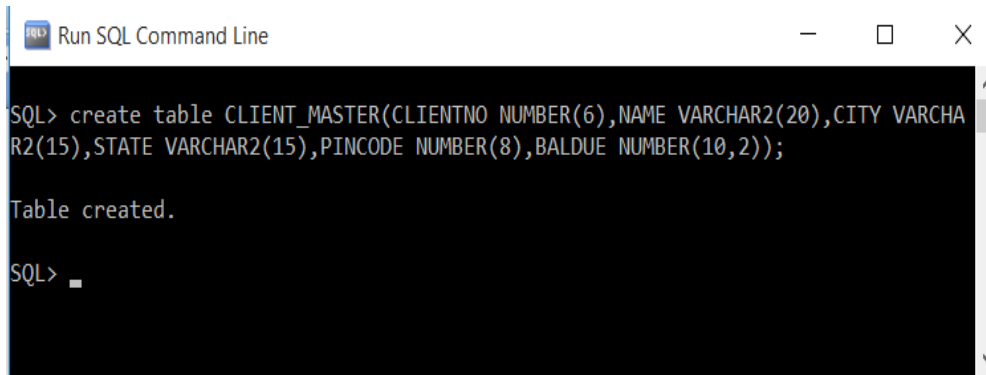


EXERCISE NO: 1

AIM : TO ILLUSTRATE THE DDL COMMANDS

1) CREATE A TABLE CALLED 'CLIENT_MASTER' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE
CLIENTNO	NUMBER	6
NAME	VARCHAR2	20
CITY	VARCHAR2	15
STATE	VARCHAR2	15
PINCODE	NUMBER	8
BALDUE	NUMBER	10,2



```
Run SQL Command Line
SQL> create table CLIENT_MASTER(CLIENTNO NUMBER(6),NAME VARCHAR2(20),CITY VARCHAR2(15),STATE VARCHAR2(15),PINCODE NUMBER(8),BALDUE NUMBER(10,2));
Table created.
SQL> 
```

2) CREATE A TABLE CALLED 'SALES_MASTER' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE
SALESMANNO	VARCHAR2	6
SALESMANNAME	VARCHAR2	10
ADDRESS1	VARCHAR2	10
ADDRESS2	VARCHAR2	10
CITY	VARCHAR2	10
PINCODE	NUMBER	8
STATE	VARCHAR2	10
SALESAMT	NUMBER	8
TGTTOGET	NUMBER	6
YTDSALES	NUMBER	6
REMARKS	VARCHAR2	15

```

SQL> CREATE TABLE SALES_MASTER(SALESMANNO VARCHAR2(6),SALESMANNAME VARCHAR2(10),
ADDRESS1 VARCHAR2(10),ADDRESS2 VARCHAR2(10),CITY VARCHAR2(10),PINCODE NUMBER(8),
STATE VARCHAR2(10),SALESAMT NUMBER(8),TGTTTOGET NUMBER(6),YTDSALES NUMBER(6),REMA
RKS VARCHAR2(15));

Table created.

SQL>
    
```

3) CREATE A TABLE CALLED 'PRODUCT_MASTER' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE
PRODUCTNO	VARCHAR2	6
DESCRIPTION	VARCHAR2	20
PROFITPERCENT	NUMBER	6,2
UNITMEASURE	VARCHAR2	10
SELLPRICE	NUMBER	8,2
COSTPRICE	NUMBER	8,2

```

SQL> CREATE TABLE PRODUCT_MASTER(PRODUCTNO VARCHAR2(6),DESCRIPTION VARCHAR2(20),
PROFITPERCENT NUMBER(6,2),UNITMEASURE VARCHAR2(10),SELLPRICE NUMBER(8,2),COSTPRI
CE NUMBER(8,2));

Table created.

SQL>
    
```

4) CREATE A TABLE CALLED 'DEPT' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
DEPTNO	NUMBER	3	PRIMARK KEY
DNAME	VARCHAR2	20	UNIQUE
DLOCATION	VARCHAR2	20	UNIQUE

```

SQL> CREATE TABLE DEPT(DEPTNO NUMBER(3) PRIMARY KEY, DNAME VARCHAR2(20) UNIQUE,
DLOCATION VARCHAR2(20) UNIQUE);

Table created.

SQL>
    
```

5) CREATE A TABLE CALLED 'EMP' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
ENO	NUMBER	4	PRIMARY KEY
ENAME	VARCHAR2	15	NOT NULL
JOB	VARCHAR2	15	
MGR	NUMBER	4	
HIREDATE	DATE		
SAL	NUMBER	8,2	
COMM	NUMBER	6,2	
DEPTNO	NUMBER	3	FOREIGN KEY

```

Run SQL Command Line
SQL> CREATE TABLE EMP(ENO NUMBER(4) PRIMARY KEY ,ENAME VARCHAR2(15) NOT NULL, JOB
VARCHAR2(15),MGR NUMBER(4),HIREDATE DATE,SAL NUMBER(8,2),COMM NUMBER(6,2),DEPTN
O NUMBER(3) REFERENCES DEPT(DEPTNO));

Table created.

SQL>
    
```

6) CREATE A TABLE CALLED 'SALGRADE' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE
GRADE	NUMBER	2
LOSAL	NUMBER	8,2
HISAL	NUMBER	8,2

```

Run SQL Command Line
SQL> CREATE TABLE SALGRADE(GRADE NUMBER(2), LOSAL NUMBER(8,2),HISAL NUMBER(8,2))
;

Table created.

SQL>
    
```

7) CREATE A TABLE CALLED 'SAILORS' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
SID	NUMBER	4	PRIMARY KEY
SNAME	VARCHAR2	15	
RATING	NUMBER	2	CHECK
AGE	NUMBER	2	CHECK

```

Run SQL Command Line
SQL> CREATE TABLE SAILORS(SID NUMBER(4) PRIMARY KEY,SNAME VARCHAR2(15), RATING NUMBER(2) CHECK (RATING <=10), AGE NUMBER(2) CHECK(AGE<=100));
Table created.
SQL>
    
```

8) CREATE A TABLE CALLED 'BOATS' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
BID	NUMBER	4	PRIMARY KEY
BNAME	VARCHAR2	15	
BCOLOR	VARCHAR2	10	

```

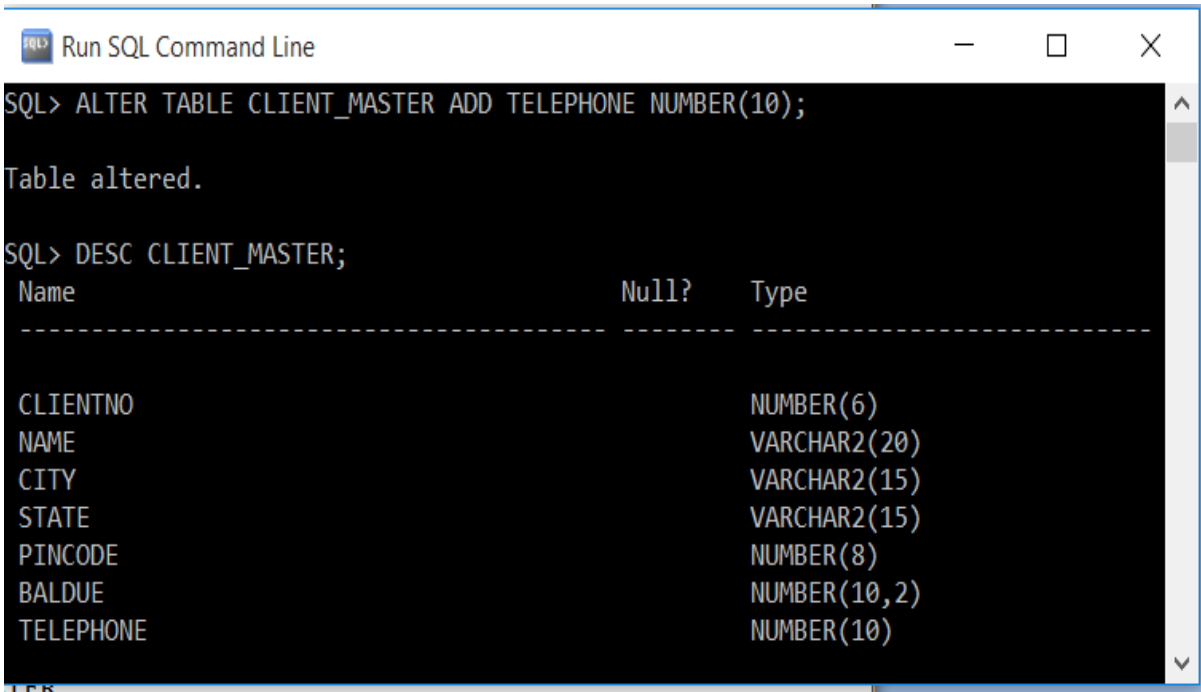
Run SQL Command Line
SQL> CREATE TABLE BOATS(BID NUMBER(4) PRIMARY KEY, BNAME VARCHAR2(15), BCOLOR VARCHAR2(10));
Table created.
SQL>
    
```

9) CREATE A TABLE CALLED 'RESERVES' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
SID	NUMBER	4	FOREIGN KEY
BID	NUMBER	4	FOREIGN KEY
DAY	DATE		

```

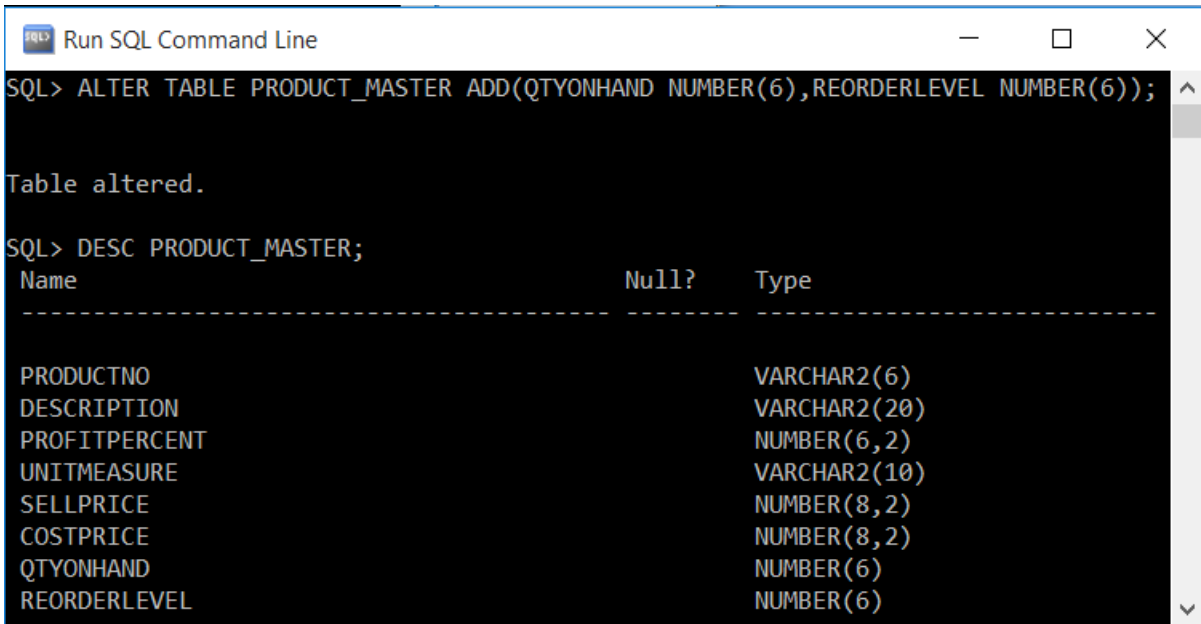
Run SQL Command Line
SQL> CREATE TABLE RESERVES(SID NUMBER(4) REFERENCES SAILORS(SID), BID NUMBER(4) REFERENCES BOATS(BID),DAY DATE);
Table created.
SQL>
    
```

10) ADD A COLUMN 'TELEPHONE' TO THE CLIENT MASTER TABLE.

```
SQL> ALTER TABLE CLIENT_MASTER ADD TELEPHONE NUMBER(10);

Table altered.

SQL> DESC CLIENT_MASTER;
Name                               Null?    Type
-----
CLIENTNO                           NUMBER(6)
NAME                                VARCHAR2(20)
CITY                                 VARCHAR2(15)
STATE                                VARCHAR2(15)
PINCODE                             NUMBER(8)
BALDUE                              NUMBER(10,2)
TELEPHONE                           NUMBER(10)
```

11) ADD 2 COLUMNS 'QTYONHAND AND REORDERLEVEL' IN PRODUCT MASTER TABLE.

```
SQL> ALTER TABLE PRODUCT_MASTER ADD(QTYONHAND NUMBER(6),REORDERLEVEL NUMBER(6));

Table altered.

SQL> DESC PRODUCT_MASTER;
Name                               Null?    Type
-----
PRODUCTNO                          VARCHAR2(6)
DESCRIPTION                          VARCHAR2(20)
PROFITPERCENT                       NUMBER(6,2)
UNITMEASURE                          VARCHAR2(10)
SELLPRICE                            NUMBER(8,2)
COSTPRICE                            NUMBER(8,2)
QTYONHAND                            NUMBER(6)
REORDERLEVEL                         NUMBER(6)
```

