Kabeer Ahmed
SE-19028

Database Management System
(DBMS – 204)

Experiment # 02

Writing Basic SQL SELECT **Restricting and Sorting Data**

Student Name: Kabeer Ahmed

Roll Number: SE-19028

Maximum Marks	**Performance = 05**	**Viva = 05**	**Total = 10**
Marks Obtained			
Remarks (if any)			

Experiment evaluated by

Instructor Name: Engr. Adiba Jafar

Signature and Date:

Kabeer Ahmed
SE-19028

OUTCOMES

Restricting and Sorting Data Objectives

After completing this lesson, you should be able to do the following:

1. Limit the rows retrieved by a query
2. Sort the rows retrieved by a query

THOERY

Limiting Rows Using a Selection

1. EMPLOYEES “retrieve all employees in department 90”
2. Limiting Rows Using a Selection
3. Limiting the Rows Selected

Restrict the rows returned by using the WHERE clause. The WHERE clause follows the FROM clause.

SELECT *|{[DISTINCT] column expression [alias],...} FROM table [WHERE condition(s)];
SELECT empno, ename, job, deptno FROM emp WHERE deptno = 90;

Character Strings and Dates

• Character strings and date values are enclosed in single quotation marks.
• Character values are case sensitive, and date values are format sensitive.
• The default date format is DD-MON-YR.

SELECT ename, job, deptno FROM emp WHERE ename = 'Goyal';

Comparison Conditions

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

Example

1. WHERE hire_date='01-JAN-95'
2. WHERE salary>=6000
3. WHERE ename='Smith'

NOTE:
An alias cannot be used in the WHERE clause.
The symbol != and ^= can also represent the not equal to condition.

Using Comparison Conditions

SELECT ename, sal FROM emp WHERE sal <= 3000;

Kabeer Ahmed
SE-19028

Using the Comparison Conditions Other Comparison Conditions

Operator	Meaning
BETWEEN	Between two values (inclusive)
...AND...	
IN(set)	Match any of a list of values
LIKE	Match a character pattern
IS NULL	Is a null value

Using the BETWEEN Condition

Use the BETWEEN condition to display rows based on a range of values. (Lower and Upper limit)

SELECT ename, sal FROM emp WHERE sal BETWEEN 2500 AND 3500;

Using the IN Condition

Use the IN membership condition to test for values in a list.

SELECT empno, ename, sal, mgr FROM emp WHERE mgr IN (7902,7698,7839);

Using the LIKE Condition

Use the LIKE condition to perform wildcard searches of valid search string values.

Search conditions can contain either literal ,characters or numbers:

1. % denotes zero or many characters.
2. _ (underscore) denotes one character.

SELECT ename FROM emp WHERE ename LIKE 'S%';

You can combine pattern-matching characters.

SELECT ename FROM emp WHERE ename LIKE '_A%';

You can use the ESCAPE identifier to search for the Actual % and _ symbols.
Where sal LIKE ‘200%’;
Where sal LIKE %200%’;
Where sal LIKE ‘_00%’;
Where sal LIKE ‘2_%_%’;
Where sal LIKE ‘%2’;
Where sal LIKE _2%3’;

Using the NULL Conditions

Test for nulls with the IS NULL operator.

SELECT ename, mgr FROM emp where mgr IS NULL;

Kabeer Ahmed
SE-19028

Logical Conditions

Meaning	Operator
AND	Returns TRUE if *both* component conditions are true
OR	Returns TRUE If *either* component condition is true
NOT	Returns TRUE if the following condition is false

Using the AND Operator

```
SELECT empno, ename, job, sal FROM   emp WHERE  sal >=10000
AND    job LIKE '%MAN%';
```

Using the OR Operator

```
SELECT empno, ename, job, sal FROM   emp WHERE  salary >= 10000
OR     job LIKE '%MAN%';
```

