

## Oracle básico (V): SQLReport

En principio me había propuesto tratar dos temas en esta entrega: reportes (informes) y menús. Pero, dada la extensión del primero de ellos, decidí dejar el diseño de menús para el siguiente artículo.

El módulo *SQLReport* de Oracle realiza de forma flexible, sencilla y eficiente la creación de reportes, informes o listados permitiendo, entre otras facilidades, la visualización previa por pantalla con una gran variedad en estilos de presentación.

### Definiciones básicas

Para adentrarnos en el tema primero veremos las definiciones básicas, fundamento del diseño del reporte en Oracle:

#### Consulta

Define las columnas y filas de una o varias tablas que serán emitidas en el reporte, así como su orden de presentación. Una consulta puede estar subordinada a otra consulta principal, relacionadas por una o varias columnas.

#### Grupo

El grupo es una sección del reporte que representa al conjunto de columnas de la consulta, como una unidad, para determinar su ubicación en el reporte y su forma de presentación.

Para cada grupo se definen los siguientes atributos:

- Ubicación.
- Forma de presentación.
- Texto de cabecera y final.
- Título de las columnas.
- Ubicación de las columnas dentro del grupo.

En el momento de la definición de una consulta se crea, de forma automática, un grupo que contiene todas las columnas presentes en la lista de la sentencia *SELECT*. El nombre de este grupo se define como el nombre de la consulta, precedido por los caracteres *G\_*. Por ejemplo, más adelante veremos como con la definición de la consulta *VENTAS* se genera, de forma automática, el grupo *G\_VENTAS*.

Una consulta puede tener asociado más de un grupo, con el fin de separar las columnas de la consulta, por ejemplo para crear diferentes niveles en el reporte, como el cálculo de subtotales.

#### Campos

Cada columna de la consulta pasa a ser un campo del reporte. Además, podemos incluir como campos del reporte las siguientes variables:

- Del sistema como *&DATE* y *&PAGE*
- Del usuario, cuyo contenido es un comando SQL para realizar cálculos.

## Parámetro

Variables definidas por el usuario para transferir datos a la consulta en el momento de la ejecución del reporte. Se hace referencia al parámetro en la consulta con su nombre precedido por dos puntos como *:nombre\_parametro*.

## Sumario

Define variables de tipo sumario para la realización de cálculos con el uso de las funciones *Sum*, *Min*, *Max*, *Count* y *Avg*.

## Texto

Define la ubicación y forma de presentación de los campos dentro del grupo, así como los textos de inicio y final para cada grupo y para el reporte.

Para ubicar un campo o un parámetro en la sección de textos se hace referencia con su nombre, precedido con el carácter & de la forma: *&nombre*.

## Primeros pasos

Veamos, a través del siguiente ejemplo, la creación del reporte *VENTAS*, cuyo objetivo es detallar las ventas realizadas en los últimos *n* días para un determinado cliente, identificado por su código y cuyo resultado debe ser presentado ordenado por artículo y fecha de venta.

Primero definimos la consulta *VENTAS* escribiendo la correspondiente sentencia *SELECT* como se muestra a continuación:

```
SELECT ARTICULO, FECHA, VALOR
FROM VENTAS
WHERE
    CODIGO=:XCOD
    AND FECHA>=SYSDATE-:XDias
ORDER BY ARTICULO, FECHA
```

Obsérvese que la sentencia *SELECT* de la consulta **no** debe terminar con punto y coma (;).

La consulta *VENTAS* debe recibir, en el momento de la realización del reporte, los siguientes parámetros:

- XCod para el código del cliente a consultar.
- XDias para la cantidad de días de las ventas.

Veamos a continuación la tabla 1, donde se da la definición de los parámetros:

Nombre del parámetro	Tipo	Ancho	Valor por defecto	Etiqueta
XCod	Num.	10	2	Código Cliente
XDias	Num.	5	15	Cantidad de días

Tabla 1: Definición de los parámetros

Desde mi punto de vista, el gran mérito del diseño de reporte en Oracle es su sencillez, ya que una vez realizadas las definiciones de consulta y parámetros, el reporte está pronto para ser emitido, gracias a que *SQLReport* se encarga del resto, incorporando, en forma automática, las siguientes definiciones:

- 1.- Creación del grupo *G\_VENTAS* asociado a la consulta *VENTAS*.