12) CHANGE THE DATATYPE OF COLUMN 'CLIENTNO' IN CLIENT MASTER TABLE TO VARCHAR2

```

Run SQL Command Line
SQL> DESC CLIENT_MASTER;
Name                               Null?    Type
-----
CLIENTNO                           NUMBER(6)
NAME                                VARCHAR2(20)
CITY                                VARCHAR2(15)
STATE                               VARCHAR2(15)
PINCODE                             NUMBER(8)
BALDUE                              NUMBER(10,2)
TELEPHONE                           NUMBER(10)

SQL> ALTER TABLE CLIENT_MASTER MODIFY CLIENTNO VARCHAR2(6);

Table altered.

SQL> DESC CLIENT_MASTER;
Name                               Null?    Type
-----
CLIENTNO                           VARCHAR2(6)
NAME                                VARCHAR2(20)
CITY                                VARCHAR2(15)
STATE                               VARCHAR2(15)
PINCODE                             NUMBER(8)
BALDUE                              NUMBER(10,2)
TELEPHONE                           NUMBER(10)
    
```

13) CHANGE THE SIZE OF TWO COLUMNS 'SELLPRICE' & 'COSTPRICE' AS 10,2 IN PRODUCT -MASTER TABLE

```

Run SQL Command Line
SQL> DESC PRODUCT_MASTER;
Name                               Null?    Type
-----
PRODUCTNO                          VARCHAR2(6)
DESCRIPTION                          VARCHAR2(20)
PROFITPERCENT                       NUMBER(6,2)
UNITMEASURE                         VARCHAR2(10)
SELLPRICE                           NUMBER(8,2)
COSTPRICE                           NUMBER(8,2)
QTYONHAND                           NUMBER(6)
REORDERLEVEL                        NUMBER(6)

SQL> ALTER TABLE PRODUCT_MASTER MODIFY(SELLPRICE NUMBER(10,2),COSTPRICE NUMBER(10,2));

Table altered.

SQL> DESC PRODUCT_MASTER;
Name                               Null?    Type
-----
PRODUCTNO                          VARCHAR2(6)
DESCRIPTION                          VARCHAR2(20)
PROFITPERCENT                       NUMBER(6,2)
UNITMEASURE                         VARCHAR2(10)
SELLPRICE                           NUMBER(10,2)
COSTPRICE                           NUMBER(10,2)
QTYONHAND                           NUMBER(6)
REORDERLEVEL                        NUMBER(6)
    
```

14) REMOVE THE 'UNITMEASUR' COLUMN FROM PRODUCT_MASTER

```

Run SQL Command Line
SQL> DESC PRODUCT_MASTER;
Name                               Null?      Type
-----
PRODUCTNO                          VARCHAR2(6)
DESCRIPTION                         VARCHAR2(20)
PROFITPERCENT                       NUMBER(6,2)
UNITMEASURE                         VARCHAR2(10)
SELLPRICE                           NUMBER(10,2)
COSTPRICE                           NUMBER(10,2)
QTYONHAND                           NUMBER(6)
REORDERLEVEL                        NUMBER(6)

SQL> ALTER TABLE PRODUCT_MASTER DROP COLUMN UNITMEASURE;

Table altered.

SQL> DESC PRODUCT_MASTER;
Name                               Null?      Type
-----
PRODUCTNO                          VARCHAR2(6)
DESCRIPTION                         VARCHAR2(20)
PROFITPERCENT                       NUMBER(6,2)
SELLPRICE                           NUMBER(10,2)
COSTPRICE                           NUMBER(10,2)
QTYONHAND                           NUMBER(6)
REORDERLEVEL                        NUMBER(6)

SQL>
    
```

15) REMOVE 'ADDRESS2' COLUMN IN 'SALES_MASTER' TABLE

```

Select Run SQL Command Line
SQL> DESC SALES_MASTER;
Name                               Null?      Type
-----
SALENMANNO                          VARCHAR2(6)
SALESMANNAME                        VARCHAR2(10)
ADDRESS1                             VARCHAR2(10)
ADDRESS2                             VARCHAR2(10)
CITY                                 VARCHAR2(10)
PINCODE                             NUMBER(8)
STATE                                VARCHAR2(10)
SALESAMT                             NUMBER(8)
TGTOGET                              NUMBER(6)
YTDSALES                             NUMBER(6)
REMARKS                              VARCHAR2(15)

SQL> ALTER TABLE SALES_MASTER DROP COLUMN ADDRESS2;

Table altered.

SQL> DESC SALES_MASTER;
Name                               Null?      Type
-----
SALENMANNO                          VARCHAR2(6)
SALESMANNAME                        VARCHAR2(10)
ADDRESS1                             VARCHAR2(10)
CITY                                 VARCHAR2(10)
PINCODE                             NUMBER(8)
STATE                                VARCHAR2(10)
SALESAMT                             NUMBER(8)
TGTOGET                              NUMBER(6)
YTDSALES                             NUMBER(6)
REMARKS                              VARCHAR2(15)
    
```

16) REMOVE 'TGTTOGET','YTDSALES','REMARKS' FROM 'SALES_MASTER' TABLE

```

Run SQL Command Line
SQL> DESC SALES_MASTER;
Name                               Null?      Type
-----
SALENMANNO                          VARCHAR2(6)
SALESMANNAME                         VARCHAR2(10)
ADDRESS1                             VARCHAR2(10)
CITY                                  VARCHAR2(10)
PINCODE                              NUMBER(8)
STATE                                VARCHAR2(10)
SALESAMT                             NUMBER(8)
TGTTOGET                             NUMBER(6)
YTDSALES                             NUMBER(6)
REMARKS                              VARCHAR2(15)

SQL> ALTER TABLE SALES_MASTER DROP (TGTTOGET,YTDSALES,REMARKS);
Table altered.

SQL> DESC SALES_MASTER;
Name                               Null?      Type
-----
SALENMANNO                          VARCHAR2(6)
SALESMANNAME                         VARCHAR2(10)
ADDRESS1                             VARCHAR2(10)
CITY                                  VARCHAR2(10)
PINCODE                              NUMBER(8)
STATE                                VARCHAR2(10)
SALESAMT                             NUMBER(8)

SQL>

```

17) CHANGE THE 'DNAME' AS DEPTNAME IN 'DEPT' TABLE

```

Run SQL Command Line
SQL> DESC DEPT;
Name                               Null?      Type
-----
DEPTNO                             NOT NULL  NUMBER(3)
DNAME                              VARCHAR2(20)
DLOCATION                            VARCHAR2(20)

SQL> ALTER TABLE DEPT RENAME COLUMN DNAME TO DEPTNAME;
Table altered.

SQL> DESC DEPT;
Name                               Null?      Type
-----
DEPTNO                             NOT NULL  NUMBER(3)
DEPTNAME                            VARCHAR2(20)
DLOCATION                            VARCHAR2(20)

SQL>

```

18) REMOVE THE TABLE 'PRODUCT_MASTER'

```
Run SQL Command Line
SQL> DESC PRODUCT_MASTER;
Name                               Null?      Type
-----
PRODUCTNO                          VARCHAR2(6)
DESCRIPTION                         VARCHAR2(20)
PROFITPERCENT                       NUMBER(6,2)
SELLPRICE                           NUMBER(10,2)
COSTPRICE                           NUMBER(10,2)
QTYONHAND                           NUMBER(6)
REORDERLEVEL                        NUMBER(6)

SQL> DROP TABLE PRODUCT_MASTER;
Table dropped.

SQL> DESC PRODUCT_MASTER;
ERROR:
ORA-04043: object PRODUCT_MASTER does not exist

SQL> _
```

19) DISPLAY THE STRUCTURE OF 'CLIENT_MASTER'

```
Run SQL Command Line
SQL> DESC CLIENT_MASTER;
Name                               Null?      Type
-----
CLIENTNO                          VARCHAR2(6)
NAME                               VARCHAR2(20)
CITY                               VARCHAR2(15)
STATE                             VARCHAR2(15)
PINCODE                           NUMBER(8)
BALDUE                            NUMBER(10,2)
TELEPHONE                         NUMBER(10)

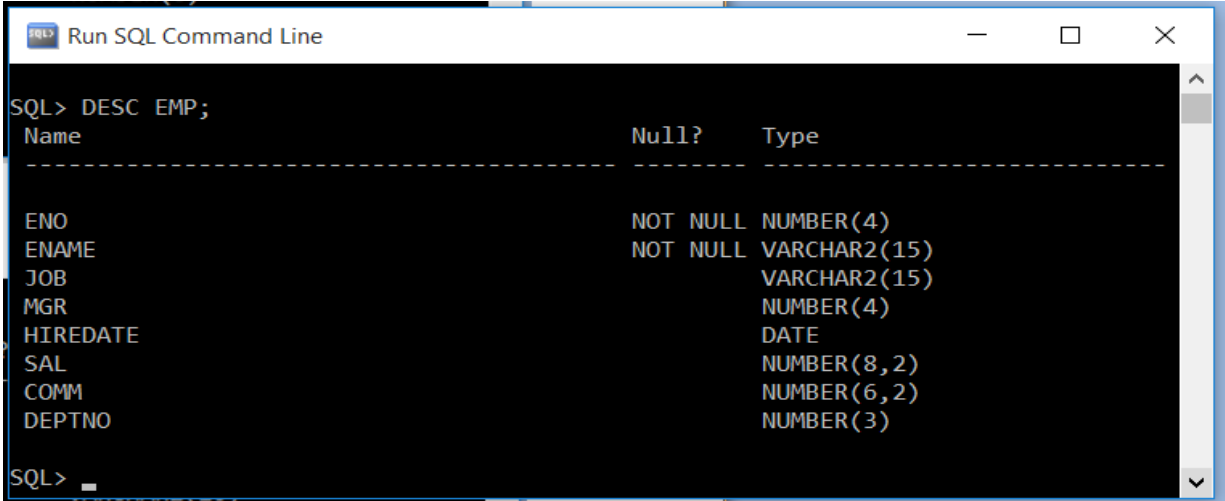
SQL>
```

20) DISPLAY THE STRUCTURE OF 'SALES_MASTER'

```
Run SQL Command Line
SQL> DESC SALES_MASTER;
Name                               Null?      Type
-----
SALENMANNO                         VARCHAR2(6)
SALESMANNAME                       VARCHAR2(10)
ADDRESS1                           VARCHAR2(10)
CITY                               VARCHAR2(10)
PINCODE                            NUMBER(8)
STATE                             VARCHAR2(10)
SALESAMT                           NUMBER(8)

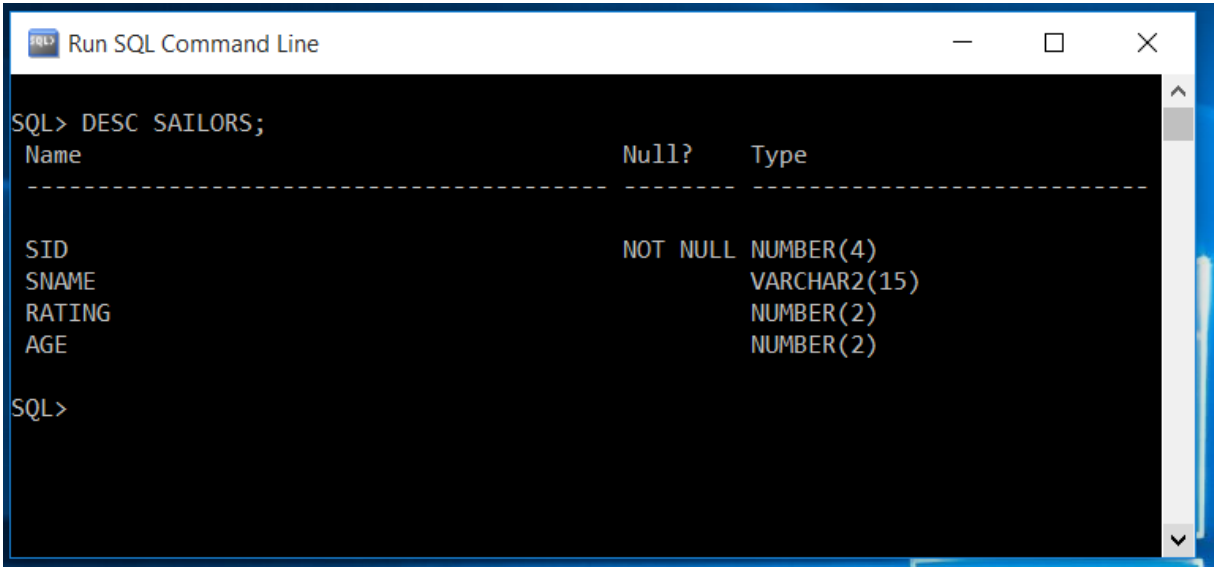
SQL>
```

21) DISPLAY THE STRUCTURE OF 'EMP'



```
Run SQL Command Line
SQL> DESC EMP;
Name                               Null?    Type
-----
ENO                                 NOT NULL NUMBER(4)
ENAME                               NOT NULL VARCHAR2(15)
JOB                                  VARCHAR2(15)
MGR                                  NUMBER(4)
HIREDATE                             DATE
SAL                                  NUMBER(8,2)
COMM                                 NUMBER(6,2)
DEPTNO                               NUMBER(3)
SQL>
```