Using the NOT Operator

```
SELECT ename, job FROM   emp
WHERE  job NOT IN ('ANALYST', 'ST_CLERK', 'MANAGER');
```

Rules of Precedence

Order Evaluated Operator

1 Arithmetic operators
2 Concatenation operators
3 Comparison conditions
4 IS [NOT] NULL , LIKE , [NOT] IN
5 [NOT] BETWEEN
6 NOT logical condition
7 AND logical condition
8 OR logical condition

Override rules of precedence by using parentheses.

```
SELECT ename, job, sal  FROM   emp WHERE  job = 'CLERK'
OR     job = 'MANAGER'  AND    sal > 15000;
```

Use parentheses to force priority.

```
SELECT ename, job, sal
FROM   emp
```

```
WHERE  (job = 'CLERK' OR     job = 'MANAGER') AND    salary > 15000;
```

ORDER BY Clause

Syntax

SELECT *expr* FROM *table* [WHERE *condition(s)*] [ORDER BY {*column*, *Expr* } [ASC|DESC]];

1. **Sort rows with the ORDER BY clause**
   ASC : ascending order (the default order)
   DESC : descending order
2. **The ORDER BY clause comes last in the SELECT statement.**

```
SELECT   ename, job, deptno, hiredate FROM     emp ORDER BY hiredate;
```

Sorting in Descending Order

```
SELECT   ename, job, deptno, hiredate FROM     emp ORDER BY hiredate DESC;
```

Sorting by Column Alias

```
SELECT empno, ename, sal*12 annsal FROM   emp ORDER BY annsal;
```

Sorting by Column Aliases

You can use a column alias in the ORDER BY clause. The slide example sorts the data by annual salary.

Sorting by Multiple Columns

1. **The order of ORDER BY list is the order of sort.**

```
SELECT ename, deptno, sal FROM   emp
ORDER BY deptno, sal DESC;
```

2. **You can sort by a column that is not in the SELECT list.**

Kabeer Ahmed
SE-19028

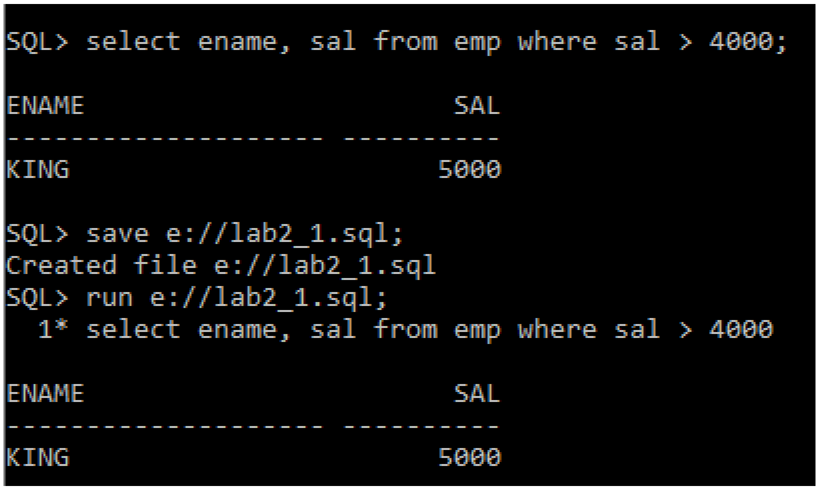
LAB # 02

Writing Basic SQL SELECT Restricting and Sorting Data

Simple Tasks

1. Create a query to display the name and salary of employees earning more than $4000. Place your SQL statement in a text file named lab2_1.sql . Run your query.

