VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SWE1004 - (L 9 + L10)

Lab Sheet 5

Date functions & Strings

Name: Dishant Naik

Reg no.: 21MIA1127

Perform following queries

(1) Retrieve all data from employee, jobs and deposit.

Answer:

```
SQL> select *
  2  from deposi_21mia1127;

A_NO  CNAME      BNAME          AMOUNT ADATE
----- ---------- ---------- ---------- ---------
101   Anil       andheri          7000 01-JAN-06
104   vijay      andheri          8000 17-SEP-06
105   keyur      dadar            7500 19-NOV-06
103   jay        villeparle       6500 12-MAR-06
102   sunil      virar            5000 15-JUL-06

SQL> select *
  2  from job_21mia1127;

JOB_ID   JOB_TITLE                       MIN_SAL    MAX_SAL
-------- ------------------------------ ---------- ----------
IT_PROG  Programmer                           4000      10000
MK_MGR   Marketing manager                    9000      15000
FI_MGR   Finance manager                      8200      12000

SQL> select *
  2  from employee_21mia1127
  3  ;

    EMP_NO EMP_NAME        EMP_SAL   EMP_COMM    DEPT_NO JOB_ID
---------- ------------ ---------- ---------- ---------- ---------------
       105 ANITA              5000      50000         10 FI_ACC
       106 SNEHA              2450      24500         10 LEC
       107 ANAMIKA            2975                    30 LEC
```

(2) Give details of account no. and deposited rupees of customers having account opened between dates 01-01-06 and 25-07-06.

Answer:

```
SQL> select a_no, amount
  2  from deposi_21mia1127
  3  where adate>'1-jan-06'
  4  and adate<'25-sep-06'
  5  ;

A_NO      AMOUNT
----- ----------
104         8000
103         6500
102         5000
```

(3) Display all jobs with minimum salary is greater than 4000.

Answer:

```
SQL> select job_title
  2  from job_21mia1127
  3  where min_sal > 4000;

JOB_TITLE
-----------------------------
Marketing manager
Finance manager
```

(4) Display name and salary of employee whose department no is 20. Give alias name to name of employee.

Answer:

```
SQL> select emp_name, emp_sal
  2  from employee_21mia1127
  3  where dept_no=20;

no rows selected
```

(5) Display employee no, name and department details of those employee whose department lies in (10, 20)

Answer:

```
SQL> select emp_name, emp_sal
  2  from employee_21mia1127
  3  where dept_no>10 and dept_no<20;

no rows selected
```

To study various options of LIKE predicate:

(6) Display all employee whose name start with ‘A’ and third character is ‘ a’.

Answer:

```
SQL> select *
  2  from employee_21mia1127
  3  WHERE emp_name LIKE 'A_A%';

    EMP_NO EMP_NAME        EMP_SAL   EMP_COMM    DEPT_NO JOB_ID
---------- ------------ ---------- ---------- ---------- ---------------
       107 ANAMIKA            2975                    30 LEC
```

(7) Display name, number and salary of those employees whose name is 5 characters long and first three characters are ‘Ani’.

Answer:

```
SQL> select emp_no, emp_name, emp_sal
  2  from employee_21mia1127
  3  where emp_name LIKE 'ANI__';

    EMP_NO EMP_NAME        EMP_SAL
---------- ------------ ----------
       105 ANITA              5000
```

(8) Display the non-null values of employees and also employee name second character should be ‘n’ and string should be 5 character long.

Answer:

```
SQL> select *
  2  from employee_21mia1127
  3  where emp_name LIKE '_N___';

    EMP_NO EMP_NAME        EMP_SAL   EMP_COMM    DEPT_NO JOB_ID
---------- ------------ ---------- ---------- ---------- ---------------
       105 ANITA              5000      50000         10 FI_ACC
       106 SNEHA              2450      24500         10 LEC
```

(9) Display the null values of employee and also employee name’s third character should be ‘a’.

Answer:

```
SQL> select *
  2  from employee_21mia1127
  3  where emp_name LIKE '__A%'
  4  AND EMP_COMM = 'NULL';

no rows selected
```

(10) What will be output if you are giving LIKE predicate as '%_%' ESCAPE '\'

Answer:

```
SQL> select *
  2  from employee_21mia1127
  3  where emp_name LIKE '%\_%' ;

no rows selected

SQL> select *
  2  from employee_21mia1127
  3  where emp_name LIKE '\' ;

no rows selected

SQL> select *
  2  from employee_21mia1127
  3  where emp_name LIKE '%\_%' ESCAPE '\';

no rows selected
```

```
SQL> select * from job_21mia1127 where JOB_ID LIKE '%\_%'ESCAPE'\'
  2  ;

JOB_ID   JOB_TITLE                      MIN_SAL    MAX_SAL
-------- ------------------------------ ---------- ----------
IT_PROG  Programmer                           4000      10000
MK_MGR   Marketing manager                    9000      15000
FI_MGR   Finance manager                      8200      12000
```

