



1

Escritura de Sentencias SQL SELECT Básicas

ORACLE®
DATABASE

11^g

Objetivos

Al finalizar esta lección, debería estar capacitado para:

- **Enumerar las capacidades de las sentencias SQL
SELECT**
- **Ejecutar una sentencia SELECT básica**
- **Diferenciar entre sentencias SQL y comandos
/SQL*Plus**

Capacidades de la Sentencias SQL SELECT

Proyección

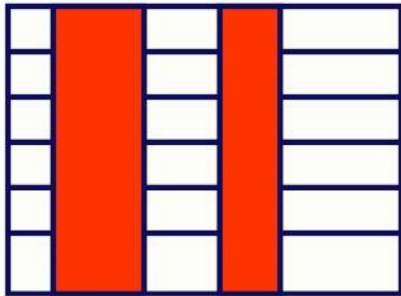


Tabla 1

Selección

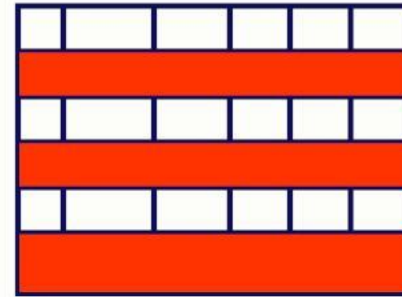


Tabla 1

Unión

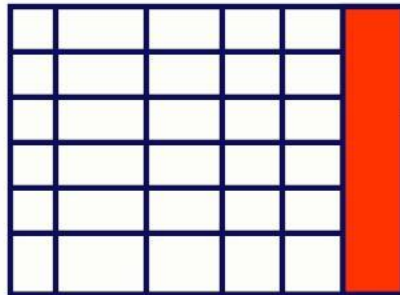


Tabla 1

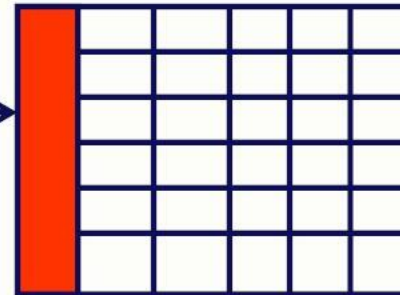


Tabla 2

Sentencias SELECT Básicas

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

- **SELECT** identifica las columnas *what*
- **FROM** identifica la tabla *which*

Selección de Todas las Columnas

SELECT

*

FROM

departments ;

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500
90	Executive	100	1700
100	Finance	108	1700

Selección de Columnas Específicas

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

DEPARTMENT_ID	LOCATION_ID
10	1700
20	1800
30	1700
40	2400
50	1500
60	1400
70	2700
80	2500
90	1700
100	1700

Escritura de Sentencias SQL

- Las sentencias SQL no son sensibles a mayúsculas/minúsculas.
- Las sentencias SQL pueden ocupar una o más líneas.
- Las palabras clave no se pueden abreviar ni dividir entre líneas.
- Las cláusulas suelen estar colocadas en líneas.
- La sangría se utiliza para mejorar la legibilidad.

Expresiones Aritméticas

Cree expresiones con datos de número y fecha mediante operadores aritméticos.

Operador	Descripción
+	Suma
-	Resta
*	Multiplicación
/	División

Uso de Operadores Aritméticos

```
SELECT    last_name, salary, salary + 300
FROM      employees;
```

LAST_NAME	SALARY	SALARY+300
King	24000	24300
Kochhar	17000	17300
De Haan	17000	17300
Hunold	9000	9300
Ernst	6000	6300
Austin	4800	5100
Pataballa	4800	5100
Lorentz	4200	4500
Greenberg	2661,12	2961,12
Faviet	19906,56	20206,56
Chen	18144	18444
Sciarra	17038,08	17338,08

Prioridad de Operador



- La multiplicación y la división tienen prioridad sobre la suma y la resta.
- Los operadores de idéntica prioridad se evalúan de izquierda a derecha.
- Los paréntesis se utilizan para forzar evaluaciones prioritarias y para clarificar sentencias.

