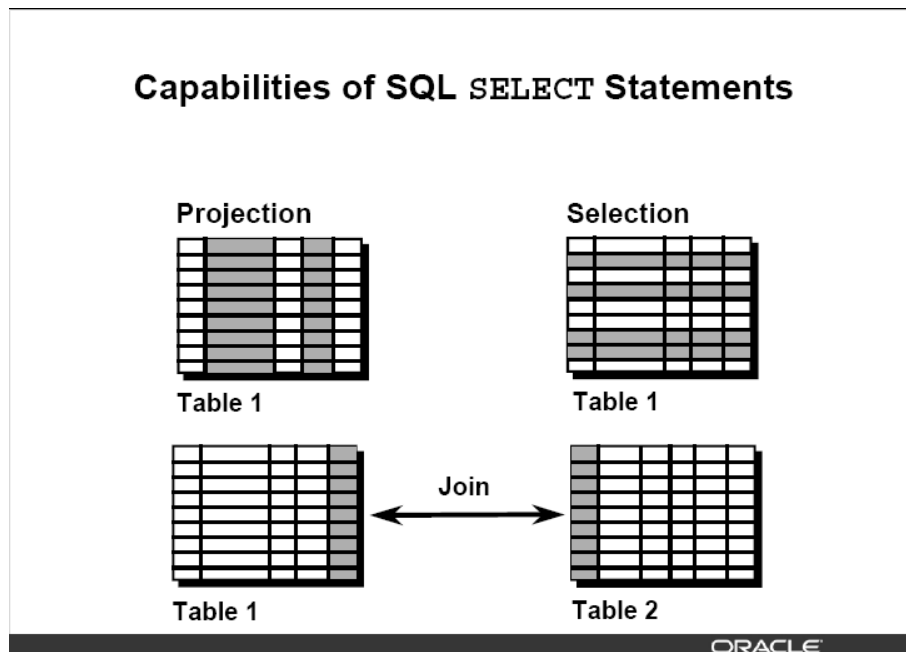


1.- Escribiendo sentencias Básicas SQL SELECT

Objetivos del capítulo

Después de completar este capítulo conocerá lo siguiente:

- Capacidades de la sentencia SELECT de SQL
- Ejecutar sentencias SELECT básicas
- Diferenciar entre las sentencias SQL y comandos de iSQL* plus



Capacidades de la sentencia SELECT de SQL

Una sentencia SELECT recupera información de una base de datos. Usando una sentencia SELECT, se puede realizar lo siguiente:

- **Proyección:** Se puede utilizar la capacidad de proyección en SQL para indicar las columnas de una tabla que se desean regresar por medio de una consulta. Se pueden seleccionar unos cuantos o todas las columnas de la tabla requerida.
- **Selección:** Se puede usar la capacidad de selección en SQL para indicar las filas de la tabla que requerimos por medio de una consulta. Se pueden usar diversos criterios para restringir las filas a visualizar.
- **Asociación o relación (Joining):** Se puede utilizar esta capacidad en SQL para obtener conjuntamente los datos almacenados en diferentes tablas por medio de la creación de una liga entre ambas tablas. Se revisará mas de este tema en los siguientes capítulos.

Basic SELECT Statement

```
SELECT  *|{ [DISTINCT] column|expression [alias],...}  
FROM    table;
```

- **SELECT** identifies *what* columns
- **FROM** identifies *which* table

ORACLE

Sentencia *SELECT* básica

Una sentencia **SELECT** debe contener lo siguiente:

- Una cláusula **SELECT**, que especifica la lista de columnas que serán desplegadas.
- Una cláusula **FROM**, que especifica la tabla que contiene las columnas identificadas en la cláusula **SELECT**.

En la sintaxis:

SELECT

*

DISTINCT

Column | *expresión*

alias

FROM *table*

Es una lista de una o mas columnas

Selecciona todas las columnas

Suprime resultados duplicados

Selecciona la columna o expresión nombrada

Da un nombre de encabezado diferente a la columna seleccionada

Especifica la tabla que contiene las columnas seleccionadas

Selecting All Columns

```
SELECT *  
FROM departments;
```

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Shipping	124	1500
40	IT	103	1400
50	Sales	149	2500
60	Executive	100	1700
70	Accounting	205	1700
80	Contracting		1700

8 rows selected.

ORACLE

Seleccionando todas las columnas con todas las filas

Se puede desplegar los datos de todas las columnas de una tabla poniendo un asterisco (*) después de la palabra reservada SELECT. En la imagen anterior, la tabla departments tiene 4 columnas: DEPARTMENT_ID, DEPARTMENT_NAME, MANAGER_ID y LOCATION_ID. Esta tabla contiene 8 filas, una por cada departamento.

También se pueden mostrar todas las columnas de esta tabla, listando el nombre de cada una de las columnas después de la palabra reservada SELECT. Por ejemplo, la siguiente sentencia SQL despliega todas las columnas de todos los departamentos de la misma forma que en la imagen anterior.

```
SELECT department_id, department_name, manager_id, location_id  
FROM departments;
```

Selecting Specific Columns

```
SELECT department_id, location_id  
FROM departments;
```

DEPARTMENT_ID	LOCATION_ID
10	1700
20	1800
50	1500
60	1400
80	2500
90	1700
110	1700
190	1700

8 rows selected.

ORACLE

Seleccionando columnas específicas con todas las filas

También se puede usar la sentencia SELECT para desplegar columnas específicas de una tabla, especificando el nombre de cada columna separada por comas. En la imagen anterior se despliegan las columnas DEPARTMENT_ID y LOCATION_ID de la tabla departments.

