

## Oracle básico (y VI): SQLMenu

*SQLMenu* es el producto de *ORACLE* destinado a producir árboles de menús que permiten al usuario desplazarse fácilmente a través de su aplicación, facilitando a su vez, la integración con los diferentes módulos de *ORACLE* analizados en nuestros artículos anteriores permitiendo, de esta forma, la conexión con *SQLForms*, *SQLReport*, *SQLPlus*, *PL/SQL* y tareas del sistema operativo.

La integración de módulos, herramienta que permite desde un producto de Oracle invocar otros, también está presente en *SQLForms*, desde donde, como veremos en este artículo, se pueden realizar tareas del sistema operativo con llamadas a los módulos *SQLReport* y *SQLPlus*.

### Definiciones

Comenzaremos explicando las diferentes partes que componen el diseño de un menú en *SQLMenu*. Ellos son:

<i>Aplicación</i>	Conjunto de uno o más menús interconectados para realizar las tareas necesarias del sistema y para el cual se definen su nombre, nombre del archivo ejecutable, fecha de creación, nombre del usuario, número de la versión, fecha de la última modificación, directorio de ubicación y su identificación.
<i>Menú</i>	Lista de opciones o ítems que realizan las tareas específicas del sistema. Cuando se crea la aplicación el primer menú que se debe definir es el principal, cuyo nombre tiene que ser el mismo que el de la aplicación. Este será el menú que se activará en forma automática al cargar la aplicación.
<i>Items</i>	Define las opciones del menú y su correspondiente acción. Un ítem puede llamar a otro menú, ejecutar un comando o un módulo de Oracle, así como cualquier otro programa. Para cada ítem se define su posición en el menú, tipo de comando, permisos de los usuarios, texto de identificación y la línea del comando a realizar.
<i>Parámetros</i>	Variable que se carga en el momento de ejecución del menú y para la cual se definen un nombre de dos letras, cantidad máxima de caracteres, texto en pantalla en el momento de su ingreso, si es o no obligatorio, con o sin conversión a mayúsculas y su valor por defecto.

Existen 5 parámetros del sistema cuya información podemos verla en la tabla 1.

Parámetro	Valor
UN	Nombre del usuario.
PW	Contraseña del usuario.
AD	Directorio actual.
SO	Opción seleccionada del menú.
TT	Tipo de terminal en uso.

Tabla 1. Parámetros del sistema.

Los parámetros se definen en la línea de comandos precedidos por `&`, como por ejemplo `&UN`. Para los bloques de código en *PL/SQL* se hace referencia precedido con dos puntos de la forma `:UN`.

## Tipos de comandos

En *SQLMenu*, desde un menú, se pueden realizar varios tipos de comandos. Pueden verse en la tabla 2.

Tipo	Descripción
1	Llamada a un submenú.
2,3	Tarea del sistema operativo sin y con pausa.
4	Conexión con SQLForms.
5	Conexión con SQLPlus.
6	Realiza un Macro de SQLMenu.
7	Realiza un procedimiento en PL/SQL.

Tabla 2. Lista de comandos que se pueden realizar

A continuación, para conocer cada tipo de comando, diseñaremos una aplicación llamada *CLIENTE*, cuyo menú principal, llamado también *CLIENTE*, esta compuesto de 6 submenús (tipo de comando 1), como se muestra en la tabla 3.

Menú	Nº Item	Tipo comando	Línea comando
CLIENTE	1	1	FORMA
	2	1	REPORTE
	3	1	LISTADO

	4	1	PROCEDIMIENTO
	5	1	MACRO
	6	1	SISTEMA

Tabla 3. Submenús del menú *CLIENTE*

La pantalla de nuestro menú *CLIENTE* se presenta de la siguiente forma:

FORMA REPORTE LISTADO PROCEDIMIENTO MACRO SISTEMA  
 \_\_\_\_\_ CLIENTE \_\_\_\_\_

Pasaremos a detallar cada submenú, para de esta forma detenernos en las particularidades de cada tipo de comando.