Group Name	G_VENTAS
Query	VENTAS
Print Direction	Down

- 2.- Definición de los campos del reporte, tomados de la lista de la sentencia *SELECT* de la consulta *VENTAS*. Ver tabla 2.

Field Name	Source	Group	Label
ARTICULO	ARTICULO	G_VENTAS	Articulo
FECHA	FECHA	G_VENTAS	Fecha
VALOR	VALOR	G_VENTAS	Valor

Tabla 2: Campos del reporte

- 3.- Adiciona el texto de cabecera de cada columnas y la ubicación de las columnas dentro del grupo. Los elementos de la pantalla que tienen asignado algún valor son:

Lines Before:	1
Repeat On Page Overflow:	X
Justification:	Left
&ARTICULO	
&FECHA	
&VALOR	

Ahora terminaremos el estudio de los elementos básicos del diseño de reportes viendo cómo emitir la identificación del cliente con su código, contenido en el parámetro *XCod*. Para esto incluiremos en el texto de cabecera del grupo del sistema *Report* (cuyo objetivo es representar parámetros y variables no asociados a ninguna consulta) la siguiente definición:

Object:	REPORT
Type:	Header
Status:	Edited
Justification:	Left
Cliente N°:	&XCOD

En este momento nuestro reporte esta pronto para ser generado y emitido, para lo cual debemos elegir las tareas del menú *Generate* y *Execute*. Al momento de ejecutar el reporte, como se muestra en la tabla 3, se pedirá que se ingresen los valores de los parámetros:

Parameter	Value
Código Cliente	2
Cantidad de días	15

Tabla 3: Valores de los parámetros

El resultado de nuestro reporte *VENTAS* para el cliente 2 es el de la tabla 4.

Artículo	Fecha	Valor
DISKETTE	21/11/95	33
DISKETTE	21/11/95	100
PAPEL	22/11/95	150
PAPEL	27/11/95	53

Tabla 4: El resultado del reporte *VENTAS*

## Mejorando nuestro reporte

Para ampliar y mejorar nuestro reporte, y con el fin de seguir avanzando en su diseño, estudiaremos cómo realizar las siguientes tareas:

- Imprimir la fecha del momento de emisión del reporte.
- Imprimir la fecha a partir de la cual se consultan las ventas.
- Incluir el nombre del cliente.

Comencemos por incluir la impresión de la fecha de emisión en la cabecera del reporte, para lo cual es necesario seguir los siguientes pasos:

- 1.- Crear el campo *EMISION* que contiene la variable del sistema *&DATE* que pertenece al grupo *REPORT*.

Field Name	EMISION
Source	&DATE
Group	REPORT
Label	Emission

- 2.- Modificar la cabecera del reporte para incluir el campo *EMISION* como se muestra a continuación (campos con datos, exclusivamente):

Object:	REPORT
Type:	Header
Status:	Edited
Justification:	Left
FECHA:	&EMISION
Cliente N°:	&XCOD

Ahora, añadiremos en el reporte la fecha a partir de la cual se consultan las ventas, que se obtiene a partir de la fórmula *SYSDATE-:XDIAS*; para lo cual debemos realizar las siguientes tareas:

- 1.- Crear el campo *XDESDE* cuyo contenido es una sentencia *SELECT* del tipo *&SQL* para realizar el cálculo *SYSDATE-:XDIAS*. Este campo pertenece al grupo *REPORT*.

```
&SQL SELECT SYSDATE-:XDIAS
        INTO :XDESDE FROM DUAL
```

El comando *SQL*, asociado al campo *XDESDE*, es una sentencia *SELECT* con las siguientes características:

- Comienza con la palabra clave *&SQL*.
  - Calcula una fórmula desde la tabla simbólica del sistema llamada *DUAL*.
  - El resultado se pasa a la propia variable *:XDESDE* con el uso de la orden *INTO*.
  - Sólo puede devolver una fila.
- 2.- Modificar la cabecera del reporte para incluir la fecha a partir de la cual se consultan las ventas, contenida en el campo *XDESDE*, como se muestra a continuación:

Object:           REPORT  
 Type:            Header  
 Status:          Edited  
 Justification:   Left  
 FECHA:           &EMISION  
 Ventas desde:    &XDESDE  
 Cliente N° :      &XCOD

Por último, veamos cómo incorporar el nombre del cliente junto a su código, con la realización de los siguientes pasos:

- 1.- Añadir una nueva consulta, llamada *CLIENTE*, con el fin de recuperar el nombre del cliente desde la tabla *CLIENTES*, como se muestra en la siguiente *SELECT*:

```
SELECT NOMBRE, CODIGO
FROM CLIENTES
WHERE CODIGO=:XCOD
```

- 2.- Modificar la consulta *VENTAS* para que pase a ser una consulta subordinada (subconsulta) a la consulta principal *CLIENTE*, relacionadas, además, por la columna *CODIGO* . Ver *SELECT* a continuación.

```
SELECT ARTICULO, FECHA, VALOR, CODIGO
FROM VENTAS
WHERE FECHA>=SYSDATE-:XDIAS
ORDER BY ARTICULO, FECHA
```

Donde

Child Columns      CODIGO  
 Parent 1 Columns   CODIGO

- 3.- Ubicar el grupo *G\_CLIENTE*, creado de forma automática por *SQLReport*, como primer grupo. Ver tabla 5.

Group name	Query	Print direction	Relative position
G_CLIENTE	CLIENTE	Down	

G_VENTAS	VENTAS	Down
----------	--------	------

Tabla 5: Ubicar el grupo G\_CLIENTE

- 4.- Cambiar la posición del grupo G\_VENTAS relativa a su grupo principal G\_CLIENTE, que por defecto es a la derecha, a la posición debajo (*Below*). Ver tabla 6


Group name	Query	Print direction	Relative position
G_CLIENTE	CLIENTE	Down	
G_VENTAS	VENTAS	Down	Below

Tabla 6: Modificación de la posición relativa

- 5.- Definir el texto para el grupo G\_CLIENTE como se muestra en la siguiente relación:

Object: G\_CLIENTE  
 Type: Body  
 Status: Edited  
 Repeat On Page Overflow: X  
 Justification: Left  
 Nombre: &NOMBRE

La emisión del reporte después de las nuevas definiciones es el de la figura 1:



Informática fungible, S.R.L.

**FECHA: 06/12/95**

**Ventas desde: 11/11/95**

**Cliente número: 2**

**Nombre: MONTEVIDEO CENTER**

Artículo	Fecha	Valor
DISKETTE	21/11/95	33
DISKETTE	21/11/95	100
PAPEL	22/11/95	150
PAPEL	27/11/95	53

Figura 1: Ejemplo de reporte

## Cálculos

Veamos a continuación cómo se incluyen cálculos en el reporte, para lo cual estudiaremos los siguientes casos:

- Cálculo de la cantidad de días de realización de cada venta.
- Venta total al cliente.
- Venta por artículo.

Empecemos por desarrollar el cálculo de la cantidad de días de realización de cada venta, para ello debemos llevar a cabo las siguientes tareas:

- 1.- Añadir en la consulta *VENTAS* la columna *DIAS* que contiene la fórmula *SYSDATE-FECHA*, como se muestra en la siguiente SELECT:

```
SELECT VALOR,ARTICULO,FECHA,
       CODIGO,SYSDATE-FECHA DIAS
FROM VENTAS
WHERE FECHA>=SYSDATE-:XDIAS
ORDER BY ARTICULO,FECHA
```

- 2.- Añadir en el texto de cabecera de las columnas del grupo *G\_VENTAS* el título para la nueva columna:

Object:	G_VENTAS
Type:	Column Heading
Status:	Edited
Repeat On Page Overflow:	X
Lines Before:	1
Justification:Left	
Valor:	Dias °

- 3.- Incluir el campo *DIAS* en el texto del grupo *G\_VENTAS*.

Object:	G_VENTAS
Type:	Body
Status:	Edited
Repeat On Page Overflow:	X
Justification:	Left
&ARTICULO	
&FECHA	
&VALOR	
&DIAS	

Ahora veamos el cálculo de la venta total al cliente, con los siguientes pasos:

- 1.- Definir una variable de tipo sumario, llamada *TOTAL*, asociada al campo *VALOR* y cuya función es su suma, siendo *REPORT* el grupo, tanto de impresión como de cálculo. Ver tabla 7.