En la cláusula SELECT se especifican las columnas y el orden que se desea tener. Por ejemplo, para mostrar la columna LOCATION_ID antes que DEPARTMENT_ID se utiliza la siguiente sentencia:

```
SELECT location_id, department_id  
FROM departments;
```

Escribiendo sentencias SQL

Usando las siguientes reglas y normas, se pueden construir sentencias válidas con una fácil lectura y edición:

- Las sentencias SQL no distinguen entre mayúsculas y minúsculas, a menos que sea indicado
- Las sentencias SQL pueden ser escritas en una o más líneas
- Las palabras reservadas no pueden ser abreviadas o cortadas entre líneas
- Las cláusulas son usualmente escritas en líneas separadas para una mejor legibilidad y edición
- La *indentación* puede ser utilizado para una mejor lectura
- Las palabras reservadas son típicamente escritas en mayúsculas y todo lo demás como nombre de tablas y columnas en minúsculas

Ejecutando sentencias SQL

Con iSQL*Plus, oprima el botón ejecutar para correr el comando o comandos de una ventana de edición.

Encabezados de columnas por defecto

A continuación se describe el formato por defecto aplicado a los encabezados:

En iSQL*Plus:

- Los encabezados se muestran justificados al centro de la columna
- Los encabezados se muestran en mayúsculas

En SQL*Plus:

- Los encabezados de las columnas de tipo carácter y fecha son justificados a la izquierda.
- Los encabezados de las columnas de tipo numérico son justificados a la derecha
- Por defecto los encabezados se muestran en mayúsculas

Los encabezados de las columnas pueden ser sobrescritos con el uso de alias. Como se vera más adelante.

Arithmetic Expressions	
Create expressions with number and date data by using arithmetic operators.	
Operator	Description
+	Add
-	Subtract
*	Multiply
/	Divide

ORACLE

Expresiones aritméticas

En algunas ocasiones se puede tener la necesidad de modificar la forma en que un dato es desplegado, ejecutar cálculos o ambos escenarios. Esto es posible con el uso de expresiones aritméticas. Una expresión aritmética puede contener nombres de columnas, valores numéricos constantes y operadores aritméticos.

Operadores aritméticos

En la imagen anterior, se muestran los operadores aritméticos disponibles en SQL. Estos pueden ser usados en cualquier cláusula de una sentencia SQL con excepción de la cláusula FROM.

Using Arithmetic Operators		
<pre>SELECT last_name, salary, salary + 300 FROM employees;</pre>		
LAST_NAME	SALARY	SALARY+300
King	24000	24300
Kochhar	17000	17300
De Haan	17000	17300
Hunold	9000	9300
Ernst	6000	6300
...		
Hartstein	13000	13300
Fay	6000	6300
Higgins	12000	12300
Gietz	8300	8600
20 rows selected		
ORACLE		

Usando operadores aritméticos

En el ejemplo de la imagen anterior, se utiliza el operador de adición para calcular un incremento de \$300 en el salario de los empleados y ser desplegado como una nueva columna llamada SALARY + 300.

Note que el cálculo resultante de la columna SALARY + 300 desplegada, no es una nueva columna de la tabla EMPLOYEES. Por defecto el nombre de columnas calculadas toma el nombre del cálculo, en este caso salary+300.

Nota: El servidor de Oracle9i ignora los espacios en blanco antes y después de operadores aritméticos.

Precedencia de los operadores

Si una expresión aritmética contiene más de un operador se siguen las siguientes reglas:

- La multiplicación y la división tienen mayor prioridad que la suma y resta.
- Los operadores de la misma prioridad son evaluados de izquierda a derecha.
- El uso de paréntesis fuerza la evaluación de prioridades e incrementa la claridad en la lectura de las sentencias.

Operator Precedence

```
SELECT last_name, salary, 12*salary+100
FROM employees;
```

LAST_NAME	SALARY	12*SALARY+100
King	24000	288100
Kochhar	17000	204100
De Haan	17000	204100
Hunold	9000	108100
Ernst	6000	72100
...		
Hartstein	13000	156100
Fay	6000	72100
Higgins	12000	144100
Gietz	8300	99700

20 rows selected.

ORACLE

En el ejemplo de la imagen anterior se despliega el nombre, salario y la compensación anual de los empleados. El calculo de la compensación anual se obtiene multiplicando el salario mensual por 12, más una compensación de \$100. Note que la multiplicación es ejecutada antes que la adición.

Nota: El uso de paréntesis consolida el orden estándar de precedencia y provee mejor claridad. Por ejemplo, la expresión de la imagen anterior puede ser escrito como $(12 * \text{salary}) + 100$ sin que se cambie el resultado.

Usando paréntesis

Se pueden sobrescribir las reglas de precedencia con el uso de paréntesis para especificar el orden en el que los operadores son ejecutados.

```
SELECT last_name, salary, 12*(salary+100)
FROM employees;
```

En la sentencia anterior se despliega el nombre, salario y la compensación anual de los empleados. El cálculo de la compensación anual es obtenido mediante la multiplicación del salario más el bono de \$100 multiplicado por 12, puesto que los paréntesis hacen que la suma tome mayor prioridad que la multiplicación.

Defining a Null Value

- A null is a value that is unavailable, unassigned, unknown, or inapplicable.
- A null is not the same as zero or a blank space.

```
SELECT last_name, job_id, salary, commission_pct  
FROM employees;
```

LAST_NAME	JOB_ID	SALARY	COMMISSION_PCT
King	AD_PRES	24000	
Kochhar	AD_VP	17000	
...			
Zlotkey	SA_MAN	10500	.2
Abel	SA_REP	11000	.3
Taylor	SA_REP	8600	.2
...			
Gietz	AC_ACCOUNT	6300	

20 rows selected.