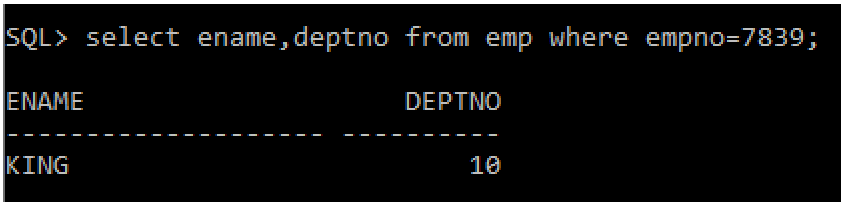
ANS.



```
SQL> select ename, sal from emp where sal > 4000;

ENAME                       SAL
-------------------- ----------
KING                       5000

SQL> save e://lab2_1.sql;
Created file e://lab2_1.sql
SQL> run e://lab2_1.sql;
  1* select ename, sal from emp where sal > 4000

ENAME                       SAL
-------------------- ----------
KING                       5000
```

2. Create a query to display the employee name and department number for employee number 7839.

ANS.

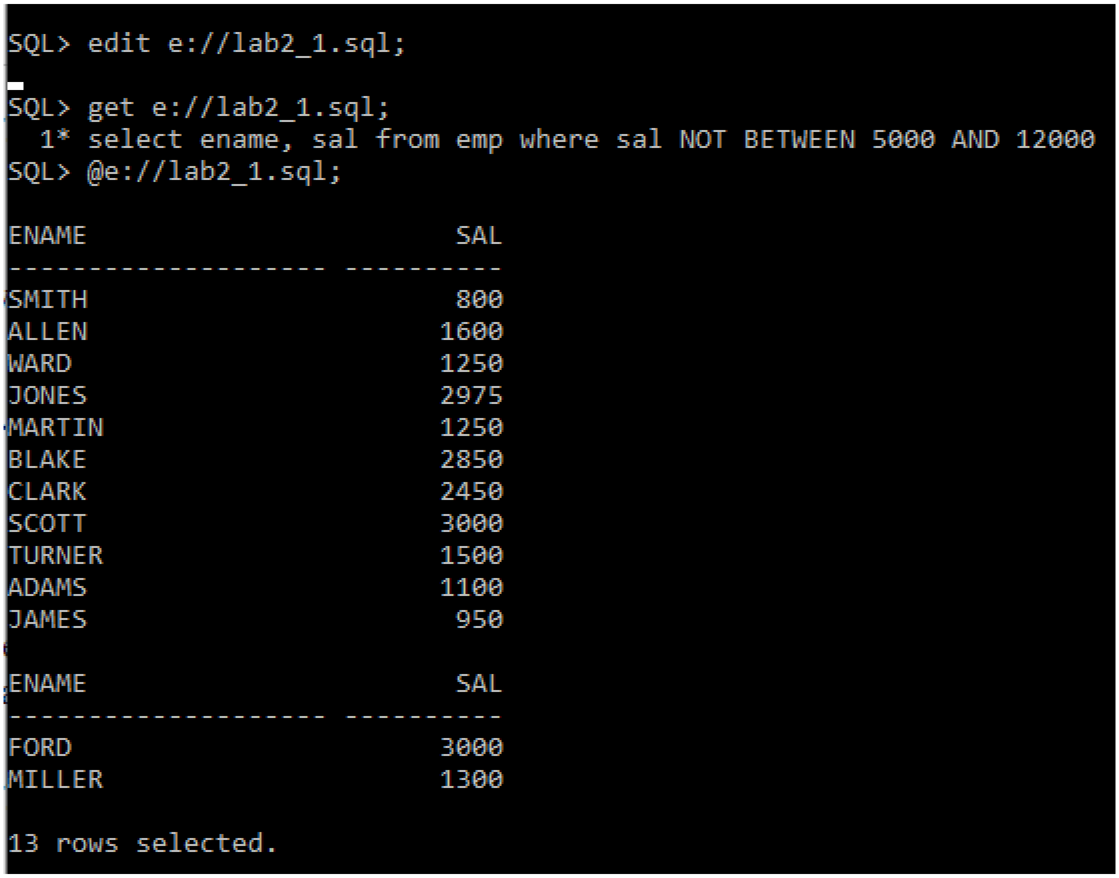


```
SQL> select ename,deptno from emp where empno=7839;

ENAME                    DEPTNO
-------------------- ----------
KING                         10
```