**FORMA** Realiza tareas del comando tipo 4 para la integración de SQLMenu con SQLForms. En este caso pasamos como línea de comando la orden RUNFORM con la información necesaria para la conexión, como son, el nombre del usuario, su contraseña y el nombre de la forma a activar. La sintaxis de este línea de comando corresponde a la de la orden RUNFORM desde el sistema operativo, siendo necesario, en este caso, pasar el nombre del usuario y su contraseña, definidos al momento de realizar la conexión con SQLMenu, a través de los parámetros UN y PW.

En la tabla 4 se muestra como se activa la forma *CLIENTES* desde el menú *FORMA*.

Menú	Nº Item	Tipo de comando	Línea de comando
FORM	1	4	RUNFORM CLIENTES &UN/&PW

Tabla 4. Cómo activar la pantalla *CLIENTES*

**REPORTE** La conexión con *SQLReport* no es de forma directa como lo es con *SQLForms*, sino que se debe realizar a través del tipo de comando 2, que se encarga de pasar la línea de comando del menú al sistema operativo para que lo procese. En este caso se siguen las mismas reglas detalladas anteriormente para *SQLForms*. En la tabla 5 se muestra como se activa el reporte llamado *VENTAS*.

Menú	Nº Item	Tipo de comando	Línea de comando
REPORT E	1	2	RUNREP VENTAS &UN/&PW

Tabla 5. Cómo activar el reporte *VENTAS*

**LISTADO** Los listados diseñados en *SQLPlus* se activan con el tipo de comando 5, que es el que permite la integración de *SQLMenu* con *SQLPlus*. La identificación del usuario y su contraseña se logra con los parámetros *UN* y *PW* como explicamos en los casos anteriores.

Menú	Nº Item	Tipo	de co m an do	Línea de comando
LISTADO	1	5		SQLPlus -S &UN/&PW @CLIENTE
	2	5		SQLPlus &UN/&PW \FAC\VENTAS

Tabla 6. Cómo activar el listado *LISTADO*

Los archivos de listado, por ejemplo *CLIENTE.sql*, debe tener *EXIT* como último comando, con el fin de garantizar la desconexión de *SQLPlus* y el regreso a *SQLMenu*. El parámetro *-S* de *SQLPlus* suprime todas las visualizaciones de especificaciones del *SQLPlus* por pantalla.

**PROCEDIMIENTO** Para ejecutar un procedimiento del sistema o del usuario, así como un bloque anónimo de *PL/SQL* escrito directamente en la línea de comando del menú, se utiliza el tipo de comando 7.

Menú	Nº Item	Tipo	de co m an do	Línea de comando
PROCEDIMIENTO	1	7		EXIT_MENU;
	2	7		PROCEDURE CAMBIO IS BEGIN ... END;
	3	7		NUEVO;

Tabla 7. Cómo ejecutar un procedimiento.

En nuestro ejemplo, el ítem 1 es un procedimiento del sistema, el 2 es un bloque anónimo y el 3 es un procedimiento creado por el usuario.

En el fuente 1 vemos la definición del procedimiento *NUEVO*, para de esta forma explicar algunos de ellos:

```
Fuente 1: Procedure Name: NUEVO
-----Procedure Text -----
PROCEDURE NUEVO IS
BEGIN
    NEW_USER('EIDOS/ESPAÑA');
    NEW_APPLICATION('VENTAS');
    MAIN_MENU;
    OS_COMMAND('DIR');
    EXIT_MENU('Fin del Menú');
END;
```

Donde:

*NEW\_USER* Desconecta al usuario actual y conecta un nuevo usuario.  
*NEW\_APPLICATION* Cambia de aplicación.  
*MAIN\_MENU* Pasa al menú principal.  
*OS\_COMMAND* Realiza un comando del sistema operativo.  
*EXIT\_MENU* Salida del Menú.