22) DISPLAY THE STRUCTURE OF 'SAILORS'



```
Run SQL Command Line
SQL> DESC SAILORS;
Name                               Null?    Type
-----
SID                                 NOT NULL NUMBER(4)
SNAME                               VARCHAR2(15)
RATING                              NUMBER(2)
AGE                                  NUMBER(2)
SQL>
```

EXERCISE NO: 2

AIM : TO ILLUSTRATE THE DML COMMANDS

1) POPULATE THE 'CLIENT_MASTER' TABLE WITH THE FOLLOWING DATA

CLIENT_NO	NAME	CITY	PINCODE	STATE	BALDUE	TELEPHONE
C00001	IVAN BAYROSS	MUMBAI	400054	MAHARASTRA	15000	9867546456
C00002	MAMTAMUZUMDAR	MUMBAI	780001	MAHARASTRA	0	9876509455
C00003	CHHAYA BANKAR	MUMBAI	400057	MAHARASTRA	5000	9342567854
C00004	ASHWINIJOSHI	BANGALORE	560001	KARNATAKA	0	9087654097
C00005	HANSELCOLACO	MUMBAI	400060	MAHARASTRA	2000	9761234321
C00006	DEEPAKSHARMA	MANGALORE	560050	KARNATAKA	0	9087654009

```

Run SQL Command Line
SQL> INSERT INTO CLIENT_MASTER VALUES('&CLIENTNO', '&NAME', '&CITY', '&STATE', &PINCODE, &BALDUE, &TELEPHONE);
Enter value for clientno: C00001
Enter value for name: IVAN BAYROSS
Enter value for city: MUMBAI
Enter value for state: MAHARASTRA
Enter value for pincode: 400054
Enter value for baldue: 1500
Enter value for telephone: 9867546456
old 1: INSERT INTO CLIENT_MASTER VALUES('&CLIENTNO', '&NAME', '&CITY', '&STATE', &PINCODE, &BALDUE, &TELEPHONE)
new 1: INSERT INTO CLIENT_MASTER VALUES('C00001', 'IVAN BAYROSS', 'MUMBAI', 'MAHARASTRA', 400054, 1500, 9867546456)

1 row created.

SQL> /
Enter value for clientno: C00002
Enter value for name: MAMTAMUZUMDAR
Enter value for city: MUMBAI
Enter value for state: MAHARASTRA
Enter value for pincode: 780001
Enter value for baldue: 0
Enter value for telephone: 9876509455
old 1: INSERT INTO CLIENT_MASTER VALUES('&CLIENTNO', '&NAME', '&CITY', '&STATE', &PINCODE, &BALDUE, &TELEPHONE)
new 1: INSERT INTO CLIENT_MASTER VALUES('C00002', 'MAMTAMUZUMDAR', 'MUMBAI', 'MAHARASTRA', 780001, 0, 9876509455)

1 row created.

SQL>
    
```

```

Run SQL Command Line
SQL> SET LINESIZE 100;
SQL> SELECT * FROM CLIENT_MASTER;

CLIENT NAME          CITY          STATE          PINCODE          BALDUE  TELEPHONE
-----
C00001 IVAN BAYROSS    MUMBAI         MAHARASTRA      400054    1500 9867546456
C00002 MAMTAMUZUMDAR  MUMBAI         MAHARASTRA      780001     0 9876509455
C00003 CHHAYA BANKAR  MUMBAI         MAHARASTRA      400057    5000 9342567854
C00004 ASHWINI JOSHI  BANGALORE     KARNATAKA       560001     0 9087654097
C00005 HANSEL COLACO  MUMBAI         MAHARASTRA      400060    2000 9761234321
C00006 DEEPAK SHARMA MANGALORE     KARNATAKA       560050     0 9087654009

6 rows selected.

SQL>
    
```

2) POPULATE THE 'SALES_MASTER' TABLE WITH THE FOLLOWING DATA

SALESMANNO	SALESMANNAME	ADDRESS1	CITY	STATE	PINCODE	SALESAMT
S0001	AMAN	BANDRA	MUMBAI	MAHARASTRA	400002	45000
S0002	OMKAR	NANMAN	MUMBAI	MAHARASTRA	400001	25000
S0003	RAJ	JUHI	MUMBAI	MAHARASTRA	400003	30000
S0004	ASHISH	KADAR	PUNE	MAHARASTRA	400044	60000

```

Run SQL Command Line
SQL> INSERT INTO SALES_MASTER VALUES('&SALESMANNO','&SALESMANNAME','&ADDRESS1','&CITY','&PINCODE','&STATE','&SALESAMT');
Enter value for salesmanno: S0001
Enter value for salesmanname: AMAN
Enter value for address1: BANDRA
Enter value for city: MUMBAI
Enter value for pincode: 400002
Enter value for state: MAHARASTRA
Enter value for salesamt: 45000
old 1: INSERT INTO SALES_MASTER VALUES('&SALESMANNO','&SALESMANNAME','&ADDRESS1','&CITY','&PINCODE','&STATE','&SALESAMT)
new 1: INSERT INTO SALES_MASTER VALUES('S0001','AMAN','BANDRA','MUMBAI',400002,'MAHARASTRA',45000)

1 row created.

SQL> /
Enter value for salesmanno: S0002
Enter value for salesmanname: OMKAR
Enter value for address1: NANMAN
Enter value for city: MUMBAI
Enter value for pincode: 400001
Enter value for state: MAHARASTRA
Enter value for salesamt: 25000
old 1: INSERT INTO SALES_MASTER VALUES('&SALESMANNO','&SALESMANNAME','&ADDRESS1','&CITY','&PINCODE','&STATE','&SALESAMT)
new 1: INSERT INTO SALES_MASTER VALUES('S0002','OMKAR','NANMAN','MUMBAI',400001,'MAHARASTRA',25000)

1 row created.

SQL> COMMIT;

Commit complete.
    
```

```

Run SQL Command Line
SQL> SELECT * FROM SALES_MASTER;

SALENM SALESMANNA ADDRESS1 CITY PINCODE STATE SALESAMT
-----
S0001 AMAN BANDRA MUMBAI 400002 MAHARASTRA 45000
S0002 OMKAR NANMAN MUMBAI 400001 MAHARASTRA 25000
S0003 RAJ JUHI MUMBAI 400003 MAHARASTRA 30000
S0004 ASHISH KADAR PUNE 400044 MAHARASTRA 60000

SQL>
    
```

3) POPULATE THE 'DEPT' TABLE WITH THE FOLLOWING DATA

DEPTNO	DEPTNAME	DEPTLOCATION
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

```

Run SQL Command Line
SQL> DESC DEPT
Name                               Null?   Type
-----
DEPTNO                             NOT NULL NUMBER(3)
DEPTNAME                           VARCHAR2(20)
DLOCATION                            VARCHAR2(20)

SQL> INSERT INTO DEPT VALUES(&DEPTNO, '&DEPTNAME', '&DLOCATION');
Enter value for deptno: 10
Enter value for deptname: ACCOUNTING
Enter value for dlocation: NEW YORK
old 1: INSERT INTO DEPT VALUES(&DEPTNO, '&DEPTNAME', '&DLOCATION')
new 1: INSERT INTO DEPT VALUES(10, 'ACCOUNTING', 'NEW YORK')

1 row created.

SQL> /
Enter value for deptno: 20
Enter value for deptname: RESEARCH
Enter value for dlocation: DALLAS
old 1: INSERT INTO DEPT VALUES(&DEPTNO, '&DEPTNAME', '&DLOCATION')
new 1: INSERT INTO DEPT VALUES(20, 'RESEARCH', 'DALLAS')

1 row created.

SQL> COMMIT;

Commit complete.
    
```

```

Run SQL Command Line
SQL> SELECT * FROM DEPT;

DEPTNO DEPTNAME      DLOCATION
-----
10 ACCOUNTING      NEW YORK
20 RESEARCH        DALLAS
30 SALES           CHICAGO
40 OPERATIONS      BOSTON

SQL>
    
```

4) POPULATE THE 'SALGRADE' TABLE WITH THE FOLLOWING DATA

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
4	2001	3000
5	3001	9999
3	1401	2000

```
Run SQL Command Line
SQL> DESC SALGRADE;
Name                               Null?   Type
-----
GRADE                               NUMBER(2)
LOSAL                               NUMBER(8,2)
HISAL                               NUMBER(8,2)

SQL> INSERT INTO SALGRADE VALUES(&GRADE,&LOSAL,&HISAL);
Enter value for grade: 1
Enter value for losal: 700
Enter value for hisal: 1200
old 1: INSERT INTO SALGRADE VALUES(&GRADE,&LOSAL,&HISAL)
new 1: INSERT INTO SALGRADE VALUES(1,700,1200)

1 row created.

SQL> /
Enter value for grade: 2
Enter value for losal: 1201
Enter value for hisal: 1400
old 1: INSERT INTO SALGRADE VALUES(&GRADE,&LOSAL,&HISAL)
new 1: INSERT INTO SALGRADE VALUES(2,1201,1400)

1 row created.

SQL> COMMIT
  2 ;
```

```
Run SQL Command Line
SQL> SELECT * FROM SALGRADE;

  GRADE   LOSAL   HISAL
-----
      1     700    1200
      2    1201    1400
      3    1401    2000
      4    2001    3000
      5    3001    9999

SQL>
```

5) POPULATE THE 'EMP' TABLE WITH THE FOLLOWING DATA

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7499	ALLEN	SALESMAN	7698	15-AUG-98	1600	300	30
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30
7566	JONES	MANAGER	7839	31-OCT-95	2975		20
7698	BLAKE	MANAGER	7839	11-JUN-92	2850		30
7782	CLARK	MANAGER	7839	14-MAY-93	2450		10
7788	SCOTT	ANALYST	7566	5-MAR-96	3000		20
7839	KING	PRESIDENT		9-JUN-90	5000	0	10
7844	TURNER	SALESMAN	7698	4-JUN-95	1500	0	30
7876	ADAMS	CLERK	7788	4-JUN-99	1100		20
7900	JAMES	CLERK	7698	23-JUN-00	950		30
7934	MILLER	CLERK	7782	21-JAN-00	1300		10
7902	FORD	ANALYST	7566	5-DEC-97	3000		20
7654	MARTIN	SALESMAN	7698	5-DEC-98	1250	1400	30

```

Run SQL Command Line
SQL> SELECT * FROM EMP;

      ENO  ENAME      JOB              MGR  HIREDATE      SAL      COMM      DEPTNO
-----
7369 SMITH      CLERK              7902 13-JUN-93      800         0         20
7499 ALLEN      SALESMAN           7698 15-AUG-98     1600        300        30
7521 WARD       SALESMAN           7698 26-MAR-96     1250        500        30
7566 JONES      MANAGER            7839 31-OCT-95     2975         0         20
7698 BLAKE      MANAGER            7839 11-JUN-92     2850         0         30
7782 CLARK      MANAGER            7839 14-MAY-93     2450         0         10
7788 SCOTT      ANALYST            7566 05-MAR-96     3000         0         20
7839 KING      PRESIDENT          7839 09-JUN-90     5000         0         10
7844 TURNER    SALESMAN           7698 04-JUN-95     1500         0         30
7876 ADAMS     CLERK              7788 04-JUN-99     1100         0         20
7900 JAMES     CLERK              7698 23-JUN-00      950         0         30

      ENO  ENAME      JOB              MGR  HIREDATE      SAL      COMM      DEPTNO
-----
7934 MILLER    CLERK              7782 21-JAN-00     1300         0         10
7902 FORD      ANALYST            7566 05-DEC-97     3000         0         20
7654 MARTIN   SALESMAN           7698 05-DEC-98     1250       1400        30

14 rows selected.
  
