variables estas tres que acabamos de definir.

Y las podremos emplear en el listado para poner las líneas de títulos superiores e inferiores, de la siguiente manera:

```
TTITLE CENTER CABECERA SKIP 1 CENTER RAYA
BTITLE LEFT PIE
```

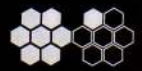
Cuando queremos eliminar las variables definidas con **DEFINE**, las podemos suprimir con **UNDEFINE**. De este modo, si ya no deseamos usar la variable **CABECERA**, escribimos desde SQL*Plus:

```
SQL> UNDEFINE CABECERA.
```

La variable **CABECERA** ya no existe y, si se ejecuta la orden **DEFINE**, ya no aparecerá en la lista.

8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



Orden ACCEPT

Hasta ahora, en el listado no se pedía introducir ninguna información por teclado: se ejecutaba sin más. No obstante, es posible diseñar un listado que pida introducir cierta información por teclado, como el número del departamento que se va a listar. De este modo, con un mismo listado se pueden obtener los empleados del departamento 10, los del departamento 20, etcétera. Para hacer este tipo de listados se dispone de la orden **ACCEPT**, que pide que se introduzcan datos por teclado. Su formato es:

```
ACCEPT Variable [NUMBER|CHAR] [PROMPT texto |NOPROMPT]
[HIDE]
```

Variable es el nombre de la variable que recibe la información tecleada.

NUMBER|CHAR determina el tipo de dato de la variable que se introduce y que, por defecto, es CHAR. Si se teclea un dato que no corresponde con el tipo especificado se produce un mensaje de error y vuelve a pedir la entrada. El siguiente ejemplo requiere que la entrada sea de tipo numérico y se teclea una cadena de caracteres:

```
SQL> ACCEPT DEPAR NUMBER PROMPT 'Departamento: '
Departamento: ss
SP2-0425: "ss" no es NUMBER válido
Departamento: 10
```

PROMPT 'texto' muestra el texto al usuario antes de teclear los datos. NOPROMPT salta una línea y espera la entrada. Ejemplo: **ACCEPT DEPAR NUMBER NOPROMPT**

HIDE es útil para contraseñas, al escribir el dato se visualizarán asteriscos. Por ejemplo:

```
SQL> ACCEPT DEPAR NUMBER PROMPT 'Departamento: ' HIDE
Departamento: **
```

Caso práctico

- 3 Utilizamos la orden **ACCEPT** para aceptar un número de departamento; el valor tecleado se almacenará en una variable (**NDEP**). Si se ejecuta la orden **DEFINE**, esta variable aparecerá con el valor que se acaba de teclear.

```
SQL> ACCEPT NDEP NUMBER PROMPT 'Teclea N° Departamento a listar : '
Teclea N° Departamento a listar : 10
```

Consultamos los datos de los empleados de la tabla EMPLE (apellido, número de empleado, oficio y salario) cuyo número de departamento coincida con el valor tecleado:

```
SELECT APELLIDO, EMP_NO, OFICIO, SALARIO
FROM EMPLE
WHERE &NDEP = DEPT_NO
ORDER BY APELLIDO;
```

(Continúa)



8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes

(Continuación)

La columna DEPT_NO se compara con la variable NDEP, la cual debe llevar un ampersand (&) delante: &NDEP. Debe ir encerrada entre comillas simples si se define de tipo CHAR.

Consultamos los datos de los empleados de la tabla EMPLE cuyo OFICIO coincida con un oficio leído de teclado:

```
SQL> ACCEPT OFI PROMPT 'Teclea Oficio a listar : '
Teclea Oficio a listar : ANALISTA
```

```
SQL> SELECT * FROM EMPLE WHERE '&OFI' = OFICIO;
```

En este caso la variable OFI es de tipo CHAR por lo que debe ir encerrada entre comillas simples: '&OFI'.

Se pueden usar las variables de entrada en las órdenes TTITLE o BTITLE. Por ejemplo:

```
TTITLE CENTER 'Listado del departamento : ' NDEP SKIP 1
BTITLE CENTER 'Listado del oficio : ' OFI SKIP 1
```



Actividades propuestas

- 5 A partir de las tablas EMPLE y DEPART visualizamos los datos de los empleados (LOCALIDAD, APELLIDO, EMP_NO, SALARIO, COMISION) del departamento y del oficio leído de teclado. Incluye en el título superior el departamento que se lista y en el título inferior el oficio que se lista. Utiliza variables DEFINE para definir cabeceras y pies.

Hemos de hacer una observación: cuando se usa la orden ACCEPT para introducir datos por teclado, la variable a la derecha del ACCEPT queda definida. Podemos verla con la orden DEFINE. Y podemos utilizar esta variable en una orden SELECT con el ampersand delante.

También existe la posibilidad de usar el ampersand como prefijo delante de una variable que no esté definida; en este caso, SQL*Plus, al encontrarse con ella, interpreta una entrada de teclado, pero no define ni guarda su valor, únicamente sustituye una vez lo que se introduce por teclado.

En la siguiente sentencia SELECT se consultan aquellos apellidos cuyo número de departamento es igual al valor introducido por teclado. Para eso se usa el ampersand acompañando a un nombre de variable cualquiera, por ejemplo, **DEPARTA** (&DEPARTA). Esta variable será sustituida por el valor que se introduce por teclado.