ORACLE

Valores Nulos

Si una fila carece de valor en alguna columna en particular, este valor es llamado *nulo*. Un valor nulo es un valor no disponible, desconocido o inaplicable. Un Nulo no es lo mismo que un cero o un espacio. El cero es un número y el espacio un carácter.

Las columnas de algunos tipos de datos pueden contener valores nulos. Sin embargo, algunas restricciones (constraints) como NOT NULL y PRIMARY KEY, previenen que se utilicen valores nulos para dichas columnas.

En la columna COMMISSION_PCT de la tabla EMPLOYEES, note que solo algunos responsables de ventas pueden obtener comisiones. Otros empleados no son considerados para obtener una comisión. Un valor nulo representa este hecho.

Null Values in Arithmetic Expressions

Arithmetic expressions containing a null value evaluate to null.

```
SELECT last_name, 12*salary*commission_pct  
FROM employees;
```

LAST_NAME	12*SALARY*COMMISSION_PCT
King	
Kochhar	
...	
Zlotkey	25200
Abel	39600
Taylor	20640
...	
Gietz	

20 rows selected.

ORACLE

Si el valor de una columna en una expresión aritmética es nulo, el resultado será nulo. Por ejemplo, si se intenta realizar una división entre cero, se obtiene un error. Sin embargo, si se divide un número con un valor nulo, el resultado es un valor nulo o desconocido.

En el ejemplo de la imagen anterior, el empleado King no tiene comisiones. Puesto que la columna COMMISSION_PCT en una expresión aritmética es nulo, el resultado también lo será.

Definiendo alias a las columnas

Cuando una consulta despliega sus resultados, iSQL*Plus normalmente usa el nombre de la columna seleccionada como encabezado. Este encabezado puede no ser descriptivo y por lo tanto puede dificultar su entendimiento. Se puede cambiar el encabezado de una columna con el uso de *ALIAS*.

Se especifica el alias después de que una columna es listada en la cláusula SELECT usando un espacio como separador. Por defecto, los alias se muestran en mayúsculas. Si un alias contiene espacios o caracteres especiales (como # o \$) o se requiere distinguir las mayúsculas de las minúsculas, se debe encerrar el alias entre dobles comillas (" ").

Using Column Aliases

```
SELECT last_name AS name, commission_pct comm
FROM employees;
```

NAME	COMM
King	
Kochhar	
De Haan	
...	

20 rows selected.

```
SELECT last_name "Name", salary*12 "Annual Salary"
FROM employees;
```

Name	Annual Salary
King	298000
Kochhar	204000
De Haan	204000
...	

20 rows selected.

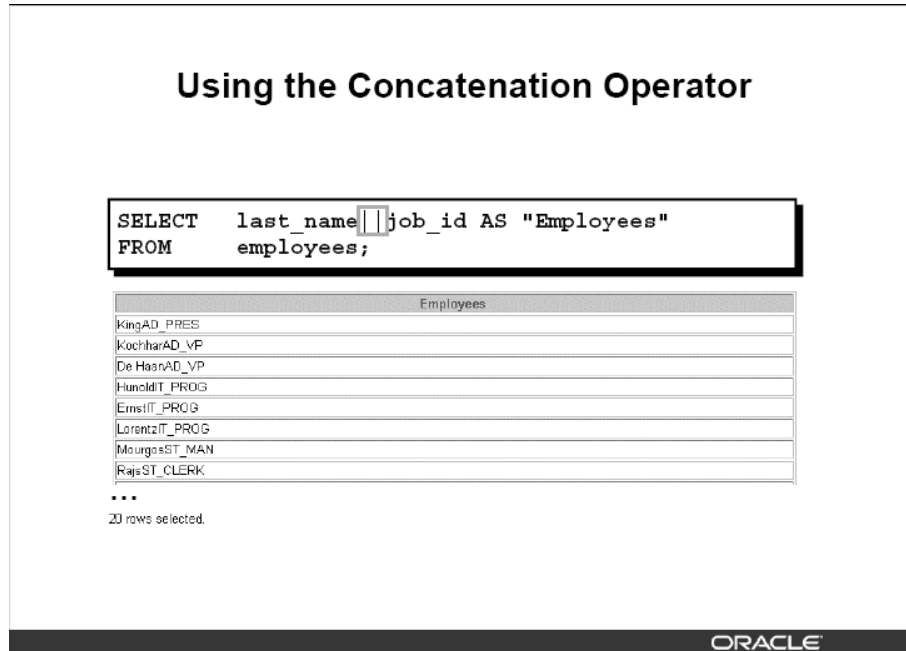
ORACLE

En el primer ejemplo se muestra el apellido y el porcentaje de comisión de todos los empleados. Note que la palabra reservada AS es opcional y es usada antes del alias. El resultado de esta consulta es igual con o sin el uso de la palabra reservada AS. Además, note que la sentencia SQL tiene los alias *name* y *comm* en minúsculas, mientras que el resultado de la consulta muestra los encabezados de las columnas en mayúsculas. Como se ha mencionado, los encabezados de las columnas se muestran en mayúsculas por defecto.

En el segundo ejemplo se despliega el apellido y el salario anual de todos los empleados. Puesto que "Annual Salary" contiene espacios, este debe ser encerrado entre dobles comillas. Note que el encabezado se muestra exactamente igual que el alias.

Operador de Concatenación

Se pueden unir una columna a otra columna, a una expresión aritmética o a un valor constante para crear una expresión del tipo carácter con el uso del operador de concatenación (||). Las columnas de ambos lados del operador de concatenación son combinadas para formar una columna.



En el ejemplo de la imagen anterior, las columnas LAST_NAME y JOB_ID están concatenadas y tienen el alias *Employees*. Note que el apellido y el código de empleo son combinados para formar una sola columna. La palabra reservada AS antes del alias, hace que la cláusula SELECT sea más fácil de leer.