```

6) POPULATE THE 'SAILORS' TABLE WITH THE FOLLOWING DATA

SID	SNAME	RATING	AGE
22	DUSTIN	7	45
29	BRUTUS	1	33
31	LUBBER	8	55.5
32	ANDY	8	25.5
58	RUSTY	10	35
64	HORATIO	7	35
71	ZORBA	10	16
74	HORATIO	9	40
85	ART	3	25.5
95	BOB	3	63.5

```

Run SQL Command Line
SQL> SELECT * FROM SAILORS;

  SID SNAME      RATING  AGE
-----
  22 Dustin         7      45
  29 Brutus         8      33
  31 Lubber         8     55
  32 Andy           8     25
  58 Rusty        10     35
  64 Horatio        7     35
  71 Zorba         10     16
  74 Horatio         9     35
  85 Art            3     25
  95 Bob            3     63
  65 smith          9     40

11 rows selected.

SQL>
    
```

7) POPULATE THE 'RESERVES' TABLE WITH THE FOLLOWING DATA

SID	BID	DAY
22	101	10-Oct-98
22	102	10-Oct-98
22	103	10-Aug-98
22	104	10-Jul-98
31	102	11-Oct-98
31	103	11-Jun-98
31	104	11-Dec-98
64	101	9-May-98
64	102	9-Aug-98
74	103	9-Aug-98

```

Run SQL Command Line
SQL> SELECT * FROM RESERVES;

  SID      BID  DAY
-----
    22     101 10-OCT-98
    22     102 10-OCT-98
    22     103 10-AUG-98
    22     104 10-JUL-98
    31     102 11-OCT-98
    31     103 11-JUN-98
    31     104 11-DEC-98
    64     101 09-MAY-98
    64     102 09-AUG-98
    74     103 09-AUG-98

10 rows selected.

SQL>
    
```

8) POPULATE THE 'BOATS' TABLE WITH THE FOLLOWING DATA

BID	BNAME	BCOLOR
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

```

Run SQL Command Line
SQL> SELECT * FROM BOATS;

  BID  BNAME      BCOLOR
-----
    101 Interlake   blue
    102 Interlake   red
    103 Clipper   green
    104 Marine    red

SQL>
    
```

9) DISPLAY ALL CLIENTS WHO STAY IN 'MUMBAI'

```

Run SQL Command Line
SQL> SET LINESIZE 100;
SQL> SELECT * FROM CLIENT_MASTER WHERE CITY = 'MUMBAI';

CLIENT NAME      CITY      STATE      PINCODE      BALDUE  TELEPHONE
-----
C00001 IVAN BAYROSS      MUMBAI      MAHARASTRA      400054      1500 9867546456
C00002 MAMTAMUZUMDAR    MUMBAI      MAHARASTRA      780001           0 9876509455
C00003 CHHAYA BANKAR    MUMBAI      MAHARASTRA      400057      5000 9342567854

SQL>
    
```

10) DISPLAY ALL EMPLOYEES WHOSE DESIGNATION IS 'SALESMAN'

```
Run SQL Command Line

SQL> SELECT * FROM EMP WHERE JOB = 'SALESMAN';

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
7499 ALLEN              SALESMAN         7698 15-AUG-98         1600      300       30
7521 WARD                SALESMAN         7698 26-MAR-96         1250      500       30
7844 TURNER             SALESMAN         7698 04-JUN-95          1500       0        30
7654 MARTIN             SALESMAN         7698 05-DEC-98          1250     1400       30

SQL>
```

11) DISPLAY EMPLOYEES WORKING IN 10TH DEPARTMENT

```
Run SQL Command Line

SQL> SELECT * FROM EMP WHERE DEPTNO = 10;

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
7782 CLARK              MANAGER         7839 14-MAY-93          2450       0        10
7839 KING                PRESIDENT              09-JUN-90          5000       0        10
7934 MILLER              CLERK           7782 21-JAN-00           1300       0        10

SQL>
```

12) DISPLAY EMPLOYEES WHO GETS SALARY GREATER THAN 3000

```
Run SQL Command Line

SQL> SELECT * FROM EMP WHERE SAL > 3000;

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
7839 KING                PRESIDENT              09-JUN-90          5000       0        10

SQL>
```

13) CHANGE THE CITY AND STATE OF CLIENT WHOSE NUMBER IS C00005 TO BANGALORE AND KARNATAKA

```
Run SQL Command Line
SQL> SELECT * FROM CLIENT_MASTER WHERE CLIENTNO = 'C00005';

CLIENT NAME          CITY          STATE          PINCODE    BALDUE  TELEPHONE
-----
C00005 HANSELCOLACO  MUMBAI        MAHARASTRA    400060    2000  9761234321

SQL> UPDATE CLIENT_MASTER SET CITY = 'BANGALORE', STATE = 'KARNATAKA' WHERE CLIENTNO='C00005';

1 row updated.

SQL> SELECT * FROM CLIENT_MASTER WHERE CLIENTNO = 'C00005';

CLIENT NAME          CITY          STATE          PINCODE    BALDUE  TELEPHONE
-----
C00005 HANSELCOLACO  BANGALORE     KARNATAKA     400060    2000  9761234321
```

14) CHANGE THE BALDUE OF CLIENT_NO 'C00006' TO 1000

```
Run SQL Command Line
SQL> SELECT * FROM CLIENT_MASTER WHERE CLIENTNO = 'C00006';

CLIENT NAME          CITY          STATE          PINCODE    BALDUE  TELEPHONE
-----
C00006 DEEPAKSHARMA  MANGALORE     KARNATAKA     560050     0  9087654009

SQL> UPDATE CLIENT_MASTER SET BALDUE = 1000 WHERE CLIENTNO='C00006';

1 row updated.

SQL> SELECT * FROM CLIENT_MASTER WHERE CLIENTNO = 'C00006';

CLIENT NAME          CITY          STATE          PINCODE    BALDUE  TELEPHONE
-----
C00006 DEEPAKSHARMA  MANGALORE     KARNATAKA     560050    1000  9087654009
```

15) LIST ALL THE EMPLOYEES

```
Run SQL Command Line
SQL> SELECT * FROM EMP;
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7499	ALLEN	SALESMAN	7698	15-AUG-98	1600	300	30
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30
7566	JONES	MANAGER	7839	31-OCT-95	2975		20
7698	BLAKE	MANAGER	7839	11-JUN-92	2850		30
7782	CLARK	MANAGER	7839	14-MAY-93	2450		10
7788	SCOTT	ANALYST	7566	05-MAR-96	3000		20
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7844	TURNER	SALESMAN	7698	04-JUN-95	1500	0	30
7876	ADAMS	CLERK	7788	04-JUN-99	1100		20
7900	JAMES	CLERK	7698	23-JUN-00	950		30
7934	MILLER	CLERK	7782	21-JAN-00	1300		10
7902	FORD	ANALYST	7566	05-DEC-97	3000		20

16) LIST ALL EMPLOYEES WHOSE NAME STARTS WITH 'S'

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE ENAME LIKE 'S%';
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7788	SCOTT	ANALYST	7566	05-MAR-96	3000		20

```
SQL>
```

17) DISPLAY EMPLOYEES WHOSE SALARY RANGES BETWEEN 2000 TO 3000

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE SAL BETWEEN 2000 AND 3000;
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7566	JONES	MANAGER	7839	31-OCT-95	2975		20
7698	BLAKE	MANAGER	7839	11-JUN-92	2850		30
7782	CLARK	MANAGER	7839	14-MAY-93	2450		10
7788	SCOTT	ANALYST	7566	05-MAR-96	3000		20
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7902	FORD	ANALYST	7566	05-DEC-97	3000		20

```
6 rows selected.
```

18) DISPLAY EMPLOYEE'S WHOSE JOB IS PRESIDENT, SALESMAN AND CLERK

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE JOB IN ('PRESIDENT','CLERK','SALESMAN');
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7499	ALLEN	SALESMAN	7698	15-AUG-98	1600	300	30
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7844	TURNER	SALESMAN	7698	04-JUN-95	1500	0	30
7876	ADAMS	CLERK	7788	04-JUN-99	1100		20
7900	JAMES	CLERK	7698	23-JUN-00	950		30
7934	MILLER	CLERK	7782	21-JAN-00	1300		10
7654	MARTIN	SALESMAN	7698	05-DEC-98	1250	1400	30

```

9 rows selected.
SQL>
DISPLAY EMPLOYEE'S WHOSE JOB IS PRESIDENT, SALESMAN AND CLERK

```

19) DISPLAY EMPLOYEE'S WHOSE NAME DOESNOT START WITH 'A'

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE ENAME NOT LIKE 'A%';
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30
7566	JONES	MANAGER	7839	31-OCT-95	2975		20
7698	BLAKE	MANAGER	7839	11-JUN-92	2850		30
7782	CLARK	MANAGER	7839	14-MAY-93	2450		10
7788	SCOTT	ANALYST	7566	05-MAR-96	3000		20
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7844	TURNER	SALESMAN	7698	04-JUN-95	1500	0	30
7900	JAMES	CLERK	7698	23-JUN-00	950		30
7934	MILLER	CLERK	7782	21-JAN-00	1300		10
7902	FORD	ANALYST	7566	05-DEC-97	3000		20
7654	MARTIN	SALESMAN	7698	05-DEC-98	1250	1400	30

```

12 rows selected.
SQL>

```

20) DISPLAY EMPLOYEE'S WHOSE NAME HAS EXACTLY 4 CHARACTERS

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE ENAME LIKE '____';
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7902	FORD	ANALYST	7566	05-DEC-97	3000		20

```

SQL>

```

21) DISPLAY EMPLOYEE'S WHO ARE NOT GETTING COMMISSION

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE COMM IS NULL;
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7566	JONES	MANAGER	7839	31-OCT-95	2975		20
7698	BLAKE	MANAGER	7839	11-JUN-92	2850		30
7782	CLARK	MANAGER	7839	14-MAY-93	2450		10
7788	SCOTT	ANALYST	7566	05-MAR-96	3000		20
7876	ADAMS	CLERK	7788	04-JUN-99	1100		20
7900	JAMES	CLERK	7698	23-JUN-00	950		30
7934	MILLER	CLERK	7782	21-JAN-00	1300		10
7902	FORD	ANALYST	7566	05-DEC-97	3000		20

```
8 rows selected.
```

22) DISPLAY EMPLOYEE'S WHO ARE GETTING COMMISSION

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE COMM IS NOT NULL;
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7499	ALLEN	SALESMAN	7698	15-AUG-98	1600	300	30
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7844	TURNER	SALESMAN	7698	04-JUN-95	1500	0	30
7654	MARTIN	SALESMAN	7698	05-DEC-98	1250	1400	30

```
6 rows selected.
SQL>
```

23) DISPLAY CLERKS WORKING IN 10TH DEPARTMENT

```
Run SQL Command Line
SQL> SELECT * FROM EMP WHERE JOB = 'CLERK' AND DEPTNO = 10;
```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7934	MILLER	CLERK	7782	21-JAN-00	1300		10

```
SQL>
```

24) DISPLAY CLERKS GETTING SALARY LESS THAN 1000

```

Run SQL Command Line
SQL> SELECT * FROM EMP WHERE JOB = 'CLERK' AND SAL < 1000;

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
      7369 SMITH         CLERK              7902 13-JUN-93         800        0         20
      7900 JAMES         CLERK              7698 23-JUN-00         950        0         30

SQL>
    
```

25) CHANGE SCOTT DEPARTMENT TO 10

```

Run SQL Command Line
SQL> SELECT * FROM EMP WHERE ENAME = 'SCOTT';

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
      7788 SCOTT         ANALYST           7566 05-MAR-96         3000        0         20

SQL> UPDATE EMP SET DEPTNO = 10 WHERE ENAME = 'SCOTT';

1 row updated.

SQL> SELECT * FROM EMP WHERE ENAME = 'SCOTT';

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
      7788 SCOTT         ANALYST           7566 05-MAR-96         3000        0         10

SQL>
    
```

26) LIST ALL CLIENTS WHO STAY IN BANGALORE OR MANGALORE

```

Run SQL Command Line
SQL> SELECT * FROM CLIENT_MASTER WHERE CITY = 'BANGALORE' OR CITY = 'MANGALORE';

CLIENT NAME          CITY              STATE              PINCODE      BALDUE  TELEPHONE
-----
C00004 ASHWINI JOSHI    BANGALORE         KARNATAKA         560001        0  9087654097
C00005 HANSEL COLACO    BANGALORE         KARNATAKA         400060       2000 9761234321
C00006 DEEPAK SHARMA  MANGALORE         KARNATAKA         560050       1000 9087654009

SQL>
    
```

27) INCREASE THE MANAGER'S SALARIES BY 10 %

```

Run SQL Command Line
SQL> SET LINESIZE 100;
SQL> SELECT * FROM EMP WHERE JOB = 'MANAGER';

      ENO ENAME          JOB              MGR HIREDATE          SAL          COMM          DEPTNO
-----
      7566 JONES          MANAGER          7839 31-OCT-95       3272.5          0             20
      7698 BLAKE          MANAGER          7839 11-JUN-92        3135            0             30
      7782 CLARK          MANAGER          7839 14-MAY-93        2695            0             10

SQL> UPDATE EMP SET SAL = SAL + 0.1*SAL WHERE JOB = 'MANAGER';

3 rows updated.

SQL> SELECT * FROM EMP WHERE JOB = 'MANAGER';

      ENO ENAME          JOB              MGR HIREDATE          SAL          COMM          DEPTNO
-----
      7566 JONES          MANAGER          7839 31-OCT-95       3599.75          0             20
      7698 BLAKE          MANAGER          7839 11-JUN-92        3448.5            0             30
      7782 CLARK          MANAGER          7839 14-MAY-93        2964.5            0             10

SQL>
    
```

28) REMOVE CLIENTS WHO STAY IN MANGALORE

```

Run SQL Command Line
SQL> SET LINESIZE 100;
SQL> SELECT * FROM CLIENT_MASTER;

CLIENT NAME          CITY          STATE          PINCODE          BALDUE TELEPHONE
-----
C00001 IVAN BAYROSS    MUMBAI        MAHARASTRA      400054          1500 9867546456
C00002 MAMTAMUZUMDAR    MUMBAI        MAHARASTRA      780001            0 9876509455
C00003 CHHAYA BANKAR MUMBAI        MAHARASTRA      400057          5000 9342567854
C00004 ASHWINI JOSHI BANGALORE     KARNATAKA       560001            0 9087654097
C00005 HANSEL COLACO  BANGALORE     KARNATAKA       400060          2000 9761234321
C00006 DEEPAK SHARMA MANGALORE     KARNATAKA       560050          1000 9087654009

6 rows selected.

SQL> DELETE FROM CLIENT_MASTER WHERE CITY = 'MANGALORE';

1 row deleted.