**MACRO** Existe un conjunto de macros incorporadas al *SQLMenu* que pueden ser llamados desde un menú, como los que se muestran la tabla 8.

Menú	Nº Item	Tipo de comando	Línea de comando
MACRO	1	6	NEWUSER;
	2	6	NEWAPL;
	3	6	MAINMENU;ASSIGN DK=A;;

Tabla 8. Ejecución de macros que pueden ser llamadas.

Los macros tienen un procedimiento equivalente en *SQLMenu* y viceversa. En los manuales de Oracle existe una tabla de correspondencia entre *Macro-Procedimiento-Tecla de Función*. Esta dualidad está dada con el fin de mantener la compatibilidad con versiones anteriores de *SQLMenu*.

Algunas macros tienen argumentos, como la macro *ASSIGN*, cuyo objetivo es asignar un nuevo valor a un parámetro definido por el usuario. La definición y el uso de los parámetros lo veremos más adelante.

### SISTEMA

Los comandos de tipo 2 y 3 son los que se usan para hacer tareas desde el sistema operativo, pasando una línea de comando directamente al sistema operativo para su ejecución, como ya vimos anteriormente para el caso de los reportes. La diferencia consiste en que después de realizar el comando de tipo 2 se retorna al menú inmediatamente, y con el tipo 3 se realiza una pausa antes de retornar al menú, quedando a la espera de que el usuario presione alguna tecla para continuar.

Menú	N° Item	Tipo de comando	Línea de comando
SISTEMA	1	2	CHKDSK &DK
A	2	3	DIR &DK

Tabla 9. Ejecución de tareas del sistema

En este ejemplo se usa el parámetro *DK* para identificar la unidad de disco con la cual se va a trabajar. En la figura 1 se muestra la pantalla de definición de este parámetro.

Parameter Definition		
Parameter	DK	—
Size	2	[ x ] Echo
Prompt	Diskette A: B: o disco (Select Menus )	[ ] Fixed Length [ ] Required [ ] Upper Case
Default Hint	C:	—

Figura 1. Pantalla de definición del parámetro *DK*.

Una vez seleccionado la opción del menú *SISTEMA* que hace referencia al parámetro *DK*, aparecerá en pantalla, para ingresar el valor del parámetro, el siguiente cuadro:

Enter Parameter Value(s)  
Diskette A: B: o disco C:

## Integración desde SQLForms

La integración de módulos, como los casos que acabamos de estudiar, también esta presente en *SQLForms*, desde donde podemos activar los módulos *SQLReport* y *SQLPlus* a través del sistema operativo.

En artículos anteriores estudiamos por separado *SQLForms* y *SQLReport*, ahora pasaremos a ver como se realiza la integración de estos dos módulos. Las definiciones a realizar para esta integración son las siguientes:

En *SQLReport*:

- Parámetros y sus valores por defecto. Esta es la vía que tenemos para pasar los datos necesarios, desde *SQLForms*, en el momento de la ejecución del reporte.

En *SQLForms*:

- Campos de la forma donde se van a ingresar los valores a pasar como parámetros al reporte.
- Procedimiento que invoca a *SQLReport* con sus correspondientes parámetros. Para esto se usa la orden *HOST*, procedimiento del sistema que tiene como función enviar un comando al sistema operativo, en este caso *RUNREP*, garantizando el paso de variables a través de los parámetros y el retorno automático a *SQLForms*.

Veamos primero la definición de los parámetros en *SQLReport*, cuyo objetivo es controlar diferentes aspectos del reporte, como son: dispositivo de salida, tipo de impresora, número de copias y datos específicos del usuario. Para ello retomaremos el ejemplo *VENTAS*, diseñado en nuestro artículo anterior *ORACLE BASICO V* (ver **Algoritmo** N°13): *Reportes con SQLReport*. En la tabla 10 podemos ver los parámetros del sistema y del usuario con sus respectivos significados.

