### **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

# <u>Capabilities of SQL SELECT statement</u>

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—givesselected columns different headings.

# Example: 1

SELECT\*FROMdepartments;

# Example: 2

SELECTlocation\_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 sued to enhance readability

# **Using Arithmetic Expressions**

BasicArithmetic operators like \*, /, +, -can be used

# Example:1

SELECT last name, salary, salary+300 FROM employees;

#### <u>Example:2</u>

SELECT last name, salary, 12\*salary+100 FROM employees;

The statement is not same as

Sarvesh S - 240701479 - DBMS RECORD

**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 Solutionforthefollowing: True OR heafselowing statement executes successfully.

# **Identify the Errors**

SELECT employee id, last name

```
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)

Dept_name Varchar(20),
Manager_id Number(6),
Dept_name Varchar(20),
Manager_id Number(4)
```

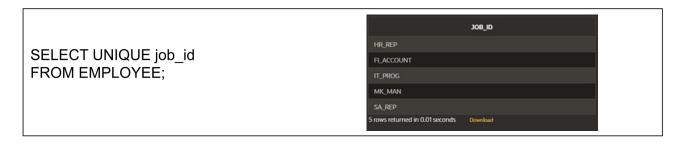
3. Create a query to display the last name, job code, hire date, and employee number for each employee, with employee number appearing first.



4. Provide an alias STARTDATE for the hire date.



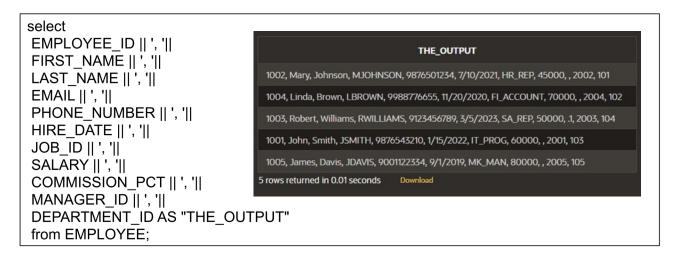
5. Create a query to display unique job codes from the employee table.



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



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



| Evaluation<br>Procedure | Marks awarded |
|-------------------------|---------------|
| Query(5)                |               |
| Execution (5)           |               |
| Viva(5)                 |               |
| Total (15)              |               |
| Faculty Signature       |               |