Retrieving Data Using the SQL SELECT Statement

Objectives

After completing this lesson, you should be able to do
the following:
• List the capabilities of SQL SELECT statements
• Execute a basic SELECT statement

Selecting All Columns

```
SELECT *
FROM departments;
```

Selecting Specific Columns

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

Writing SQL Statements

- SQL statements are not case-sensitive.
- SQL statements can be on one or more lines.
- Keywords cannot be abbreviated or split across lines.
- Clauses are usually placed on separate lines.
- Indents are used to enhance readability.
- In SQL*plus, you are required to end each SQL statement with a semicolon (;).

Column Heading Defaults

SQL*Plus:

– **Character and Date column headings are left-aligned**

– **Number column headings are right-aligned**

– **Default heading display: Uppercase**

Arithmetic Expressions

Create expressions with number and date data by using arithmetic operators.

Operator	Description
+	Add
-	**Subtract**
*	**Multiply**
/	**Divide**

Using Arithmetic Operators

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

Operator Precedence

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

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

Defining a Null Value

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

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

Null Values
in Arithmetic Expressions

Arithmetic expressions containing a null value evaluate to null.

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

Defining a Column Alias

A column alias:

- Renames a column heading
- Is useful with calculations
- Immediately follows the column name (There can also be the optional AS keyword between the column name and alias.)
- Requires double quotation marks if it contains spaces or special characters or if it is case sensitive

Using Column Aliases

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

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

Concatenation Operator

A concatenation operator:

- Links columns or character strings to other columns
- Is represented by two vertical bars (||)
- Creates a resultant column that is a character expression

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

Literal Character Strings

- A literal is a character, a number, or a date that is included in the SELECT statement.
- Date and character literal values must be enclosed by single quotation marks.
- Each character string is output once for each row returned.

Using Literal Character Strings

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

Duplicate Rows

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

```
SELECT department_id
FROM employees;
```

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

Displaying Table Structure

```
DESC[RIBE] tablename
```

```
DESCRIBE employees
```