Kabeer Ahmed
SE-19028

3. Modify lab2_1.sql to display the name and salary for all employees whose salary is not in the range of $5,000 and $12,000. Place your SQL statement in a text file named lab2_3.sql

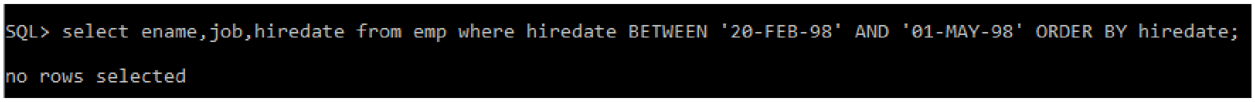
ANS.



```
SQL> edit e://lab2_1.sql;

SQL> get e://lab2_1.sql;
  1* select ename, sal from emp where sal NOT BETWEEN 5000 AND 12000
SQL> @e://lab2_1.sql;

ENAME                       SAL
-------------------- ----------
SMITH                       800
ALLEN                      1600
WARD                       1250
JONES                      2975
MARTIN                     1250
BLAKE                      2850
CLARK                      2450
SCOTT                      3000
TURNER                     1500
ADAMS                      1100
JAMES                       950

ENAME                       SAL
-------------------- ----------
FORD                       3000
MILLER                     1300

13 rows selected.
```

4. Display the employee name, job , and hiredate of employees hired between February 20, 1998, and May 1, 1998. Order the query in ascending order by hiredate.

ANS.

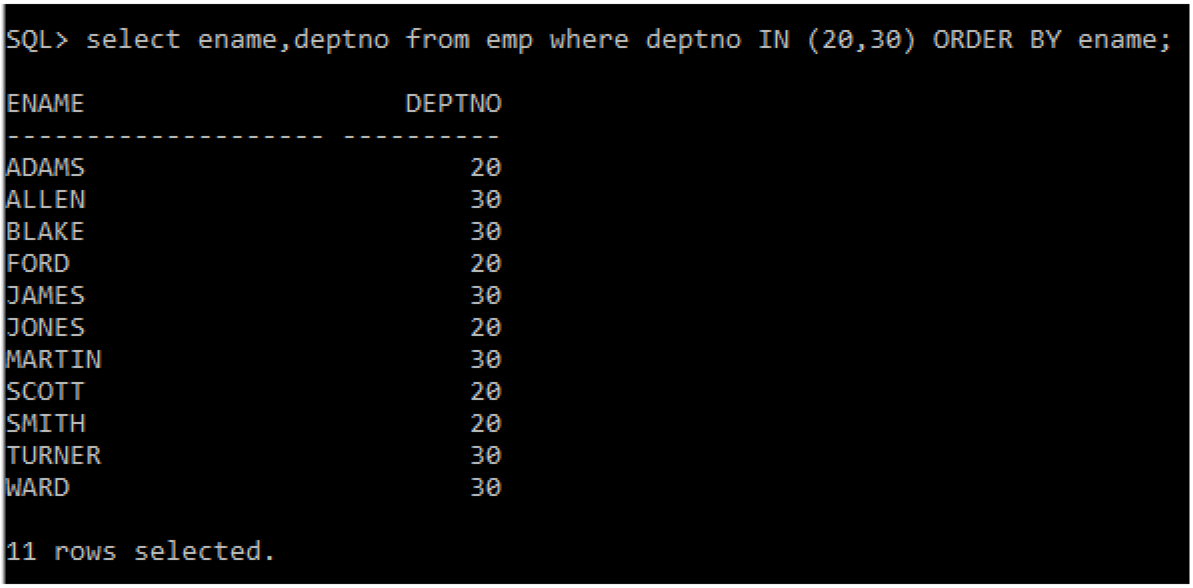


```
SQL> select ename,job,hiredate from emp where hiredate BETWEEN '20-FEB-98' AND '01-MAY-98' ORDER BY hiredate;

no rows selected
```