8. Generación de informes SQL*Plus

8.3 Órdenes para generar informes



En la siguiente sentencia SELECT se consultan aquellos apellidos cuyo número de departamento es igual al valor introducido por teclado. Para eso se usa el ampersand acompañando a un nombre de variable cualquiera, por ejemplo, DEPARTA (&DEPARTA).

Esta variable será sustituida por el valor que se introduce por teclado:

```
SQL> SELECT APELLIDO FROM EMPLE WHERE DEPT_NO=&DEPARTA;
```

Introduzca un valor para departa: 20

```
antiguo 1: SELECT APELLIDO FROM EMPLE WHERE DEPT_NO=&DEPARTA
```

```
nuevo 1: SELECT APELLIDO FROM EMPLE WHERE DEPT_NO=20
```

Apellidos

SANCHEZ

JIMENEZ

GIL

ALONSO

FERNANDEZ

La variable DEPARTA no formará parte de la lista de variables definidas y no se visualizará con la orden DEFINE.



8. Generación de informes SQL*Plus Conceptos básicos

Conceptos básicos

Desde la opción de menú *Opciones/Entorno* (véase Figura 8.3) se pueden modificar variables de la sesión actual, las mismas variables que se pueden modificar con la orden SET.

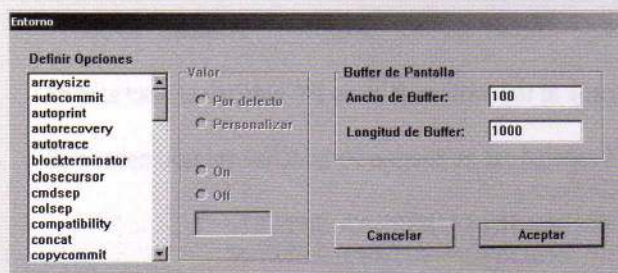


Fig. 8.3. Entorno SQL*Plus.

Se visualiza un cuadro de diálogo con tres áreas:

Definir Opciones: En este área se presenta la lista de variables cuyos valores se pueden modificar. Tal es el caso de LINESIZE, AUTOCOMMIT, TERMOUT, PAGESIZE, etcétera.

Valor: Permite dar valores a las variables seleccionadas. Las opciones son: *Por defecto*, *Personalizar*, *On* y *Off*.

Buffer de Pantalla: Permite definir el ancho y la longitud del buffer de pantalla. En el campo *Ancho de Buffer* se sitúa el número de caracteres que puede mostrar SQL*Plus en una línea. En el campo *Longitud de Buffer* se coloca el número de líneas que puede mostrar SQL*Plus en la pantalla.

Escribiendo desde el indicador SQL*Plus: HELP INDEX se obtienen los comandos a utilizar en SQL*Plus (ver Figura 8.4).

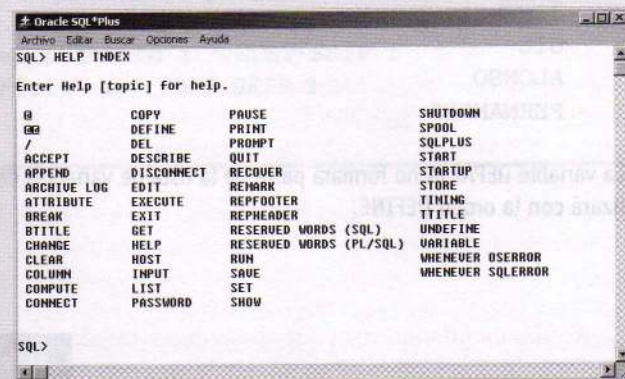


Fig. 8.4. Comandos a usar en SQL*Plus.



Actividades complementarias



- 1 Partiendo de las tablas EMPLE y DEPART, obtén por cada departamento los datos de los empleados APELLIDO y SALARIO, el salario máximo, el salario mínimo, el total salario y el salario medio. Obtén los mismos cálculos por report.

- El título superior del listado es: 'LISTADO DE LOS SALARIOS DE LOS EMPLEADOS'.
- El título inferior del listado es: 'Con resumen por departamento y por reporte'.
- La cabecera del listado es:

Nombre	Apellido	Salario
-----	-----	-----

- 2 Realiza un informe en el que aparezca por cada departamento el salario máximo, el salario mínimo, el total salario, el salario medio y el número de empleados.

- El título superior del listado es: 'RESUMEN DEPARTAMENTAL'.
- La cabecera del listado contendrá las siguientes columnas:

Nombre Departamento, Máximo Salario, Mínimo Salario, Suma Salarios, Salario Medio, Número Empleados

- 3 Haz el mismo informe que se pide en la Actividad 1, con la diferencia de que nos pida introducir por teclado el nombre del departamento que se va a listar. No se han de obtener los cálculos por REPORT ni imprimir el título inferior.

A partir de las tablas CLIENTES08, PRODUCTOS08 y PEDIDOS08:

- 4 Lee por teclado un nombre de cliente y obtén sus pedidos. En la cabecera debe aparecer el nombre del cliente leído, la línea de detalle debe contener la siguiente información:

Nº	Pedido,	Descripción	Producto,
Fecha,	Unidades,	Importe	

Al final del listado se debe obtener el número de pedidos y la suma de todos los importes.

- 5 Obtén por cada producto con una línea de detalle por producto: la descripción del producto, el número de pedidos que se han realizado con dicho producto y el total de de la cantidad pedida. La cabecera deberá contener la siguiente información:

PEDIDOS REALIZADOS POR PRODUCTO

Al final del informe consigue el número total de pedidos y la suma total de las unidades pedidas.