	Nombre	Tipo	Ancho	Valor	Identificación
SISTEMA	DESTYPE	CHAR	80	Printer	Tipo de salida: Screen, File o Printer
	DESNOME	CHAR	80	PRN	Nombre de Dispositivo o archivo
	DESFORM	CHAR	80	dflt	Especifica tipo de

AT					formato de la impresora para sus códigos de control.
	COPIES	NUM	2	1	Número de Copias
USUARI O	XCOD	NUM	10	2	Código Cliente
	XDIAS	NUM	5	30	Cantidad de Días

Tabla 10. Parámetros del sistema y del usuario.

Ahora crearemos, en *SQLForms*, el procedimiento llamado *REPORTE*, cuyo objetivo es realizar el reporte *VENTAS*. (Véase el fuente 2)

```

/* Fuente 2 */
PROCEDURE REPORTE IS
BEGIN
    DECLARE
        /*
        Variables locales de memoria, de tipo carácter, cuyo
        objetivo es cargar los valores de los campos
        numéricos
        de la forma.
        */
        XC CHAR(10);
        XD CHAR(5);
        BEGIN
            /*
            Asignación de los valores de los campos de la forma
            YCOD y YDIAS a las variables locales con su
            correspondiente
            conversión a carácter.
            */
            XC:=TO_CHAR(:YCOD);
            XD:=TO_CHAR(:YDIAS);
            /*
            Llamada a SQLReport por medio de la orden HOST,
            con la correspondiente cadena de caracteres
            de definición de parámetros.
            */
            HOST ('RUNREP REPORT=VENTAS
                USERID=EIDOS/ESPAÑA
                PARAMFORM=NO
                DESTYPE=PRINTER
                DESNAME=PRN
                COPIES=3

                XCOD='||XC||
                ' XDIAS='||XD

            );
        END;
    END;
END;

```

Donde:



*REPORT=VENTAS* Indica que el reporte a ejecutar esta en el archivo *VENTAS.rep*, donde quedó registrado el resultado de la compilación efectuada desde *SQLReport*.

*USERID=EIDOS/ESPAÑA* Identifica el nombre del usuario y su contraseña.

*PARAMFORM=NO* Indica que los parámetros se pasan directamente en la línea de comando, de lo contrario se solicitaría su ingreso por pantalla en el momento de la ejecución.

*DESTYPE=PRINTER* Salida por impresora.

*DESNAME=PRN* Nombre de la impresora.

*COPIES=3* Cantidad de copias.

En esta integración no existe límite en la cantidad de parámetros a transferir, existiendo la posibilidad de construir un archivo de parámetros. Por ejemplo, si escribimos los parámetros en el archivo *ventas.para*, la conexión con *SQLReport* desde el procedimiento *REPORTE* sería de la siguiente forma:

```
HOST    ('RUNREP REPORT=VENTAS
          CMDFILE=VENTAS . PAR'
        );
```

La integración de *SQLForms* con *SQLPlus* cumple las mismas reglas detalladas anteriormente para *SQLReport*, sólo varía la sintaxis en la orden *HOST* ya que ésta corresponde a las especificaciones del comando *SQLPLUS* como se muestra a continuación:

```
HOST    ('SQLPLUS EIDOS/ESPAÑA @LISTADO ' ||
          :F1 |
          :F2 |
          :F2
        );
```

Donde:

*EIDOS/ESPAÑA* Identificación de usuario y su contraseña.

*@LISTADO* Archivo que contiene los comandos SQL a ejecutar.

:F1 y :F2

Campos de la forma, de tipo carácter, cuyos valores se pasa como parámetros al *SQLPlus* en el momento de la ejecución.

Con este artículo llegamos al final de la serie sobre *Oracle Básico*, donde hemos analizado sus aspectos básicos, así como sus rasgos más significativos. Espero haber logrado, por un lado, introducir al lector en el mundo de bases de datos relacionales y *SQL* y, en el caso de aquellos que ya desarrollan aplicaciones en este entorno, impulsarlos a seguir avanzando en el estudio de Oracle.

## 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.  
1999