Kabeer Ahmed
SE-19028

5. Display the name and department number of all employees in departments 20 and 30 in alphabetical order by name.

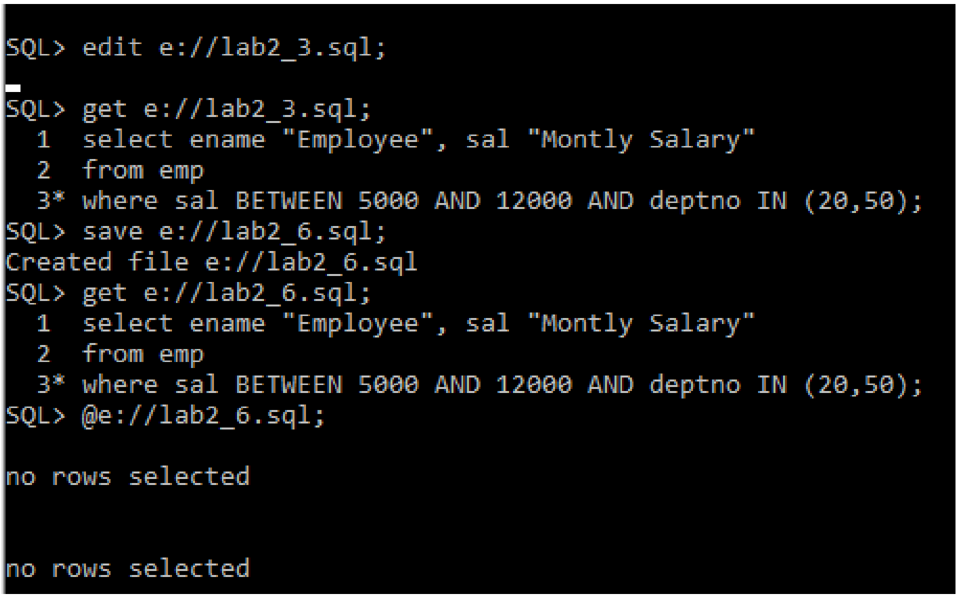
ANS.



```
SQL> select ename,deptno from emp where deptno IN (20,30) ORDER BY ename;

ENAME                          DEPTNO
-------------------- ----------
ADAMS                              20
ALLEN                              30
BLAKE                              30
FORD                               20
JAMES                              30
JONES                              20
MARTIN                             30
SCOTT                              20
SMITH                              20
TURNER                             30
WARD                               30

11 rows selected.
```

6. Modify lab2_3.sql to list the name and salary of employees who earn between $5,000 and $12,000, and are in department 20 or 50. Label the columns Employee and Monthly Salary , respectively. Resave lab2_3.sql as lab2_6.sql . Run the statement in lab2_6.sql .

ANS.

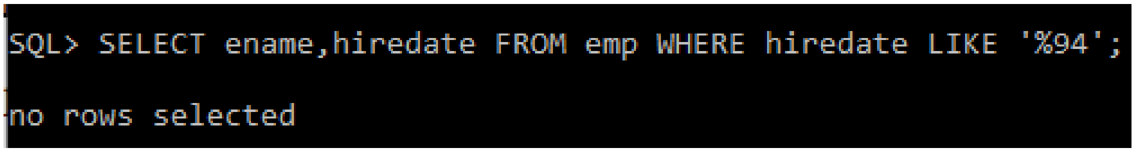


```
SQL> edit e://lab2_3.sql;

SQL> get e://lab2_3.sql;
  1  select ename "Employee", sal "Montly Salary"
  2  from emp
  3* where sal BETWEEN 5000 AND 12000 AND deptno IN (20,50);
SQL> save e://lab2_6.sql;
Created file e://lab2_6.sql
SQL> get e://lab2_6.sql;
  1  select ename "Employee", sal "Montly Salary"
  2  from emp
  3* where sal BETWEEN 5000 AND 12000 AND deptno IN (20,50);
SQL> @e://lab2_6.sql;

no rows selected


no rows selected
```