Summary name	Field	Funtion	Data type	Print group	Reset group
--------------	-------	---------	-----------	-------------	-------------

TOTAL	VALOR	Sum	NUM	REPORT	REPORT
-------	-------	-----	-----	--------	--------

Tabla 7: Variable *TOTAL* de tipo sumario

2.- Incluir el campo *TOTAL* en el texto final del reporte como se muestra en los datos siguientes:

Object: REPORT  
 Type: Footer  
 Status: Edited  
 Justification: Left  
 Spaces Before: 30  
 TOTAL  
 &TOTAL

Una vez terminadas las nuevas definiciones la emisión del reporte brindará el resultado que se ve en la figura 2:


 <b>FECHA: 06/12/95</b>			
<b>Ventas desde:</b> 11/11/95			
<b>Cliente número:</b> 2			
<b>Nombre:</b> MONTEVIDEO CENTER			
Artículo	Fecha	Valor	Días
DISKETTE	21/11/95	33	15
DISKETTE	21/11/95	100	15
PAPEL	22/11/95	150	14
PAPEL	27/11/95	53	9
<b>TOTAL</b>		336	

Figura 2: Ejemplo de reporte con total

Por último estudiaremos el cálculo de la venta por artículo (subtotales), para lo cual se necesita realizar las siguientes tareas:



- 1.- Definir un segundo grupo para la consulta *VENTAS*, llamado *G\_ARTICULO*, que identificará el campo, para el cual se van a calcular los subtotales que, en nuestro caso, es el campo *ARTICULO*. La ubicación de este grupo debe ser anterior al grupo *G\_VENTAS*.

Es imprescindible que la consulta esté ordenada por el campo en que se van a calcular los subtotales. En nuestro caso la sentencia *SELECT* de la consulta *VENTAS* ya incluye la orden *ORDER BY ARTICULO*. Ver tabla 8.

Group name	Query	Print direction
G_CLIENTE	CLIENTE	Down
G_ARTICULO	VENTAS	Down
G_VENTAS	VENTAS	Down

Tabla 8: Modificación de la posición relativa

- 2.- Cambiar de grupo al campo *ARTICULO* pasándolo de *G\_VENTAS* a *G\_ARTICULO*.

Field Name	ARTICULO
Source	ARTICULO
Group	G_ARTICULO
Label	Articulo

- 3.- Definir la variable *SUBTOTAL*, de tipo sumario, cuyo fin es calcular los subtotales de las ventas, (efectuando la suma del campo *VALOR*), su grupo de cálculo es *G\_ARTICULO* y el de impresión es *G\_VENTAS*. Ver tabla 9.


Summary name	Field	Funtion	Data type	Print group	Reset group
SUBTOTAL	VALOR	Sum	NUM	G_VENTAS	G_ARTICULO

Tabla 9: Variable *SUBTOTAL* de tipo sumario

- 4.- Ubicar el campo *SUBTOTAL* en el texto final del grupo *G\_VENTAS*.

Object:	G_VENTAS
Type:	Footer
Status:	Edited
Justification:	Left
SUBTOTAL	
&SUBTOTAL	

El resultado final de nuestro reporte *VENTAS*, con el cálculo de las ventas por artículo, es el de la figura 3.



Informática fungible, S.R.L.

**FECHA: 06/12/95**

**Ventas desde:** 11/11/95

**Cliente número:** 2

**Nombre:** MONTEVIDEO CENTER

Artículo	Fecha	Valor	Días
DISKETTE	21/11/95	33	19
		100	19
SUBTOTAL .....		133	
PAPEL	22/11/95	150	18
	27/11/95	53	13
SUBTOTAL .....		203	
PAPEL FOTOCOPIA	11/11/95	330	29
SUBTOTAL .....		330	
<b>TOTAL</b>		666	

Figura 3: Ejemplo de reporte con subtotales

Hasta aquí los conceptos básicos del diseño de reportes en Oracle con *SQLReport*. En este caso, como en todos los artículos anteriores, mi recomendación es que a partir de lo estudiado el lector puede seguir profundizando en el tema con el uso de los manuales de Oracle, que son didácticos y con muchos ejemplos ilustrativos.

En nuestro próximo artículo estudiaremos el tema del diseño de menús y la integración de los diferentes módulos estudiados en nuestra serie de artículos.

## Bibliografía

*Oracle 7 Manual de Referencia*  
Koch, George.  
Osborne/McGraw-Hill  
1999.

*Oracle Manual de Referencia.*  
Koch, George.  
Osborne/McGraw-Hill.  
1997.

*Mastering Oracle.*  
Cronin, Daniel.

Hayden Books.  
1990