SQL> SELECT * FROM CLIENT_MASTER;

CLIENT NAME          CITY          STATE          PINCODE          BALDUE TELEPHONE
-----
C00001 IVAN BAYROSS    MUMBAI        MAHARASTRA      400054          1500 9867546456
C00002 MAMTAMUZUMDAR    MUMBAI        MAHARASTRA      780001            0 9876509455
C00003 CHHAYA BANKAR MUMBAI        MAHARASTRA      400057          5000 9342567854
C00004 ASHWINI JOSHI BANGALORE     KARNATAKA       560001            0 9087654097
C00005 HANSEL COLACO  BANGALORE     KARNATAKA       400060          2000 9761234321

SQL>
    
```

29) REMOVE SALESPERSON WHO STAY IN PUNE

```

Run SQL Command Line

SQL> SELECT * FROM SALES_MASTER;

SALENM SALESMANNA ADDRESS1 CITY PINCODE STATE SALESAMT
-----
S0001 AMAN BANDRA MUMBAI 400002 MAHARASTRA 45000
S0002 OMKAR NANMAN MUMBAI 400001 MAHARASTRA 25000
S0003 RAJ JUHI MUMBAI 400003 MAHARASTRA 30000
S0004 ASHISH KADAR PUNE 400044 MAHARASTRA 6000

SQL> DELETE FROM SALES_MASTER WHERE CITY = 'PUNE';

1 row deleted.

SQL> SELECT * FROM SALES_MASTER;

SALENM SALESMANNA ADDRESS1 CITY PINCODE STATE SALESAMT
-----
S0001 AMAN BANDRA MUMBAI 400002 MAHARASTRA 45000
S0002 OMKAR NANMAN MUMBAI 400001 MAHARASTRA 25000
S0003 RAJ JUHI MUMBAI 400003 MAHARASTRA 30000

SQL>
    
```

30) CHANGE PRESIDENT'S COMMISION TO 1000

```

Run SQL Command Line

SQL> SELECT * FROM EMP WHERE JOB = 'PRESIDENT';

ENO ENAME JOB MGR HIREDATE SAL COMM DEPTNO
-----
7839 KING PRESIDENT 09-JUN-90 5000 0 10

SQL> UPDATE EMP SET COMM = 1000 WHERE JOB = 'PRESIDENT';

1 row updated.

SQL> SELECT * FROM EMP WHERE JOB = 'PRESIDENT';

ENO ENAME JOB MGR HIREDATE SAL COMM DEPTNO
-----
7839 KING PRESIDENT 09-JUN-90 5000 1000 10

SQL>
    
```

31) INCREASE ALLEN SALARY BY 400 RUPEES

```

Run SQL Command Line
SQL> SELECT * FROM EMP WHERE ENAME = 'ALLEN';

  ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
 7499 ALLEN          SALESMAN         7698 15-AUG-98       1600      300       30

SQL> UPDATE EMP SET SAL = SAL+400 WHERE ENAME = 'ALLEN';

1 row updated.

SQL> SELECT * FROM EMP WHERE ENAME = 'ALLEN';

  ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
 7499 ALLEN          SALESMAN         7698 15-AUG-98       2000      300       30

SQL>
    
```

32) DISPLAY SAILORS WHOSE RATING IS GREATER THAN 7

```

Run SQL Command Line
SQL> SELECT * FROM SAILORS WHERE RATING > 7;

  SID SNAME          RATING  AGE
-----
 29 Brutus           8       33
 31 Lubber           8       55
 32 Andy             8       25
 58 Rusty           10      35
 71 Zorba           10      16
 74 Horatio          9       35
 65 smith            9       40

7 rows selected.

SQL>
    
```

33) DISPLAY SAILORS WITH 35 YEARS OLD

```

Run SQL Command Line
SQL> SELECT * FROM SAILORS WHERE AGE = 35;

  SID SNAME          RATING  AGE
-----
 58 Rusty           10      35
 64 Horatio          7       35
 74 Horatio          9       35

SQL>
    
```

EXERCISE NO. 3

AIM : TO ILLUSTRATE SQL FUNCTIONS

1) **DISPLAY TOTAL SALARY PAID TO EMPLOYEES**

```
Run SQL Command Line
SQL> SELECT SUM(SAL) FROM EMP;

SUM(SAL)
-----
 30252.5

SQL>
```

2) **DISPLAY TOTAL BALANCE DUE OF CLIENTS**

```
Run SQL Command Line
SQL> SELECT SUM(BALDUE) FROM CLIENT_MASTER;

SUM(BALDUE)
-----
      8500

SQL>
```

3) **DISPLAY AVERAGE SALARY PAID TO EMPLOYEES**

```
Run SQL Command Line
SQL> SELECT AVG(SAL) FROM EMP;

AVG(SAL)
-----
2160.89286

SQL>
```

4) DISPLAY MINIMUM SALARY PAID TO EMPLOYEES

```
Run SQL Command Line

SQL> SELECT MIN(SAL) FROM EMP;

  MIN(SAL)
-----
        800

SQL>
```

5) DISPLAY MAXIMUM SALARY PAID TO EMPLOYEES

```
Run SQL Command Line

SQL> SELECT MAX(SAL) FROM EMP;

  MAX(SAL)
-----
       5000

SQL>
```

6) DISPLAY MAXIMUM DUE BY THE CLIENTS

```
Run SQL Command Line

SQL> SELECT MAX(BALDUE) FROM CLIENT_MASTER;

MAX(BALDUE)
-----
       5000

SQL>
```

7) DISPLAY TOTAL NUMBER OF EMPLOYEES IN THE COMPANY

```
Run SQL Command Line

SQL> SELECT COUNT(*) FROM EMP;

  COUNT(*)
-----
        14

SQL>
```

8) DISPLAY ALL SAILOR NAMES IN UPPER CASE

```

Run SQL Command Line
SQL> SELECT SNAME, UPPER(SNAME) FROM SAILORS;

SNAME          UPPER(SNAME)
-----
Dustin         DUSTIN
Brutus        BRUTUS
Lubber        LUBBER
Andy          ANDY
Rusty         RUSTY
Horatio       HORATIO
Zorba         ZORBA
Horatio       HORATIO
Art           ART
Bob           BOB
smith         SMITH

11 rows selected.

SQL>
    
```

9) EXTRACT ONLY LAST DIGIT OF CLIENT NO.

```

Run SQL Command Line
SQL> SELECT CLIENTNO, SUBSTR(CLIENTNO,6,6) FROM CLIENT_MASTER;

CLIENT S
-----
C00001 1
C00002 2
C00003 3
C00004 4
C00005 5

SQL>
    
```

10) DISPLAY THE POSITION OF FIRST OCCURRENCE OF 'A' IN EMPLOYEE NAME

```

Run SQL Command Line
SQL> SELECT ENAME, INSTR(ENAME, 'A') FROM EMP;

ENAME          INSTR(ENAME, 'A')
-----
SMITH          0
ALLEN         1
WARD          2
JONES         0
BLAKE         3
CLARK         3
SCOTT         0
KING          0
TURNER        0
ADAMS         1
JAMES         2

ENAME          INSTR(ENAME, 'A')
-----
MILLER        0
FORD          0
MARTIN        2

14 rows selected.
    
```

11) DISPLAY LENGTH OF SAILORS NAME

```

Run SQL Command Line
SQL> SELECT SNAME,LENGTH(SNAME) FROM SAILORS;

SNAME                LENGTH(SNAME)
-----
Dustin                6
Brutus                6
Lubber                6
Andy                  4
Rusty                 5
Horatio               7
Zorba                 5
Horatio               7
Art                   3
Bob                   3
smith                 5

11 rows selected.

SQL>
    
```

12) REPLACE ALL OCCURRENCE OF 'A' IN EMPLOYEE NAME WITH 'Z'

```

Run SQL Command Line
SQL> SELECT ENAME,REPLACE(ENAME,'A','Z') FROM EMP;

ENAME                REPLACE(ENAME,'
-----
SMITH                SMITH
ALLEN                ZLLEN
WARD                 WZRD
JONES                JONES
BLAKE                BLZKE
CLARK                CLZRK
SCOTT                SCOTT
KING                 KING
TURNER               TURNER
ADAMS                ZDZMS
JAMES                JZMES

ENAME                REPLACE(ENAME,'
-----
MILLER               MILLER
FORD                 FORD
MARTIN               MZRTIN

14 rows selected.
    
```

13) ILLUSTRATE LTRIM AND RTRIM

```

Run SQL Command Line
SQL> SELECT LTRIM('$DATABASE','$'), RTRIM('DATABASE#','#') FROM DUAL;

LTRIM('$ RTRIM('D
-----
DATABASE DATABASE

SQL>
    
```

14) ILLUSTRATE LPAD AND RPAD

```

Run SQL Command Line
SQL> SELECT LPAD('DATABASE',10,'$'),RPAD('DATABASE',10,'$') FROM DUAL;

LPAD('DATA  RPAD('DATA
-----
$$DATABASE DATABASE$$

SQL>
    
```

15) ILLUSTRATE ALL NUMBER FUNCTIONS

```

Run SQL Command Line
SQL> SELECT ABS(-3),POWER(3,2),SQRT(49) FROM DUAL;

ABS(-3) POWER(3,2)  SQRT(49)
-----
3          9          7

SQL> SELECT CEIL(34.3),FLOOR(34.3) FROM DUAL;

CEIL(34.3) FLOOR(34.3)
-----
35          34

SQL> SELECT GREATEST(23,45,67), LEAST(23,45,67) FROM DUAL;

GREATEST(23,45,67) LEAST(23,45,67)
-----
67                  23

SQL> SELECT ROUND(45.6),ROUND(45.678,2),TRUNC(45.6),TRUNC(45.678,2) FRO
M DUAL;

ROUND(45.6) ROUND(45.678,2) TRUNC(45.6) TRUNC(45.678,2)
-----
46          45.68          45          45.67

SQL>
    
```

16) ILLUSTRATE ALL DATE FUNCTIONS

```
Run SQL Command Line

SQL> SELECT ADD_MONTHS('01-JUL-20',3) FROM DUAL;

ADD_MONTH
-----
01-OCT-20

SQL> SELECT MONTHS_BETWEEN('01-DEC-20','01-JUL-20') FROM DUAL;

MONTHS_BETWEEN('01-DEC-20','01-JUL-20')
-----
5

SQL> SELECT NEXT_DAY('01-JUL-20','THURSDAY') FROM DUAL;

NEXT_DAY(
-----
02-JUL-20

SQL> SELECT LAST_DAY('01-JUL-20') FROM DUAL;

LAST_DAY(
-----
31-JUL-20

SQL>
```

17) ILLUSTRATE CONVERSION FUNCTIONS

```
Run SQL Command Line

SQL> SELECT TO_CHAR(13000,'$99,999') FROM DUAL;

TO_CHAR(
-----
$13,000

SQL> SELECT TO_NUMBER(SUBSTR('$56784',2,4)) FROM DUAL;

TO_NUMBER(SUBSTR('$56784',2,4))
-----
5678

SQL> SELECT TO_DATE('06 JUL 2021') FROM DUAL;

TO_DATE('
-----
06-JUL-21

SQL> SELECT TO_CHAR(SYSDATE,'YYYY') FROM DUAL;

TO_C
----
2021

SQL> SELECT TO_CHAR(SYSDATE,'MON') FROM DUAL;

TO_
--
JUL

SQL> SELECT TO_CHAR(SYSDATE,'DAY') FROM DUAL;

TO_CHAR(S
-----
TUESDAY

SQL>
```

EXERCISE NO.4

AIM : TO ILLUSTRATE SQL CLAUSES LIKE DISTINCT, ORDER BY, GROUP BY AND HAVING

1) DISPLAY DISTINCT CLIENT CITIES

```
Run SQL Command Line
SQL> SELECT DISTINCT CITY FROM CLIENT_MASTER;

CITY
-----
MUMBAI
MANGALORE
BANGALORE

SQL>
```

2) DISPLAY EMPLOYEES DISTINCT DEPARTMENT NUMBERS

```
Run SQL Command Line
SQL> SELECT DISTINCT DEPTNO FROM EMP;

DEPTNO
-----
30
20
10

SQL>
```

3) DISPLAY SAILORS DISTINCT RATINGS

```
Run SQL Command Line
SQL> SELECT DISTINCT RATING FROM SAILORS;

RATING
-----
8
7
3
10
9

SQL>
```

4) DISPLAY EMPLOYEES IN AN APLABETIC ORDER

```

Run SQL Command Line
SQL> SET LINESIZE 100;
SQL> SELECT * FROM EMP ORDER BY ENAME;

```

ENO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7876	ADAMS	CLERK	7788	04-JUN-99	1100		20
7499	ALLEN	SALESMAN	7698	15-AUG-98	1600	300	30
7698	BLAKE	MANAGER	7839	11-JUN-92	3135		30
7782	CLARK	MANAGER	7839	14-MAY-93	2695		10
7902	FORD	ANALYST	7566	05-DEC-97	3000		20
7900	JAMES	CLERK	7698	23-JUN-00	950		30
7566	JONES	MANAGER	7839	31-OCT-95	3272.5		20
7839	KING	PRESIDENT		09-JUN-90	5000	0	10
7654	MARTIN	SALESMAN	7698	05-DEC-98	1250	1400	30
7934	MILLER	CLERK	7782	21-JAN-00	1300		10
7788	SCOTT	ANALYST	7566	05-MAR-96	3000		10
7369	SMITH	CLERK	7902	13-JUN-93	800	0	20
7844	TURNER	SALESMAN	7698	04-JUN-95	1500	0	30
7521	WARD	SALESMAN	7698	26-MAR-96	1250	500	30

```

14 rows selected.

```

5) DISPLAY EMPLOYEE'S SALARY IN DESCENDING ORDER

```

Run SQL Command Line
SQL> SELECT ENAME ,SAL FROM EMP ORDER BY SAL DESC;

```

ENAME	SAL
KING	5000
JONES	3272.5
BLAKE	3135
SCOTT	3000
FORD	3000
CLARK	2695
ALLEN	1600
TURNER	1500
MILLER	1300
MARTIN	1250
WARD	1250
ADAMS	1100
JAMES	950
SMITH	800

```

14 rows selected.

```

6) DISPLAY SAILORS RATINGS IN DESCENDING ORDER

```
Run SQL Command Line
SQL> SELECT * FROM SAILORS ORDER BY RATING DESC;

  SID  SNAME          RATING  AGE
-----
   58  Rusty           10      35
   71  Zorba           10      16
   74  Horatio         9       35
   65  smith           9       40
   29  Brutus          8       33
   31  Lubber          8       55
   32  Andy            8       25
   22  Dustin          7       45
   64  Horatio         7       35
   95  Bob             3       63
   85  Art             3       25

11 rows selected.

