

## **EXERCISE-4**

### **Writing Basic SQL SELECT Statements**

#### **OBJECTIVES**

After the completion of this exercise, the students will be able to do the following:

- List the capabilities of SQL SELECT Statement
- Execute a basic SELECT statement

#### **Capabilities of SQL SELECT statement**

A SELECT statement retrieves information from the database. Using a select statement, we can perform

- ✓ Projection: To choose the columns in a table
- ✓ Selection: To choose the rows in a table
- ✓ Joining: To bring together the data that is stored in different tables

#### **Basic SELECT Statement**

##### **Syntax**

```
SELECT *|DISTINCT Column_name| alias  
FROM table_name;
```

##### **NOTE:**

DISTINCT—Suppr

ess the duplicates.

Alias—gives selected columns different headings.

##### **Example: 1**

```
SELECT * FROM departments;
```

##### **Example: 2**

```
SELECT location_id, department_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

#### **Using Arithmetic Expressions**

Basic Arithmetic operators like \*, /, +, - can be used

##### **Example:1**

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

##### **Example:2**

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

The statement is not same as

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

### **Example:3**

SELECT last\_name, job\_id, salary, commission\_pct FROM employees;

### **Example:4**

SELECT last\_name, job\_id, salary, 12\*salary\*commission\_pct FROM employees;

### **Using Column Alias**

- To rename a column heading with or without AS keyword.

#### **Example:1**

SELECT last\_name AS Name  
FROM employees;

#### **Example: 2**

SELECT last\_name "Name" salary\*12 "Annual Salary "  
FROM employees;

### **Concatenation Operator**

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

#### **Example:**

SELECT last\_name||job\_id AS "EMPLOYEES JOB" FROM employees;

### **Using Literal Character String**

- A literal is a character, a number, or a date included in the SELECT list.
- Date and character literal values must be enclosed within single quotation marks.

#### **Example:**

SELECT last\_name||'is a'||job\_id AS "EMPLOYEES JOB" FROM employees;

### **Eliminating Duplicate Rows**

- Using DISTINCT keyword.

#### **Example:**

SELECT DISTINCT department\_id FROM employees;

### **Displaying Table Structure**

- Using DESC keyword.

#### **Syntax**

DESC table\_name;

#### **Example:**

DESC employees;

### **Find the Solution for the following:**

True OR False

1. The following statement executes successfully.

### **Identify the Errors**

SELECT employee\_id, last\_name  
sal\*12 ANNUAL SALARY

```

1  SELECT employee_id, last_name,
2  sal*12 AS "ANNUAL SALARY";

```

FROM employees;

## Queries

2. Show the structure of departments the table. Select all the data from it.

```

CREATE TABLE DEPARTMENTS(
  Dept_id Number(6),
  Dept_name Varchar(20),
  Manager_id Number(6),
  Location_id Number(4)
);

```

Table created.

0.02 seconds

3. Create a query to display the last name, job code, hire date, and employee number for each employee, with employee number appearing first.

```
SELECT employee_id,last_name,first_name,job_id,hire_date FROM EMPLOYEE;
```

EMPLOYEE_ID	LAST_NAME	FIRST_NAME	JOB_ID	HIRE_DATE
1002	Johnson	Mary	HR_REP	7/10/2021
1004	Brown	Linda	FL_ACCOUNT	11/20/2020
1003	Williams	Robert	SA_REP	3/5/2023
1001	Smith	John	IT_PROG	1/15/2022
1005	Davis	James	MK_MAN	9/1/2019

5 rows returned in 0.01 seconds [Download](#)

4. Provide an alias STARTDATE for the hire date.

```
SELECT hire_date AS "START_DATE"
FROM EMPLOYEE;
```

START_DATE
7/10/2021
11/20/2020
3/5/2023
1/15/2022
9/1/2019

5 rows returned in 0.01 seconds [Download](#)

5. Create a query to display unique job codes from the employee table.

```
SELECT UNIQUE job_id
FROM EMPLOYEE;
```

JOB_ID
HR_REP
FL_ACCOUNT
IT_PROG
MK_MAN
SA_REP

5 rows returned in 0.01 seconds [Download](#)

6. Display the last name concatenated with the job ID , separated by a comma and space, and name the column EMPLOYEE and TITLE.

```
SELECT last_name || ', ' || job_id
AS "EMPLOYEE AND TITLE"
FROM EMPLOYEE;
```

EMPLOYEE AND TITLE
Johnson, HR_REP
Brown, FI_ACCOUNT
Williams, SA_REP
Smith, IT_PROG
Davis, MK_MAN

5 rows returned in 0.01 seconds   [Download](#)

7. Create a query to display all the data from the employees table. Separate each column by a comma. Name the column THE\_OUTPUT.

```
select
EMPLOYEE_ID || ', ' ||
FIRST_NAME || ', ' ||
LAST_NAME || ', ' ||
EMAIL || ', ' ||
PHONE_NUMBER || ', ' ||
HIRE_DATE || ', ' ||
JOB_ID || ', ' ||
SALARY || ', ' ||
COMMISSION_PCT || ', ' ||
MANAGER_ID || ', ' ||
DEPARTMENT_ID AS "THE_OUTPUT"
from EMPLOYEE;
```

THE_OUTPUT
1002, Mary, Johnson, MJOHNSON, 9876501234, 7/10/2021, HR_REP, 45000, , 2002, 101
1004, Linda, Brown, LBROWN, 9988776655, 11/20/2020, FI_ACCOUNT, 70000, , 2004, 102
1003, Robert, Williams, RWILLIAMS, 9123456789, 3/5/2023, SA_REP, 50000, .1, 2003, 104
1001, John, Smith, JSMITH, 9876543210, 1/15/2022, IT_PROG, 60000, , 2001, 103
1005, James, Davis, JDAVIS, 9001122334, 9/1/2019, MK_MAN, 80000, , 2005, 105

5 rows returned in 0.01 seconds   [Download](#)

Evaluation Procedure	Marks awarded
Query(5)	
Execution (5)	
Viva(5)	
Total (15)	
Faculty Signature	