Literales

Un literal es un carácter, número o fecha que es incluido en la lista de una cláusula SELECT y no representa el nombre de una columna o alias. Este es desplegado por cada fila.

Los literales del tipo carácter y fecha deben estar entre comillas simples (' '), para los literales del tipo numérico, no es necesario.

Using Literal Character Strings

```
SELECT last_name || ' is a ' || job_id  
       AS "Employee Details"  
FROM   employees;
```

Employee Details
King is a AD_PRES
Kochhar is a AD_VP
De Haan is a AD_VP
Hunold is a IT_PROG
Ernst is a IT_PROG
Lorentz is a IT_PROG
Mourgos is a ST_MAN
Rajs is a ST_CLERK

...
20 rows selected.

ORACLE

En el ejemplo de la imagen anterior se despliega el apellido y código del puesto para todos los empleados. La columna tiene como encabezado *Employee Details*. Note los espacios entre comillas sencillas en la sentencia SELECT. Estos espacios proporcionan una mejor lectura.

En el siguiente ejemplo el apellido y salario de cada empleado se concatena con un literal para cada fila y proporcionar un mejor significado.

```
SELECT last_name || ': 1 Month salary = ' || salary Monthly  
FROM   employees;
```

MONTHLY
King: 1 Month salary = 24000
Kochhar: 1 Month salary = 17000
De Haan: 1 Month salary = 17000
Hunold: 1 Month salary = 9000
Ernst: 1 Month salary = 6000
Lorentz: 1 Month salary = 4200
Mourgos: 1 Month salary = 5800
Rajs: 1 Month salary = 3500

...
20 rows selected.

Duplicate Rows

The default display of queries is all rows, including duplicate rows.

```
SELECT department_id  
FROM employees;
```

DEPARTMENT_ID	
	90
	90
	90
	60
	60
	60
	50
	50
	50

20 rows selected.

ORACLE

Filas duplicadas

A menos que indiques lo contrario, iSQL*Plus despliega los resultados de una consulta sin eliminar filas duplicadas. En el ejemplo de la imagen anterior, se despliegan todos los números de departamento de la tabla empleados, aunque estos se repitan.

Eliminating Duplicate Rows

Eliminate duplicate rows by using the **DISTINCT** keyword in the **SELECT** clause.

```
SELECT DISTINCT department_id  
FROM employees;
```

DEPARTMENT_ID	
	10
	20
	50
	60
	80
	90
	110

8 rows selected.

ORACLE

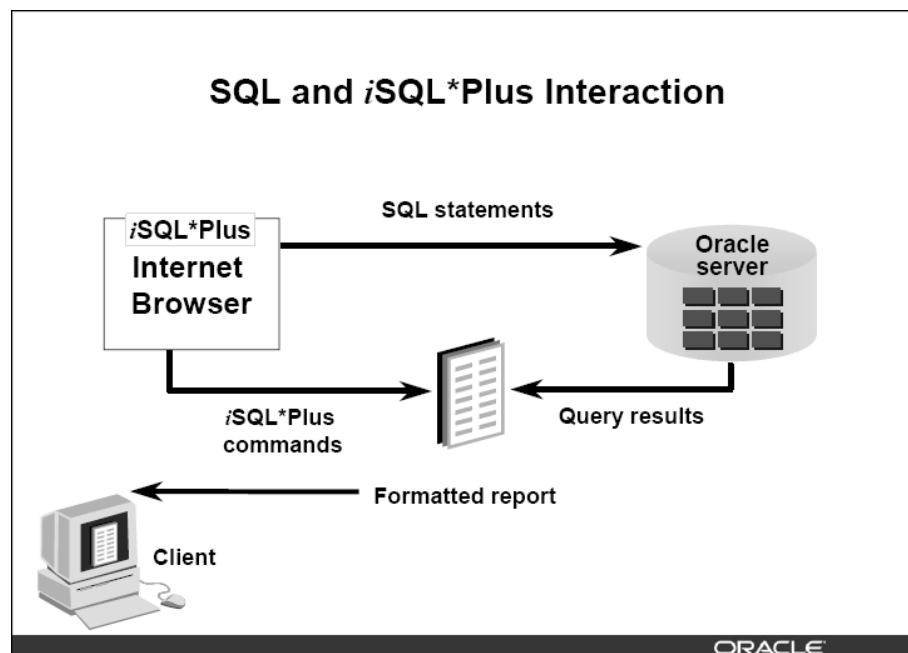
Para eliminar filas duplicadas en el resultado, se incluye la palabra reservada **DISTINCT** en la cláusula **SELECT** inmediatamente después de la palabra reservada **SELECT**. En el ejemplo de la imagen anterior, la tabla **EMPLOYEES** actualmente contiene 20 filas pero de ellas solo 7 números de departamento son únicos.

Se pueden especificar múltiples columnas después de la palabra reservada DISTINCT. El DISTINCT afecta a todas las columnas seleccionadas y el resultado será distinto dependiendo de la combinación de columnas.

```
SELECT DISTINCT department_id, job_id
FROM employees;
```

DEPARTMENT_ID	JOB_ID
10	AD_ASST
20	MK_MAN
20	MK_REP
50	ST_CLERK
50	ST_MAN
60	IT_PROG
...	
	SA_REP

13 rows selected.



SQL e iSQL*Plus

SQL es un lenguaje de comandos para comunicarse con el servidor Oracle desde alguna herramienta o aplicación. El SQL de Oracle contiene muchas adiciones.

iSQL*Plus es una herramienta de Oracle que reconoce y envía sentencias SQL a un servidor Oracle para ser ejecutadas, el cual contiene sus propios comandos.