SQL>
```

7) DISPLAY SALESMAN NAMES IN ASCENDING ORDER

```
Run SQL Command Line
SQL> SELECT * FROM SALES_MASTER ORDER BY SALESMANNAME;

SALENM SALESMANNA ADDRESS1  CITY          PINCODE STATE  SALESAMT
-----
S0001  AMAN          BANDRA    MUMBAI        400002 MAHARASTRA  45000
S0004  ASHISH        KADAR     PUNE           400044 MAHARASTRA  6000
S0002  OMKAR        NANMAN    MUMBAI        400001 MAHARASTRA  25000
S0003  RAJ           JUHI      MUMBAI        400003 MAHARASTRA  30000

SQL>
```

8) DISPLAY MAXIMUM SALARY IN EACH DEPARTMENT

```
Run SQL Command Line
SQL> SELECT DEPTNO,MAX(SAL) FROM EMP GROUP BY DEPTNO;

  DEPTNO  MAX(SAL)
-----
    30      3135
    20     3272.5
    10      5000

SQL>
```

9) DISPLAY TOTAL SALARY PAID TO EACH DEPARTMENT

```
Run SQL Command Line
SQL> SELECT DEPTNO,SUM(SAL) FROM EMP GROUP BY DEPTNO;

  DEPTNO    SUM(SAL)
-----
      30      9685
      20     8172.5
      10     11995

SQL>
```

10) DISPLAY NUMBER OF EMPLOYEES IN EACH DEPARTMENT

```
Run SQL Command Line
SQL> SELECT DEPTNO,COUNT(*) FROM EMP GROUP BY DEPTNO;

  DEPTNO    COUNT(*)
-----
      30         6
      20         4
      10         4

SQL>
```

11) DISPLAY MAXIMUM SALARY PAID TO EACH DESIGNATION

```
Run SQL Command Line
SQL> SELECT JOB,MAX(SAL) FROM EMP GROUP BY JOB;

JOB          MAX(SAL)
-----
CLERK             1300
SALESMAN          1600
PRESIDENT         5000
MANAGER           3272.5
ANALYST           3000

SQL>
```

12) DISPLAY NUMBER OF EMPLOYEES IN EACH DESIGNATION

```
Run SQL Command Line
SQL> SELECT JOB,COUNT(*) FROM EMP GROUP BY JOB;

JOB                COUNT(*)
-----
CLERK                4
SALESMAN             4
PRESIDENT            1
MANAGER              3
ANALYST              2

SQL>
```

13) DISPLAY NUMBER OF SAILORS IN EACH RATING

```
Run SQL Command Line
SQL> SELECT RATING, COUNT(*) FROM SAILORS GROUP BY RATING;

RATING  COUNT(*)
-----
      8         3
      7         2
      3         2
     10         2
      9         2

SQL>
```

14) DISPLAY MINIMUM SALARY IN EACH DESIGNATION

```
Run SQL Command Line
SQL> SELECT JOB,MIN(SAL) FROM EMP GROUP BY JOB;

JOB                MIN(SAL)
-----
CLERK                800
SALESMAN             1250
PRESIDENT            5000
MANAGER              2695
ANALYST              3000

SQL>
```

15) DISPLAY DEPARTMENT NUMBER THAT HAS MORE THAN 2 EMPLOYEES

```
Run SQL Command Line

SQL> SELECT DEPTNO, COUNT(*) FROM EMP GROUP BY DEPTNO HAVING COUNT(*)>2;

  DEPTNO  COUNT(*)
-----
      30         6
      20         4
      10         4

SQL>
```

16) DISPLAY RATINGS WHICH HAS MORE THAN 2 SAILORS

```
Run SQL Command Line

SQL> SELECT RATING, COUNT(*) FROM SAILORS GROUP BY RATING HAVING COUNT(*) > 2;

  RATING  COUNT(*)
-----
       8         3

SQL>
```

EXERCISE NO.5

AIM : TO ILLUSTRATE SQL CONSTRAINTS

1) CREATE A TABLE CALLED 'PERSON' WITH THE FOLLOWING STRUCTURE(USING SYSTEM GENERATED CONSTRAINTS)

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
PERSON_ID	VARCHAR2	10	PRIMARY KEY
FNAME	VARCHAR2	5	UNIQUE(TABLE LEVEL CONSTRAINT)
LNAME	VARCHAR2	15	
CITY	VARCHAR2	15	NOT NULL
AGE	NUMBER	2	<=100
MOBILE_NO	NUMBER	10	UNIQUE

```

Run SQL Command Line
SQL> CREATE TABLE PERSON(PERSON_ID VARCHAR2(10) PRIMARY KEY, FNAME VARCHAR2(5),L
NAME VARCHAR2(15),CITY VARCHAR2(15) NOT NULL,AGE NUMBER(2) CHECK (AGE<=100),MOBI
LE_NO NUMBER(10) UNIQUE, UNIQUE(FNAME,LNAME));

Table created.

SQL>

```

2) DISPLAY THE CONSTRAINTS OF 'PERSON' TABLE

```

Run SQL Command Line
SQL> SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'PERSON';

CONSTRAINT_NAME          C
-----
SYS_C004466              C
SYS_C004467              C
SYS_C004468              P
SYS_C004469              U
SYS_C004470              U

SQL>

```

3) DROP THE UNIQUE CONSTRAINT ATTACHED TO MOBILE_NO

```

Run SQL Command Line
SQL> SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'PERSON';

CONSTRAINT_NAME          C
-----
SYS_C004466              C
SYS_C004467              C
SYS_C004468              P
SYS_C004469              U
SYS_C004470              U

SQL> ALTER TABLE PERSON DROP CONSTRAINT SYS_C004469;

Table altered.

SQL> SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'PERSON';

CONSTRAINT_NAME          C
-----
SYS_C004466              C
SYS_C004467              C
SYS_C004468              P
SYS_C004470              U

SQL>
    
```

NOTE : DIFFICULT TO IDENTIFY THE UNIQUE CONSTRAINT OF MOBILE_NO

4) CREATE A TABLE CALLED 'PERSON1' WITH THE FOLLOWING STRUCTURE(USING USER DEFINED CONSTRAINTS)

COLUMN NAME	DATATYPE	SIZE	CONSTRAINTS
PERSON_ID	VARCHAR2	10	PRIMARY KEY
PERSON_NAME	VARCHAR2	5	UNIQUE
CITY	VARCHAR2	15	NOT NULL
AGE	NUMBER	2	<=100
MOBILE_NO	NUMBER	10	UNIQUE

```

Run SQL Command Line
SQL> CREATE TABLE PERSON1(PERSON_ID VARCHAR2(10) CONSTRAINT PK PRIMARY KEY,PERSON_NAME VARCHAR2(
15) CONSTRAINT NAME_UK UNIQUE,CITY VARCHAR2(15) CONSTRAINT CITY_NN NOT NULL,AGE NUMBER(2) CONSTR
AINT AGE_CK CHECK(AGE<=100),MOBILE_NO NUMBER(10) CONSTRAINT MOBILE_UK UNIQUE);

Table created.

SQL>
    
```

5) DISPLAY THE CONSTRAINTS OF 'PERSON1' TABLE

```

Run SQL Command Line
SQL> SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'PERSON1';

CONSTRAINT_NAME          C
-----
CITY_NN                   C
AGE_CK                    C
PK                         P
NAME_UK                   U
MOBILE_UK                 U

SQL>
    
```

NOTE : EASY TO DROP A CONSTRAINT IF IT IS A USER DEFINED CONSTRAINT

6) DROP THE UNIQUE CONSTRAINT OF MOBILE_NO

```

Run SQL Command Line
SQL> SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'PERSON1';

CONSTRAINT_NAME          C
-----
CITY_NN                   C
AGE_CK                    C
PK                         P
NAME_UK                   U
MOBILE_UK                 U

SQL> ALTER TABLE PERSON1 DROP CONSTRAINT MOBILE_UK;

Table altered.

SQL> SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE FROM USER_CONSTRAINTS WHERE TABLE_NAME = 'PERSON1';

CONSTRAINT_NAME          C
-----
CITY_NN                   C
AGE_CK                    C
PK                         P
NAME_UK                   U

SQL>
    
```

7) CREATE A TABLE CALLED 'COURSES' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE
COURSE_ID	VARCHAR2	10
COURSE_NAME	VARCHAR2	20

```
Run SQL Command Line
SQL> CREATE TABLE COURSES(CORSE_ID VARCHAR2(10), COURSE_NAME VARCHAR2(20));
Table created.
SQL>
```

8) CREATE A TABLE CALLED 'STUDENT' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATATYPE	SIZE
ROLLNO	NUMBER	6
STUNAME	VARCHAR2	15
COURSE_ID	VARCHAR2	10
COLLEGENAME	VARCHAR2	15

```
Run SQL Command Line
SQL> CREATE TABLE STUDENT(ROLLNO NUMBER(6),STUNAME VARCHAR2(15),COURSE_ID VARCHAR2(10),COLLEGENA^
ME VARCHAR2(15));
Table created.
SQL>
```

9) ADD PRIMARY KEY TO THE 'COURSE_ID' COLUMN OF 'COURSES' TABLE

```
Run SQL Command Line
SQL> ALTER TABLE COURSES RENAME COLUMN CORSE_ID TO COURSE_ID;
Table altered.
SQL> ALTER TABLE COURSES ADD PRIMARY KEY(COURSE_ID);
Table altered.
SQL>
```

10) ADD UNIQUE CONSTRAINT TO 'COURSE_NAME' COLUMN OF 'COURSES' TABLE USING USER DEFINED CONSTRAINT

```
Run SQL Command Line
SQL> ALTER TABLE COURSES ADD UNIQUE(COURSE_NAME);
Table altered.
SQL>
```

11) ADD PRIMARY KEY TO THE COLUMN 'ROLLNO' OF 'STUDENT' TABLE USING USER DEFINED CONSTRAINT

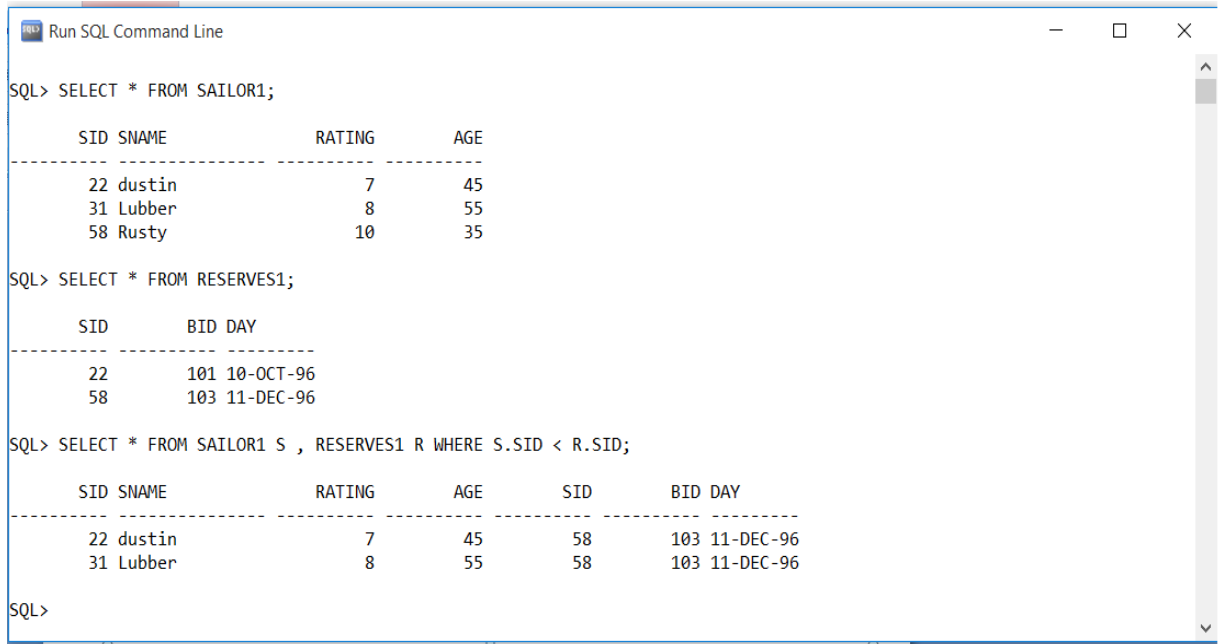
```
Run SQL Command Line
SQL> ALTER TABLE STUDENT ADD CONSTRAINT ROLLNO_PK PRIMARY KEY(ROLLNO);
Table altered.
SQL>
```

12) ADD FOREIGN KEY TO 'COURSE_ID' OF 'STUDENT' TABLE