Kabeer Ahmed
SE-19028

7. Display the name and hire date of every employee who was hired in 1994.

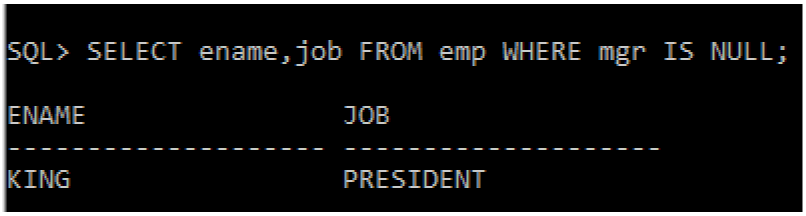
ANS.



```
SQL> SELECT ename,hiredate FROM emp WHERE hiredate LIKE '%94';

no rows selected
```

8. Display the name and job title of all employees who do not have a manager.

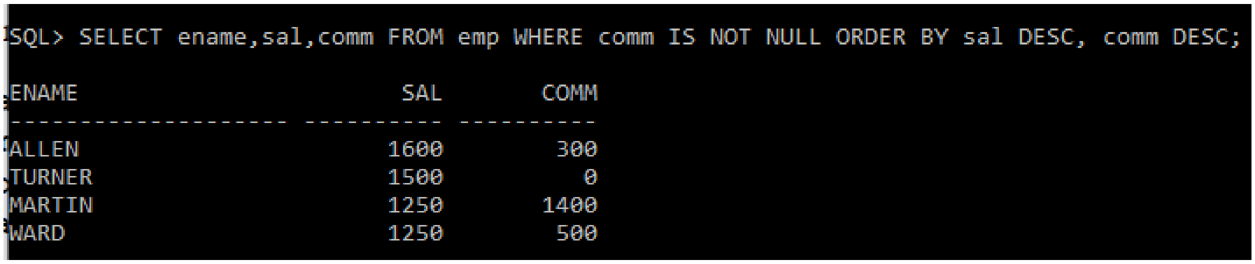
ANS.



```
SQL> SELECT ename,job FROM emp WHERE mgr IS NULL;

ENAME                JOB
-------------------- --------------------
KING                 PRESIDENT
```

9. Display the name, salary, and commission for all employees who earn commissions. Sort data in descending order of salary and commissions.

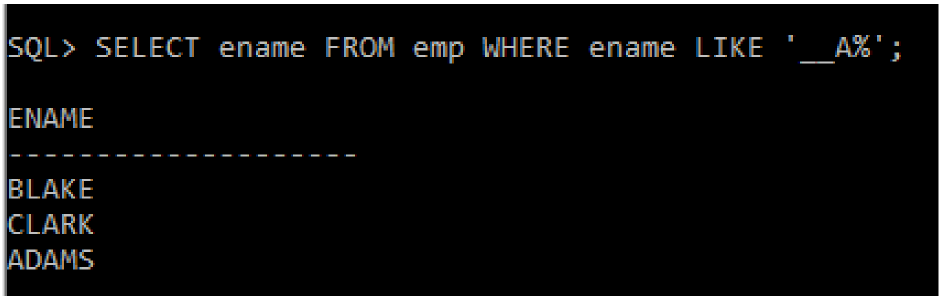
ANS.



```
SQL> SELECT ename,sal,comm FROM emp WHERE comm IS NOT NULL ORDER BY sal DESC, comm DESC;

ENAME                       SAL       COMM
-------------------- ---------- ----------
ALLEN                      1600        300
TURNER                     1500          0
MARTIN                     1250       1400
WARD                       1250        500
```

10. Display the names of all employees where the third letter of the name is an *a*.

ANS.