Características de SQL

- Puede ser utilizado por diversos usuarios, incluyendo aquellos con poca o nula experiencia en programación
- Es un lenguaje no procedural
- Reduce la cantidad de tiempo requerido para crear y mantener sistemas
- Es un lenguaje parecido al inglés

Características de iSQL*Plus

- Utilizado desde un browser
- Acepta sentencias con fines específicos
- Proporciona edición en línea para modificar sentencias SQL
- Controla la configuración del entorno o ambiente
- Estructura el resultado de las consultas en reportes básicos
- Accesa a bases de datos locales y remotas

La siguiente tabla muestra un comparativo entre SQL e iSQL*Plus:

Sentencias SQL	Comandos iSQL*Plus
Es un lenguaje para comunicarse con el servidor de Oracle y recuperar datos	Reconoce sentencias SQL y las envía a un servidor
Esta basado en los estándares del Instituto nacional Americano (ANSI) como un SQL estándar	Es una interfaz de Oracle para ejecutar sentencias SQL
Manipula datos y definiciones de tablas en una base de datos	No permite la manipulación de valores en una base de datos
No tiene un carácter de continuación	Cuenta con un carácter de continuación (-) si el comando es muy largo y ocupa mas de una línea
No pueden ser abreviados	Pueden ser abreviados
Usa funciones para ejecutar algunos formatos	Usa comandos para formatear datos
	Corre en un browser o navegador
	Se utiliza de manera centralizada no tiene que ser implementado en cada computadora

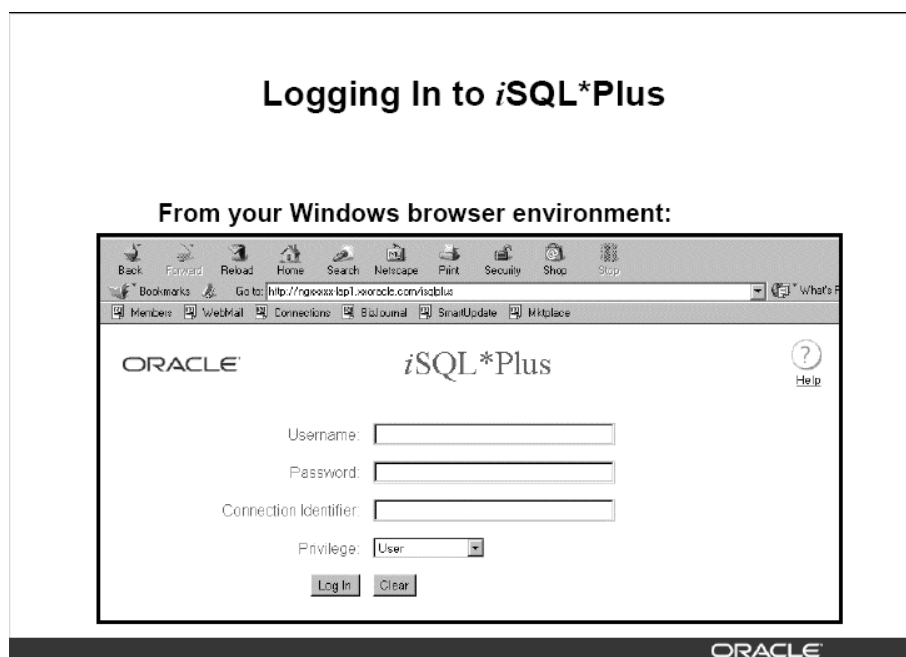
iSQL*Plus

iSQL*Plus es un ambiente en el que se puede realizar lo siguiente:

- Ejecutar sentencias SQL para recuperar, modificar, añadir, y eliminar datos de una base de datos
- Formato, ejecuta cálculos, almacena e imprime los resultados de las consultas en un reporte formateado
- Crea scripts para almacenar sentencias SQL para usos futuros

Los comandos de iSQL*Plus pueden ser divididos en las siguientes categorías principales:

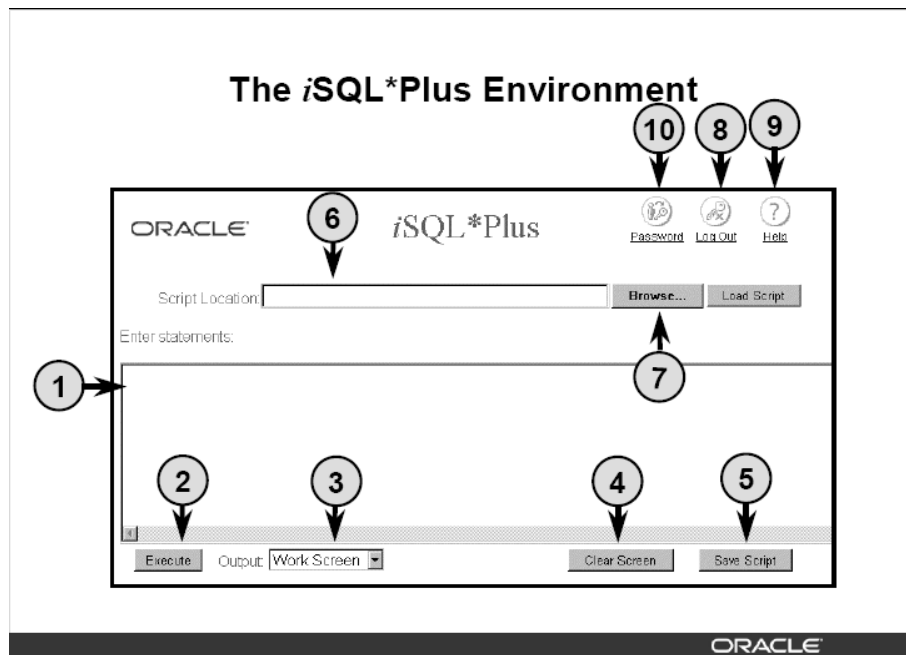
Categoría	Propósito
Ambiente	Afecta el comportamiento general de las sentencias SQL de una sesión
Formato	Los resultados de las consultas se muestran con un formato
Manipulación de archivos	Se pueden almacenar sentencias en archivos de texto (scripts) para ser ejecutadas después
Ejecución	Envía sentencias SQL desde un browser a un servidor de Oracle
Edición	Modifica sentencias SQL en una ventana de edición
Interacción	Permite crear y pasar variables a sentencias SQL, imprimir sus valores e imprimir mensajes en la pantalla
Varios	Cuenta con varios comandos para conectarse a una base de datos, manipular el ambiente iSQL*Plus y mostrar las definiciones de columnas de las tablas de la base de datos