```
Run SQL Command Line
SQL> ALTER TABLE STUDENT ADD FOREIGN KEY (COURSE_ID) REFERENCES COURSES(COURSE_ID);
Table altered.
SQL>
```

13) ADD NOT NULL CONSTRAINT TO 'COLLEGENAME' COLUMN OF 'STUDENT' TABLE

```
Run SQL Command Line
SQL> ALTER TABLE STUDENT MODIFY COLLEGENAME VARCHAR2(15) NOT NULL;
Table altered.
SQL>
```

EXERCISE NO.6**AIM : TO ILLUSTRATE SQL JOINS****1) INNER JOIN****1.1) THETA JOIN**

```
Run SQL Command Line
SQL> SELECT * FROM SAILOR1;

  SID SNAME      RATING  AGE
-----
   22 dustin         7    45
   31 Lubber         8    55
   58 Rusty        10    35

SQL> SELECT * FROM RESERVES1;

  SID  BID DAY
-----
   22  101 10-OCT-96
   58  103 11-DEC-96

SQL> SELECT * FROM SAILOR1 S , RESERVES1 R WHERE S.SID < R.SID;

  SID SNAME      RATING  AGE  SID  BID DAY
-----
   22 dustin         7    45   58  103 11-DEC-96
   31 Lubber         8    55   58  103 11-DEC-96

SQL>
```

1.2) EQUI JOIN

```
Run SQL Command Line

SQL> SELECT * FROM SAILOR1;

  SID SNAME      RATING  AGE
-----
  22 dustin        7     45
  31 Lubber        8     55
  58 Rusty        10     35

SQL> SELECT * FROM RESERVES1;

  SID  BID DAY
-----
  22   101 10-OCT-96
  58   103 11-DEC-96

SQL> SELECT * FROM SAILOR1 S , RESERVES1 R WHERE S.SID = R.SID;

  SID SNAME      RATING  AGE  SID  BID DAY
-----
  22 dustin        7     45   22   101 10-OCT-96
  58 Rusty        10     35   58   103 11-DEC-96

SQL>
```

1.3) NATURAL JOIN

```
Run SQL Command Line

SQL> SELECT * FROM SAILOR1;

  SID SNAME          RATING  AGE
-----
  22 dustin           7       45
  31 Lubber           8       55
  58 Rusty           10      35

SQL> SELECT * FROM RESERVES1;

  SID  BID DAY
-----
  22   101 10-OCT-96
  58   103 11-DEC-96

SQL> SELECT * FROM SAILOR1 S NATURAL JOIN RESERVES1 R;

  SID SNAME          RATING  AGE  BID DAY
-----
  22 dustin           7       45  101 10-OCT-96
  58 Rusty           10      35  103 11-DEC-96

SQL>
```

2) OUTER JOIN

2.1) LEFT OUTER JOIN

```
Select Run SQL Command Line

SQL> SELECT * FROM DEPT;

  DEPTNO DEPTNAME          DLOCATION
-----
  10 ACCOUNTING          NEW YORK
  20 RESEARCH            DALLAS
  30 SALES                CHICAGO
  40 OPERATIONS          BOSTON

SQL> SELECT * FROM EMPLOYEE;

  ENO ENAME          SAL  DEPTNO
-----
  7369 SMITH          800   20
  7499 ALLEN         1600  30
  7521 WARD          1250  30
  7566 JONES         3272.5 50
  7782 CLARK         2695  10

SQL> SELECT * FROM DEPT D LEFT OUTER JOIN EMPLOYEE E ON D.DEPTNO = E.DEPTNO;

  DEPTNO DEPTNAME          DLOCATION  ENO ENAME          SAL  DEPTNO
-----
  20 RESEARCH            DALLAS   7369 SMITH          800   20
  30 SALES                CHICAGO  7499 ALLEN         1600  30
  30 SALES                CHICAGO  7521 WARD          1250  30
  10 ACCOUNTING          NEW YORK  7782 CLARK         2695  10
  40 OPERATIONS          BOSTON

SQL>
```

2.2) RIGHT OUTER JOIN

```
Select Run SQL Command Line
SQL> SELECT * FROM DEPT;

  DEPTNO DEPTNAME          DLOCATION
-----
    10 ACCOUNTING          NEW YORK
    20 RESEARCH             DALLAS
    30 SALES                 CHICAGO
    40 OPERATIONS           BOSTON

SQL> SELECT * FROM EMPLOYEE;

  ENO ENAME          SAL   DEPTNO
-----
  7369 SMITH           800     20
  7499 ALLEN          1600    30
  7521 WARD            1250    30
  7566 JONES          3272.5  50
  7782 CLARK           2695    10

SQL> SELECT * FROM DEPT D RIGHT OUTER JOIN EMPLOYEE E ON D.DEPTNO = E.DEPTNO;

  DEPTNO DEPTNAME          DLOCATION          ENO ENAME          SAL   DEPTNO
-----
    20 RESEARCH             DALLAS           7369 SMITH           800     20
    30 SALES                 CHICAGO          7499 ALLEN          1600    30
    30 SALES                 CHICAGO          7521 WARD            1250    30
    10 ACCOUNTING           NEW YORK         7566 JONES          3272.5  50
    7782 CLARK           2695    10

SQL>
```

2.3) FULL OUTER JOIN

```
Select Run SQL Command Line
SQL> SELECT * FROM DEPT;

  DEPTNO DEPTNAME          DLOCATION
-----
    10 ACCOUNTING          NEW YORK
    20 RESEARCH             DALLAS
    30 SALES                 CHICAGO
    40 OPERATIONS           BOSTON

SQL> SELECT * FROM EMPLOYEE;

  ENO ENAME          SAL   DEPTNO
-----
  7369 SMITH           800     20
  7499 ALLEN          1600    30
  7521 WARD            1250    30
  7566 JONES          3272.5  50
  7782 CLARK           2695    10

SQL> SELECT * FROM DEPT D FULL OUTER JOIN EMPLOYEE E ON D.DEPTNO = E.DEPTNO;

  DEPTNO DEPTNAME          DLOCATION          ENO ENAME          SAL   DEPTNO
-----
    20 RESEARCH             DALLAS           7369 SMITH           800     20
    30 SALES                 CHICAGO          7499 ALLEN          1600    30
    30 SALES                 CHICAGO          7521 WARD            1250    30
    10 ACCOUNTING           NEW YORK         7782 CLARK           2695    10
    40 OPERATIONS           BOSTON
    7566 JONES          3272.5  50

6 rows selected.

SQL>
```

3) SELF JOIN

```

Select Run SQL Command Line
SQL> SELECT * FROM EMP;

      ENO ENAME          JOB              MGR HIREDATE          SAL        COMM        DEPTNO
-----
7369 SMITH             CLERK              7902 13-JUN-93          800           0           20
7499 ALLEN             SALESMAN           7698 15-AUG-98         1600          300          30
7521 WARD              SALESMAN           7698 26-MAR-96         1250          500          30
7566 JONES             MANAGER            7839 31-OCT-95         3272.5         0           20
7698 BLAKE             MANAGER            7839 11-JUN-92          3135           0           30
7782 CLARK             MANAGER            7839 14-MAY-93          2695           0           10
7788 SCOTT            ANALYST            7566 05-MAR-96          3000           0           10
7839 KING              PRESIDENT          09-JUN-90         5000           0           10
7844 TURNER           SALESMAN           7698 04-JUN-95          1500           0           30
7876 ADAMS            CLERK              7788 04-JUN-99          1100           0           20
7900 JAMES            CLERK              7698 23-JUN-00           950           0           30

      ENO ENAME          JOB              MGR HIREDATE          SAL        COMM        DEPTNO
-----
7934 MILLER           CLERK              7782 21-JAN-00          1300           0           10
7902 FORD              ANALYST            7566 05-DEC-97          3000           0           20
7654 MARTIN           SALESMAN           7698 05-DEC-98          1250          1400         30

14 rows selected.

SQL> SELECT E.ENAME,M.ENAME FROM EMP E, EMP M WHERE E.MGR = M.ENO;

ENAME          ENAME
-----
SMITH          FORD
ALLEN          BLAKE
WARD           BLAKE
JONES          KING
BLAKE          KING
CLARK          KING
SCOTT          JONES
TURNER         BLAKE
ADAMS          SCOTT
JAMES          BLAKE
MILLER         CLARK

ENAME          ENAME
-----
FORD           JONES
MARTIN         BLAKE
    
```

4) NON EQUI JOIN

```

Run SQL Command Line
SQL> SELECT * FROM EMPLOYEE;

      ENO ENAME          SAL        DEPTNO
-----
7369 SMITH             800         20
7499 ALLEN            1600        30
7521 WARD             1250        30
7566 JONES            3272.5      50
7782 CLARK            2695        10

SQL> SELECT * FROM SALGRADE;

      GRADE      LOSAL      HISAL
-----
1           700       1200
2          1201      1400
3          1401      2000
4          2001      3000
5          3001      9999

SQL> SELECT ENO, ENAME,SAL,GRADE FROM EMPLOYEE E, SALGRADE S WHERE E.SAL BETWEEN LOSAL AND HISAL;

      ENO ENAME          SAL        GRADE
-----
7369 SMITH             800         1
7499 ALLEN            1600        3
7521 WARD             1250        2
7566 JONES            3272.5      5
7782 CLARK            2695        4

SQL>
    
```

QUERIES ON JOINS

1) GET EMPLOYEES WITH THEIR DEPARTMENT NAMES

```

Select Run SQL Command Line
SQL> SELECT * FROM DEPT;

  DEPTNO DEPTNAME          DLOCATION
-----
      10 ACCOUNTING          NEW YORK
      20 RESEARCH           DALLAS
      30 SALES               CHICAGO
      40 OPERATIONS          BOSTON

SQL> SELECT * FROM EMPLOYEE;

  ENO ENAME          SAL      DEPTNO
-----
  7369 SMITH            800        20
  7499 ALLEN          1600       30
  7521 WARD            1250       30
  7566 JONES          3272.5     50
  7782 CLARK           2695       10

SQL> SELECT ENO,ENAME,E.DEPTNO,SAL,DEPTNAME,DLOCATION FROM EMPLOYEE E, DEPT D WHERE E.DEPTNO = D.DEPTNO;

  ENO ENAME          DEPTNO      SAL DEPTNAME          DLOCATION
-----
  7369 SMITH            20          800 RESEARCH          DALLAS
  7499 ALLEN           30         1600 SALES             CHICAGO
  7521 WARD            30         1250 SALES             CHICAGO
  7782 CLARK            10         2695 ACCOUNTING        NEW YORK

SQL>
    
```

2) GET SAILORS WHO HAVE RESERVED BOATS

```

Run SQL Command Line
SQL> SELECT * FROM SAILORS;

  SID SNAME          RATING      AGE
-----
   22 Dustin            7           45
   29 Brutus           8           33
   31 Lubber           8           55
   32 Andy             8           25
   58 Rusty           10          35
   64 Horatio          7           35
   71 Zorba            10          16
   74 Horatio          9           35
   85 Art              3           25
   95 Bob              3           63
   65 smith            9           40

11 rows selected.

SQL> SELECT * FROM RESERVES;

  SID      BID DAY
-----
   22      101 10-OCT-98
   22      102 10-OCT-98
   22      103 10-AUG-98
   22      104 10-JUL-98
   31      102 11-OCT-98
   31      103 11-JUN-98
   31      104 11-DEC-98
   64      101 09-MAY-98
   64      102 09-AUG-98
   74      103 09-AUG-98

10 rows selected.

SQL> SELECT * FROM SAILORS NATURAL JOIN RESERVES;

  SID SNAME          RATING      AGE      BID DAY
-----
   22 Dustin            7           45      101 10-OCT-98
   22 Dustin            7           45      102 10-OCT-98
   22 Dustin            7           45      103 10-AUG-98
   22 Dustin            7           45      104 10-JUL-98
   31 Lubber           8           55      102 11-OCT-98
   31 Lubber           8           55      103 11-JUN-98
   31 Lubber           8           55      104 11-DEC-98
   64 Horatio          7           35      101 09-MAY-98
   64 Horatio          7           35      102 09-AUG-98
   74 Horatio          9           35      103 09-AUG-98
    
```

3) DISPLAY SAILOR NAMES WHO HAVE RESERVED BOATS

```

Run SQL Command Line
SQL> SELECT * FROM SAILORS;

  SID  SNAME      RATING  AGE
-----
  22  Dustin      7        45
  29  Brutus     8        33
  31  Lubber     8        55
  32  Andy       8        25
  58  Rusty     10        35
  64  Horatio    7        35
  71  Zorba     10        16
  74  Horatio    9        35
  85  Art        3        25
  95  Bob        3        63
  65  smith     9        40

11 rows selected.