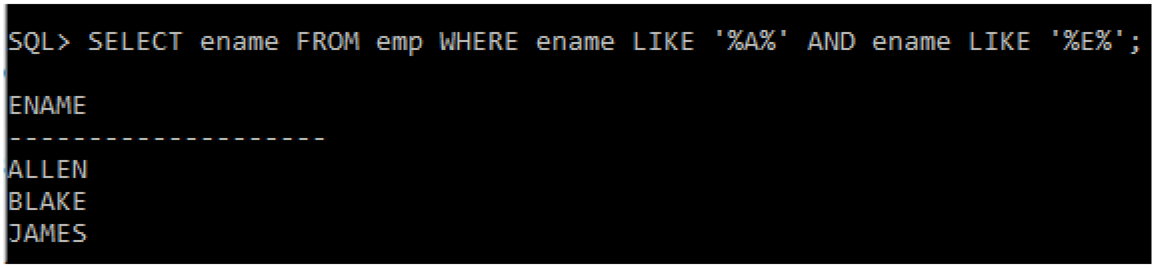


```
SQL> SELECT ename FROM emp WHERE ename LIKE '__A%';

ENAME
--------------------
BLAKE
CLARK
ADAMS
```

Kabeer Ahmed
SE-19028

11. Display the name of all employees who have an *a* and an *e* in their last name.

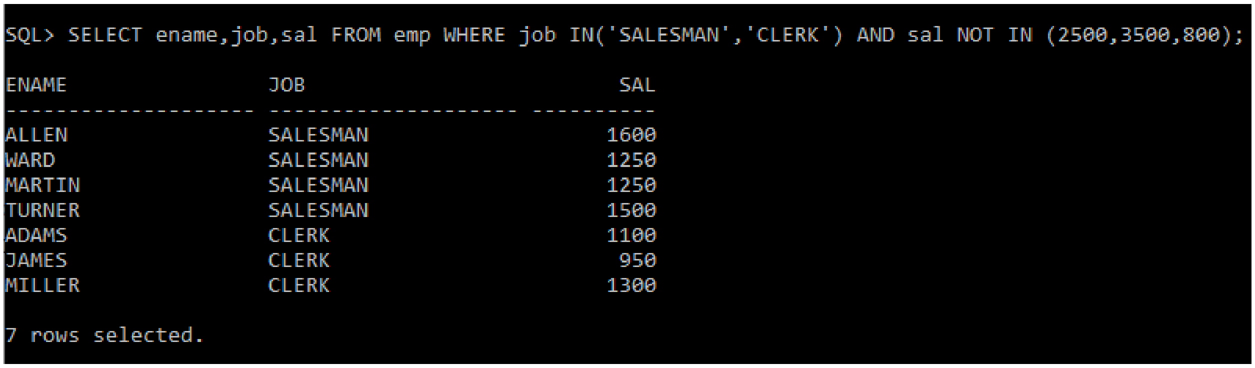
ANS.



```
SQL> SELECT ename FROM emp WHERE ename LIKE '%A%' AND ename LIKE '%E%';

ENAME
--------------------
ALLEN
BLAKE
JAMES
```

12. Display the employee name, job, and salary for all employees whose job is salesman or clerk and whose salary is not equal to $2,500, $3,500, or $800.

ANS.



```
SQL> SELECT ename,job,sal FROM emp WHERE job IN('SALESMAN','CLERK') AND sal NOT IN (2500,3500,800);

ENAME                JOB                        SAL
-------------------- ------------------- ----------
ALLEN                SALESMAN                  1600
WARD                 SALESMAN                  1250
MARTIN               SALESMAN                  1250
TURNER               SALESMAN                  1500
ADAMS                CLERK                     1100
JAMES                CLERK                      950
MILLER               CLERK                     1300

7 rows selected.
```