Prioridad de Operador

```
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
Austin	4800	57700
Pataballa	4800	57700
Lorentz	4200	50500
Greenberg	2661,12	32033,44
Faviet	19906,56	238978,72
Chen	18144	217828

Uso de Paréntesis

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

LAST_NAME	SALARY	12*(SALARY+100)
King	24000	289200
Kochhar	17000	205200
De Haan	17000	205200
Hunold	9000	109200
Ernst	6000	73200
Austin	4800	58800
Pataballa	4800	58800
Lorentz	4200	51600
Greenberg	2661,12	33133,44
Faviet	19906,56	240078,72

Definición de un Valor Nulo

- Un valor nulo es un valor no disponible, no asignado, desconocido, o no aplicable.
- Un valor nulo no es lo mismo que cero ni que un espacio en blanco.

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

LAST_NAME	JOB_ID	SALARY	COMMISSION_PCT
Davies	ST_CLERK	3100	
Matos	ST_CLERK	2600	
Vargas	ST_CLERK	2500	
Russell	SA_MAN	14000	,4
Partners	SA_MAN	13500	,3
Errazuriz	SA_MAN	12000	,3
Cambrault	SA_MAN	11000	,3
Zlotkey	SA_MAN	10500	,2
Tucker	SA_REP	10000	,3
Bernstein	SA_REP	9500	,25
Hall	SA_REP	9000	,25

Valores Nulos en Expresiones Aritméticas

Las expresiones aritméticas que contienen un valor nulo tienen una evaluación nula.

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

LAST_NAME	SALARY	COMMISSION_PCT	12*SALARY*COMMISSION_PCT
Davies	3100		
Matos	2600		
Vargas	2500		
Russell	14000	.4	67200
Partners	13500	.3	48600
Errazuriz	12000	.3	43200
Cambrault	11000	.3	39600
Zlotkey	10500	.2	25200
Tucker	10000	.3	36000

Definición de una Alias de Columna

Un alias de columna:

- **Cambia el nombre de una cabecera de columna.**
- **Resulta útil con cálculos.**
- **Se sitúa inmediatamente después del nombre de la columna, también puede existir la palabra clave opcional `AS` entre el nombre de la columna y el alias.**
- **Requiere comillas dobles si contiene espacios, caracteres especiales o si es sensible a mayúsculas/minúsculas.**

Uso de Alias de Columna

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

NOMBRE	COMM
King	
Kochhar	

```
SELECT    last_name "Nombre", salary*12 "Salario Anual"
FROM      employees;
```

Nombre	Salario Anual
King	288000
Kochhar	204000
De Haan	204000
Hunold	108000

Operador de Concatenación

Un operador de concatenación:

- **Concatena columnas o cadenas de caracteres a otras columnas.**
- **Está representado por dos barras verticales (||).**
- **Crea una columna resultante que es una expresión de caracteres.**

Uso del Operador de Concatenación

```
SELECT    last_name || job_id AS "Empleados"  
FROM      employees;
```

Empleados
KingAD_PRES
KochharAD_VP
De HaanAD_VP
HunoldIT_PROG
ErnstIT_PROG
AustinIT_PROG

Cadenas de Caracteres Literales

- Un literal es un caracter, un número o una fecha incluida en la lista `SELECT`.
- Los valores literales de caracteres y fecha se deben escribir entre comillas simples.
- Cada cadena de caracteres tiene una salida para cada fila devuelta.

Uso de Cadenas de Caracteres Literales

```
SELECT      last_name || ' is a ' || job_id
            AS "Detalle de Empleados"
FROM        Employees;
```

Detalle de Empleados
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
Austin is a IT_PROG
Pataballa is a IT_PROG
Lorentz is a IT_PROG
Greenberg is a FI_MGR
Faviet is a FI_ACCOUNT
Chen is a FI_ACCOUNT

Filas Duplicadas

La visualización por defecto de las consultas son todas las filas, incluidas las filas duplicadas.

```
SELECT    department_id
FROM      employees;
```

DEPARTMENT_ID	
	90
	90
	90
	60
	60
	60
	60
	60
	60
	100
	100

Eliminación de Filas Duplicadas

Elimine filas duplicadas mediante la palabra clave **DISTINCT** de la cláusula **SELECT**.

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

DEPARTMENT_ID	
	10
	20
	30
	40
	50
	60
	70
	80
	90
	100
	110

Visualización de la Estructura de Tabla

Utilice el comando `DESCRIBE` para ver la estructura de una tabla.

```
DESC [RIBE] tablename
```

Visualización de la Estructura de Tabla

```
DESC employees;
```

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

Resumen

En esta lección, ha aprendido a:

- **Escribir una sentencia `SELECT` que:**
 - Devuelva todas las filas y las columnas de una tabla.
 - Devuelva columnas específicas de una tabla.
 - Utilice alias de columna para asignar cabeceras de columna descriptivas.

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

Visión General de la Práctica 1

Esta práctica cubre los siguientes temas:

- **Selección de todos los datos de tablas diferentes**
- **Descripción de la estructura de las tablas**
- **Realización de cálculos aritméticos y especificación de nombres de columna**