Create tables according to the following definition.
`CREATE TABLE JOB_21mia1127 (JOB_ID VARCHAR2(8), JOB_TITLE VARCHAR2(30), MIN_SAL NUMBER(7,2),MAX_SAL NUMBER(7,2));`
`CREATE TABLE EMPLOYEE_21mia11127 (EMP_NO NUMBER(3), EMP_NAME VARCHAR2(12), EMP_SAL NUMBER(8,2), EMP_COMM NUMBER(6,1) DEPT_NO NUMBER(3), JOB_ID VARCHAR2(15));`
`CREATE TABLE DEPOSI_21mia1127 (A_NO VARCHAR2(5), CNAME VARCHAR2(10), BNAME VARCHAR2(10), AMOUNT NUMBER(7,2), ADATE DATE);`

```
CREATE TABLE BORR_21mia1127 (LOANNO VARCHAR2(5), CNAME VARCHAR2(10),
BNAME VARCHAR2(10), AMOUNT NUMBER (7,2));
```

Table values

INSERT INTO EMPLOYEE_21mia1127 VALUES (105,'ANITA',5000,50000,10,'FI_ACC');

(EMP_NO,EMP_NAME,EMP_SALARY,EMP_COMM,DEPT_NO,JOB_ID)

VALUES (105,'ANITA',5000,50,000,10,'FI_ACC');

INSERT INTO EMPLOYEE_21mia1127 VALUES (106,'SNEHA',2450, 24500,10, 'LEC');

(EMP_NO,EMP_NAME,EMP_SALARY,EMP_COMM,DEPT_NO,JOB_ID)

VALUES (106,'SNEHA',2450, 24,500,10, 'LEC');

INSERT INTO EMPLOYEE_21mia1127 VALUES (107,'ANAMIKA',2975, NULL,30,'LEC');

(EMP_NO,EMP_NAME,EMP_SALARY,EMP_COMM,DEPT_NO,JOB_ID)

VALUES (107,'ANAMIKA',2975, NULL,30,'LEC');

INSERT INTO JOB_21mia1127 VALUES ('IT_PROG ',' Programmer', 4000, 10000);

(JOB_ID,_JOB_NAME,MIN_SAL,MAX_SAL)

VALUES ('IT_PROG ',' Programmer', 4000, 10000);

INSERT INTO JOB_21mia1127 VALUES ('MK_MGR',' Marketing manager', 9000, 15000);

(JOB_ID,_JOB_NAME,MIN_SAL,MAX_SAL)

VALUES ('MK_MGR',' Marketing manager', 9000, 15000);

INSERT INTO JOB_21mia1127 VALUES ('FI_MGR','Finance manager ', 8200,12000);

(JOB_ID,_JOB_NAME,MIN_SAL,MAX_SAL)

VALUES ('FI_MGR','Finance manager ', 8200,12000);

INSERT INTO DEPOSI_21mia1127 VALUES ('101','Anil','andheri',7000,'01-jan-06');

(A_NO,CNAME,BNAME,AMOUNT,A_DATE)

VALUES ('101','Anil','andheri',7000,'01-jan-06');

INSERT INTO DEPOSI_21mia1127 VALUES ('102','sunil','virar',5000,'15-jul-06');

(A_NO,CNAME,BNAME,AMOUNT,A_DATE)

VALUES ('102','sunil','virar',5000,'15-jul-06');

INSERT INTO DEPOSI_21mia1127 VALUES ('103','jay','villeparle',6500,'12-mar-06');

(A_NO,CNAME,BNAME,AMOUNT,A_DATE)

VALUES ('103','jay','villeparle',6500,'12-mar-06');

INSERT INTO DEPOSI_21mia1127 VALUES ('104','vijay','andheri', 8000,'17-sep-06');

(A_NO,CNAME,BNAME,AMOUNT,A_DATE)

VALUES ('104','vijay','andheri', 8000,'17-sep-06');

INSERT INTO DEPOSI_21mia1127 VALUES ('105','keyur','dadar', 7500,'19-nov-06');

(A_NO,CNAME,BNAME,AMOUNT,A_DATE)

VALUES ('105','keyur','dadar', 7500,'19-nov-06');

Important guidelines to be followed

- Every table created should have the last four digits of your registration number **together with the table name(eg:emp_1021)**
- Insert **records** for each table as per the requirements of the query.
- No query must return an answer **"No rows found"**
- *Upload a **PDF** document with*
  - *the screenshots of tables created, Records of the tables,*
  - *SQL queries with answers*

String Operations

- SQL includes a string-matching operator for comparisons on character strings. The operator **like** uses patterns that are described using two special characters:
  - percent (%). The % character matches any substring.
  - underscore (_). The _ character matches any character.
- Find the names of all instructors whose name includes the substring "dar".

```
select name
from instructor
where name like '%dar%'
```

- Match the string "100%"

```
like '100 \%'  escape  '\'
```

in that above we use backslash (\) as the escape character.

- Patterns are case sensitive.
- Pattern matching examples:
  - ‘Intro%’ matches any string beginning with “Intro”.
  - ‘%Comp%’ matches any string containing “Comp” as a substring.
  - ‘_ _ _’ matches any string of exactly three characters.
  - ‘_ _ _ %’ matches any string of at least three characters.

- SQL supports a variety of string operations such as
  - concatenation (using “||”)
  - converting from upper to lower case (and vice versa)
  - finding string length, extracting substrings, etc.