Ingresando al ambiente iSQL*Plus

Para iniciar al ambiente a través de un browser o navegador se realiza:

1. Iniciar un navegador
2. Ingresar la URL del ambiente iSQL*Plus
3. Escribir el usuario y password que identifica la conexión al servidor de Oracle



El ambiente iSQL*Plus

Dentro del navegador, el iSQL*Plus tiene las siguientes áreas:

1. Ventana de edición: En esta área se escriben las sentencias SQL y los comandos iSQL*Plus
2. Botón de ejecución: Al oprimirlo, se ejecuta la sentencia o comando escrita en la ventana de edición
3. Opción de salida: Por defecto se trabaja en la pantalla, cuando se muestran los resultados de las sentencias SQL debajo de la ventana de edición. Las otras opciones son Archivo o Ventana. La opción archivo almacena el resultado en el archivo especificado. La opción ventana muestra el resultado en otra ventana de Windows
4. Botón para limpiar pantalla: Limpia el texto de la ventana de edición
5. Botón para guardar el script: Almacena el contenido de la ventana de edición en un archivo
6. Localización del script: Identifica el nombre y localización del script que se desea ejecutar
7. Botón del navegador: Se utiliza para buscar el script usando el diálogo para abrir archivos de Windows
8. Icono para salir: Finaliza la sesión de iSQL*Plus y regresa el navegador a la ventana de inicio de iSQL*Plus
9. Icono de Ayuda: Proporciona acceso a la documentación de ayuda de iSQL*Plus
10. Botón de password: Es usado para cambiar el password del usuario.

Desplegando la estructura de tablas

En iSQL*Plus, se puede desplegar la estructura de una tabla usando el comando DESCRIBE. Este comando muestra el nombre y tipo de dato de las columnas, así como también si debe o no contener datos.

La sintaxis es:

DESC[RIBE] tablename

Donde:

tablename es el nombre de alguna tabla, vista o sinónimo accesible al usuario.

Displaying Table Structure		
<pre>DESCRIBE employees</pre>		
Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(30)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(30)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

ORACLE

En el ejemplo de la imagen anterior se muestra la estructura de la tabla DEPARTMENTS.

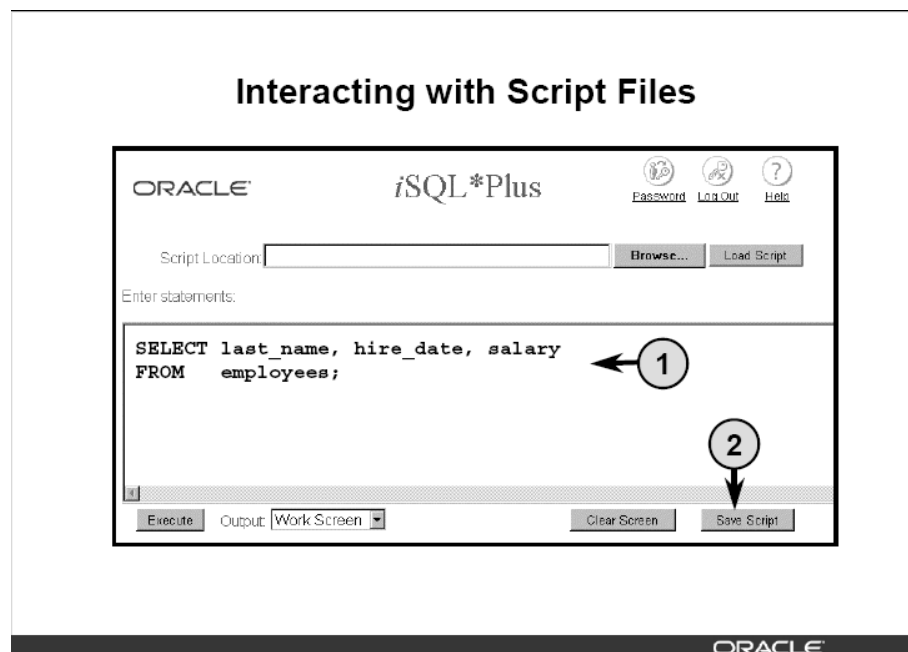
Donde:

Null? Indica si una columna debe contener datos; NOT NULL indica que la columna obligatoriamente debe contener datos

Type Muestra el tipo de dato de la columna

Los tipos de datos se describen a continuación:

Tipo de dato	Descripción
NUMBER(<i>p,s</i>)	Valor numérico que tiene un número máximo de <i>p</i> dígitos, con <i>s</i> dígitos a la derecha del punto decimal
VARCHAR2(<i>s</i>)	Cadena de longitud variable con un tamaño máximo representado por <i>s</i>
DATE	Fecha y hora en el rango del 1 de Enero de 4712 A.C. al 31 de Diciembre de 9999 D.C.
CHAR(<i>s</i>)	Cadena de longitud fija con un tamaño representado por <i>s</i>

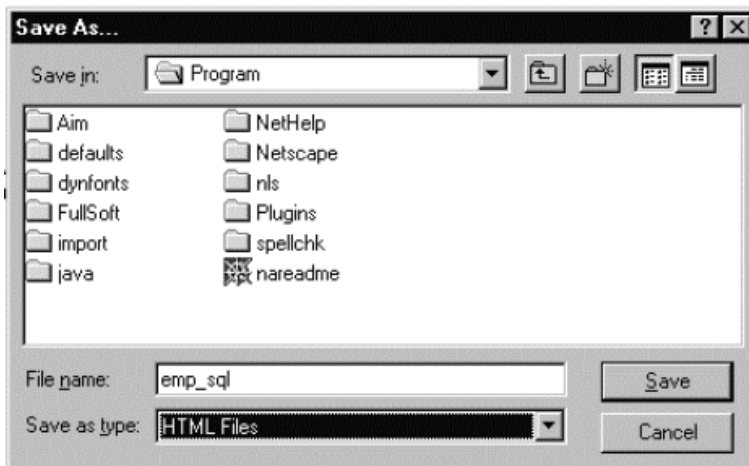


Interactuando con scripts

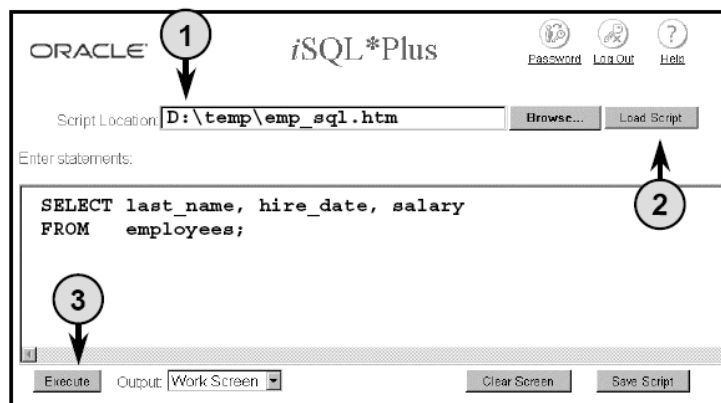
Colocando sentencias y comando en un Script

Se pueden almacenar comandos y sentencias desde la ventana de edición de iSQL*Plus en un script de la siguiente manera:

1. Escriba las sentencias SQL en la ventana de edición de iSQL*Plus
2. Oprima el botón Guardar script (save script). Este abre un diálogo de Windows para guardar archivos. Se indica el nombre del archivo que por defecto tiene la extensión *.html* .La extensión puede ser cambiada a *.sql*



Interacting with Script Files

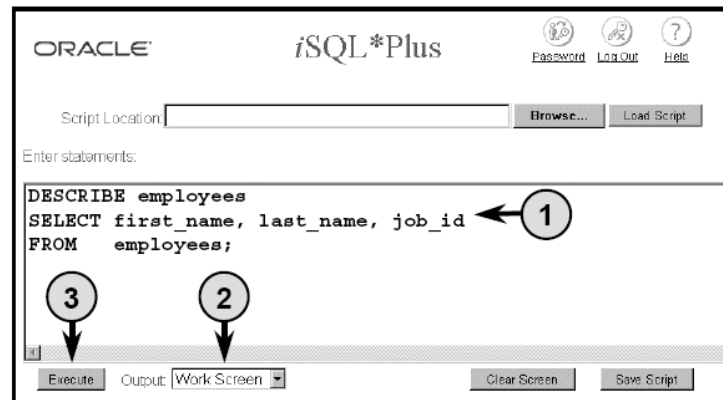


*Usando sentencias y comandos desde un script en iSQL*Plus*

Se pueden utilizar sentencias y comandos previamente almacenados en un script desde iSQL*Plus de la siguiente manera:

1. Escribiendo el nombre y localización del script o utilizando el botón de búsqueda (browse) para localizar el archivo y su ruta
2. Oprimir el botón de carga de scripts (Load script). Este botón lee el contenido del script y lo muestra en la ventana de edición
3. Oprimir el botón ejecutar (execute) para correr el contenido de la ventana de edición

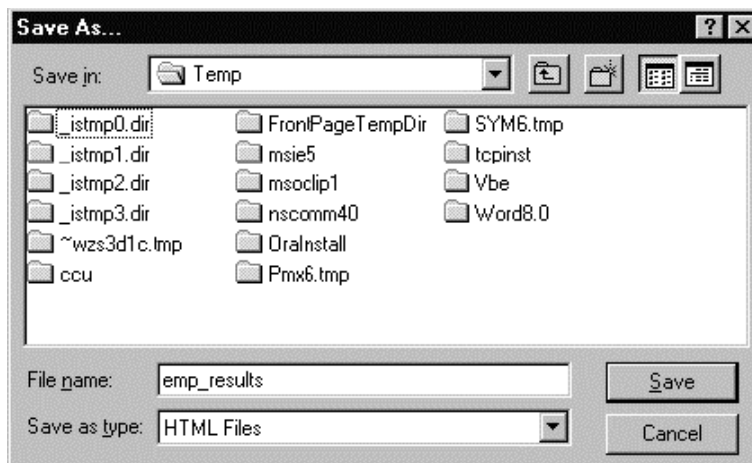
Interacting with Script Files



Guardando en un archivo el resultado

Se puede guardar el resultado generado de una sentencia SQL o comando iSQL*Plus en un archivo de la siguiente forma:

1. Se escriben las sentencias o comandos en la ventana de edición
2. Se cambia la opción de salida a Guardar (save)
3. Se ejecuta el contenido de la ventana de edición. Esta acción abre el diálogo de Windows para guardar archivos. Se identifica el nombre del archivo, que por defecto es .html, pudiendo ser modificado. El resultado es enviado al archivo especificado.



Resumen

En este capítulo se ha revisado la forma de recuperar datos de una tabla de la base de datos por medio de la sentencia SELECT.

```
SELECT * | { [DISTINCT ] column [alias], ... }  
FROM table;
```

Donde:

SELECT	Es la lista de una o mas columnas
*	Selecciona todas las columnas
DISTINCT	Suprime duplicados
Column expresión	Muestra la columna o expresión
Alias	Asigna a las columnas seleccionadas un diferente encabezado
FROM table	Especifica la tabla que contiene las columnas

iSQL*Plus

iSQL*Plus es un ambiente de ejecución que puede ser usado para enviar sentencias SQL a un servidor de base de datos y editar y guardar sentencias SQL. Las sentencias pueden ser ejecutadas desde la línea de comandos o desde un script.

Practica 1

1. Inicie una sesión de iSQL*Plus usando el usuario y password proporcionado por el instructor
2. ¿Los comandos de iSQL*Plus accedan la base de datos? Cierta/Falso
3. ¿La siguiente sentencia SELECT se ejecuta satisfactoriamente? Cierta/Falso

```
SELECT last_name, job_id, salary as Sal
FROM employees;
```

4. ¿La siguiente sentencia SELECT se realiza sin errores? Cierta/Falso

```
SELECT *
FROM job_grades;
```

5. Existen 4 errores en la siguiente sentencia. ¿Puedes identificarlos?

```
SELECT employee_id, last_name
Sal * 12 ANNUAL SALARY
FROM employees;
```

6. Muestre la estructura de la tabla DEPARTMENTS y ejecute una sentencia SELECT para mostrar todos los datos de la tabla.

Name	Null?	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting		1700

7. Muestre la estructura de la tabla EMPLOYEES. Genere una consulta que despliegue las columnas apellidos, clave del puesto, fecha de contratación y número de empleado, mostrando como primera columna el número de empleado. Proporcione el alias STARTDATE para la columna fecha de contratación. Guarde la sentencia SQL en un archivo llamado lab1_7.sql

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

8. Ejecute la consulta del archivo lab1_7.sql
9. Genere una consulta que muestre de forma única las claves de puestos de la tabla empleados

JOB_ID
AC_ACCOUNT
AC_MGR
AD_ASST
AD_PRES
AD_VP
IT_PROG
MK_MAN
MK_REP
SA_MAN
SA_REP
ST_CLERK
ST_MAN

Si aun se cuenta con tiempo, realice los siguientes ejercicios

10. Copia la sentencia del archivo lab1_7.sql en la ventana de edición de iSQL*Plus. Nombre los encabezados de las columnas como Emp#, Employee, Job y Hire Date, respectivamente. Ejecute la consulta nuevamente.

Emp #	Employee	Job	Hire Date
100	King	AD_PRES	17-JUN-87
101	Kochhar	AD_VP	21-SEP-89
102	De Haan	AD_VP	13-JAN-93
103	Hunold	IT_PROG	03-JAN-90
104	Ernst	IT_PROG	21-MAY-91
107	Lorentz	IT_PROG	07-FEB-99
124	Mourgos	ST_MAN	16-NOV-99
141	Rajs	ST_CLERK	17-OCT-95
142	Davies	ST_CLERK	29-JAN-97
143	Matos	ST_CLERK	15-MAR-98
144	Vargas	ST_CLERK	09-JUL-98
■ ■ ■			
206	Gietz	AC_ACCOUNT	07-JUN-94

11. Despliegue el apellido concatenado con el código de puesto, separado por una coma y un espacio. El encabezado debe ser Employee and Title

Employee and Title
King, AD_PRES
Kochhar, AD_VP
De Haan, AD_VP
Hunold, IT_PROG
Ernst, IT_PROG
Lorentz, IT_PROG
Mourgos, ST_MAN
Rajs, ST_CLERK
Davies, ST_CLERK
■ ■ ■
Gietz, AC_ACCOUNT

12. Genere una consulta que despliegue todos los datos de la tabla EMPLOYEES. Separe cada columna con una coma y muéstrela como una sola columna con el encabezado THE_OUTPUT.

THE_OUTPUT													
100	Steven	King	SKING	515.123.4567	AD_PRES	,	17-JUN-87	24000	,	90			
101	Neena	Kochhar	NKOCHHAR	515.123.4568	AD_VP	,	21-SEP-89	17000	,	90			
102	Lex	De Haan	LDEHAAN	515.123.4569	AD_VP	,	13-JAN-93	17000	,	90			
103	Alexander	Hunold	AHUNOLD	590.423.4567	IT_PROG	,	03-JAN-90	9000	,	60			
104	Bruce	Ernst	BERNST	590.423.4568	IT_PROG	,	21-MAY-91	6000	,	60			
107	Diana	Lorentz	DLORENTZ	590.423.5567	IT_PROG	,	07-FEB-99	4200	,	60			
124	Kevin	Mourgos	KMOURGOS	650.123.5234	ST_MAN	,	16-NOV-99	5800	,	50			
141	Trenna	Rajs	TRAJS	650.121.8009	ST_CLERK	,	17-OCT-95	3500	,	50			
142	Curtis	Davies	CDAVIES	650.121.2994	ST_CLERK	,	29-JAN-97	3100	,	50			
143	Randall	Matos	RMATOS	650.121.2874	ST_CLERK	,	15-MAR-98	2600	,	50			
144	Peter	Vargas	PVARGAS	650.121.2004	ST_CLERK	,	09-JUL-98	2500	,	50			
■ ■ ■													
206	William	Gietz	WGIETZ	515.123.8181	AC_ACCOUNT	,	07-JUN-94	8300	,	110			