SQL> SELECT * FROM RESERVES;

  SID  BID  DAY
-----
  22  101 10-OCT-98
  22  102 10-OCT-98
  22  103 10-AUG-98
  22  104 10-JUL-98
  31  102 11-OCT-98
  31  103 11-JUN-98
  31  104 11-DEC-98
  64  101 09-MAY-98
  64  102 09-AUG-98
  74  103 09-AUG-98

10 rows selected.

SQL> SELECT DISTINCT SNAME FROM SAILORS NATURAL JOIN RESERVES;

SNAME
-----
Lubber
Dustin
Horatio
    
```

4) FIND SAILORS WHO HAVE RESERVED 102 BOAT

```

Run SQL Command Line

SQL> SELECT DISTINCT SNAME FROM SAILORS S ,RESERVES R WHERE S.SID = R.SID AND BID = 103;

SNAME
-----
Lubber
Dustin
Horatio

SQL>
    
```

5) DISPLAY SAILOR NAMES WHO HAVE RESERVED RED BOATS

```

Run SQL Command Line

SQL> SELECT * FROM BOATS;

  BID  BNAME      BCOLOR
-----
  101  Interlake  blue
  102  Interlake  red
  103  Clipper    green
  104  Marine     red

SQL> SELECT SNAME,BCOLOR FROM SAILORS S, RESERVES R, BOATS B WHERE S.SID=R.SID AND R.BID = B.BID AND B.BCOLOR = 'red';

SNAME      BCOLOR
-----
Dustin     red
Dustin     red
Lubber     red
Lubber     red
Horatio    red

SQL>
    
```

6) FIND SAILORS WHO HAVE RESERVED 'INTERLAKE' BOATS

```

Run SQL Command Line
SQL> SELECT * FROM BOATS;

  BID  BNAME      BCOLOR
-----
  101  Interlake   blue
  102  Interlake   red
  103  Clipper     green
  104  Marine      red

SQL> SELECT SNAME,BNAME FROM SAILORS S, RESERVES R, BOATS B WHERE S.SID=R.SID AND R.BID = B.BID AND B.BNAME = 'Interlake';

SNAME      BNAME
-----
Dustin     Interlake
Dustin     Interlake
Lubber     Interlake
Horatio    Interlake
Horatio    Interlake

SQL>
    
```

7) FIND THE BOATS RESERVED BY 'LUBBER'

```

Run SQL Command Line
SQL> SELECT BNAME FROM SAILORS S, RESERVES R, BOATS B WHERE SNAME = 'Lubber' AND S.SID = R.SID AND R.BID = B.BID;

BNAME
-----
Interlake
Clipper
Marine

SQL>
    
```

8) FIND THE COLOR OF BOATS RESERVED BY 'DUSTIN'

```

Run SQL Command Line
SQL> SELECT BCOLOR FROM SAILORS S, RESERVES R, BOATS B WHERE SNAME = 'Dustin' AND S.SID = R.SID AND R.BID = B.BID;

BCOLOR
-----
blue
red
green
red

SQL>
    
```

EXERCISE NO 8

AIM : TO ILLUSTRATE SQL VIEWS

1) CREATING A VIEW WITHOUT EXCLUDING A PRIMARY KEY ATTRIBUTE AND PERFORMING OPERATIONS ON SUCH VIEW

```
Run SQL Command Line

SQL> SELECT * FROM EMPLOYEE;

      ENO ENAME          SAL      DEPTNO
-----
7369 SMITH              800        20
7499 ALLEN             1600        30
7521 WARD              1250        30
7566 JONES             3272.5      50
7782 CLARK             2695        10

SQL> CREATE VIEW SALHIDING AS (SELECT ENO,ENAME,DEPTNO FROM EMPLOYEE);

View created.

SQL> SELECT * FROM SALHIDING;

      ENO ENAME          DEPTNO
-----
7369 SMITH              20
7499 ALLEN              30
7521 WARD              30
7566 JONES              50
7782 CLARK              10

SQL>
```

```
Run SQL Command Line

SQL> CREATE VIEW SALHIDING AS (SELECT ENO,ENAME,DEPTNO FROM EMPLOYEE);

View created.

SQL> SELECT * FROM SALHIDING;

      ENO ENAME          DEPTNO
-----
7369 SMITH              20
7499 ALLEN              30
7521 WARD              30
7566 JONES              50
7782 CLARK              10

SQL> INSERT INTO SALHIDING VALUES(7689,'IVAN',20);

1 row created.

SQL> UPDATE SALHIDING SET DEPTNO = 10 WHERE ENAME = 'SMITH';

1 row updated.

SQL> DELETE FROM SALHIDING WHERE ENAME='ALLEN';

1 row deleted.

SQL>
```

2) CREATING A VIEW BY EXCLUDING THE PRIMARY KEY ATTRIBUTE(ENO) AND PERFORMING OPERATIONS ON SUCH VIEW

```

Run SQL Command Line

SQL> CREATE VIEW SALHIDING1 AS (SELECT ENAME,DEPTNO FROM EMPLOYEE);
View created.

SQL> SELECT * FROM SALHIDING1;

ENAME          DEPTNO
-----
SMITH           10
WARD            30
JONES           50
CLARK           10
IVAN            20

SQL> INSERT INTO SALHIDING1 VALUES('BAYROSS',20);
INSERT INTO SALHIDING1 VALUES('BAYROSS',20)
*
ERROR at line 1:
ORA-01400: cannot insert NULL into ("SCOTT"."EMPLOYEE"."ENO")

SQL> UPDATE SALHIDING1 SET DEPTNO = 20 WHERE ENAME ='SMITH';
1 row updated.

SQL> DELETE FROM SALHIDING1 WHERE ENAME='WARD';
1 row deleted.

SQL>
    
```

3) CREATE A VIEW FROM MULTIPLE TABLES WHICH HAVE A REFERENCING CLASS.

```

Run SQL Command Line

SQL> SELECT * FROM DEPT;

DEPTNO DEPTNAME      DLOCATION
-----
10 ACCOUNTING     NEW YORK
20 RESEARCH      DALLAS
30 SALES         CHICAGO
40 OPERATIONS    BOSTON

SQL> SELECT * FROM EMPLOYEE;

ENO ENAME          SAL      DEPTNO
-----
7369 SMITH          800      20
7566 JONES        3272.5   50
7782 CLARK        2695     10
7689 IVAN         2000     20

SQL> CREATE VIEW VIEW4 AS (SELECT * FROM EMPLOYEE E NATURAL JOIN DEPT D);
View created.

SQL> SELECT * FROM VIEW4;

DEPTNO      ENO ENAME          SAL DEPTNAME      DLOCATION
-----
20          7369 SMITH          800 RESEARCH      DALLAS
10          7782 CLARK        2695 ACCOUNTING    NEW YORK
20          7689 IVAN         2000 RESEARCH      DALLAS

SQL>
    
```

PERFORM OPERATIONS

```

Run SQL Command Line

SQL> INSERT INTO VIEW4 VALUES (10,7555,'BAYROSS',400,'ACCOUNTING','NEW YORK');
INSERT INTO VIEW4 VALUES (10,7555,'BAYROSS',400,'ACCOUNTING','NEW YORK')
*
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table

SQL> UPDATE VIEW4 SET SAL = 600 WHERE ENAME = 'IVAN';

1 row updated.

SQL> DELETE FROM VIEW4 WHERE ENAME = 'CLARK';

1 row deleted.

SQL> UPDATE VIEW4 SET DEPTNO = 10 WHERE ENAME ='IVAN';
UPDATE VIEW4 SET DEPTNO = 10 WHERE ENAME ='IVAN'
*
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table

SQL>
    
```

4) CREATE A VIEW FROM MULTIPLE TABLES WHICH DOES NOT HAVE A REFERENCING CLASS.

```

Run SQL Command Line

SQL> SELECT * FROM EMPLOYEE;

      ENO ENAME          SAL      DEPTNO
-----
      7369 SMITH           800         20
      7566 JONES          3272.5      50
      7689 IVAN           600         20

SQL> SELECT * FROM SALGRADE;

      GRADE      LOSAL      HISAL
-----
          1         700      1200
          2        1201      1400
          3        1401      2000
          4        2001      3000
          5        3001      9999

SQL> CREATE VIEW VIEWS AS (SELECT ENO,ENAME,SAL,GRADE FROM EMPLOYEE ,SALGRADE WHERE SAL BETWEEN LOSAL AND HISAL);

View created.

SQL> SELECT * FROM VIEWS;

      ENO ENAME          SAL      GRADE
-----
      7369 SMITH           800         1
      7566 JONES          3272.5      5

SQL>
    
```

PERFORMING OPERATIONS

```

Run SQL Command Line

SQL> SELECT * FROM VIEWS;

      ENO ENAME          SAL      GRADE
-----
7369 SMITH              800         1
7566 JONES             3272.5       5

SQL> INSERT INTO VIEWS VALUES(7657,'IVAN',900,1);
INSERT INTO VIEWS VALUES(7657,'IVAN',900,1)
*
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table

SQL> UPDATE VIEWS SET SAL = 900 WHERE ENAME = 'SMITH';
UPDATE VIEWS SET SAL = 900 WHERE ENAME = 'SMITH'
*
ERROR at line 1:
ORA-01779: cannot modify a column which maps to a non key-preserved table

SQL> DELETE FROM VIEWS WHERE ENAME = 'JONES';
DELETE FROM VIEWS WHERE ENAME = 'JONES'
*
ERROR at line 1:
ORA-01752: cannot delete from view without exactly one key-preserved table

SQL>
    
```

5) CREATE A ROW SECURITY VIEW

```

Run SQL Command Line

SQL> SELECT * FROM EMP;

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
7369 SMITH      CLERK              7902 13-JUN-93          800         0         20
7499 ALLEN      SALESMAN           7698 15-AUG-98         1600        300        30
7521 WARD        SALESMAN           7698 26-MAR-96         1250        500        30
7566 JONES      MANAGER            7839 31-OCT-95         3272.5       0         20
7698 BLAKE      MANAGER            7839 11-JUN-92          3135         0         30
7782 CLARK      MANAGER            7839 14-MAY-93          2695         0         10
7788 SCOTT      ANALYST            7566 05-MAR-96          3000         0         10
7839 KING      PRESIDENT          09-JUN-90         5000         0         10
7844 TURNER     SALESMAN           7698 04-JUN-95          1500         0         30
7876 ADAMS      CLERK              7788 04-JUN-99          1100         0         20
7900 JAMES      CLERK              7698 23-JUN-00           950         0         30

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
7934 MILLER     CLERK              7782 21-JAN-00          1300         0         10
7902 FORD        ANALYST            7566 05-DEC-97          3000         0         20
7654 MARTIN     SALESMAN           7698 05-DEC-98         1250        1400        30

14 rows selected.

SQL> CREATE VIEW VIEW6 AS (SELECT * FROM EMP WHERE DEPTNO =10);

View created.

SQL> SELECT * FROM VIEW6;

      ENO ENAME          JOB              MGR HIREDATE          SAL      COMM      DEPTNO
-----
7782 CLARK      MANAGER            7839 14-MAY-93          2695         0         10
7788 SCOTT      ANALYST            7566 05-MAR-96          3000         0         10
7839 KING      PRESIDENT          09-JUN-90         5000         0         10
7934 MILLER     CLERK              7782 21-JAN-00          1300         0         10

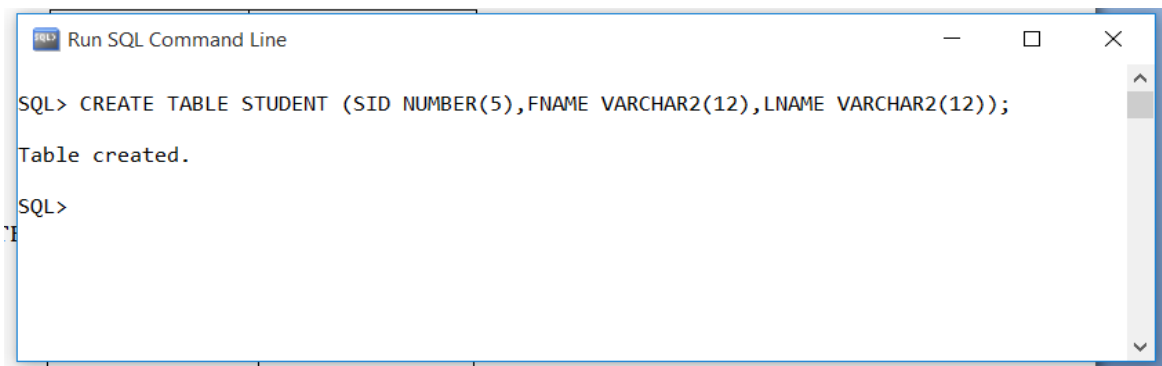
SQL>
    
```

Exercise No:7

Aim : To Illustrate SQL Sequence and Set Operators

1) CREATE A TABLE CALLED 'STUDENT' WITH THE FOLLOWING STRUCTURE

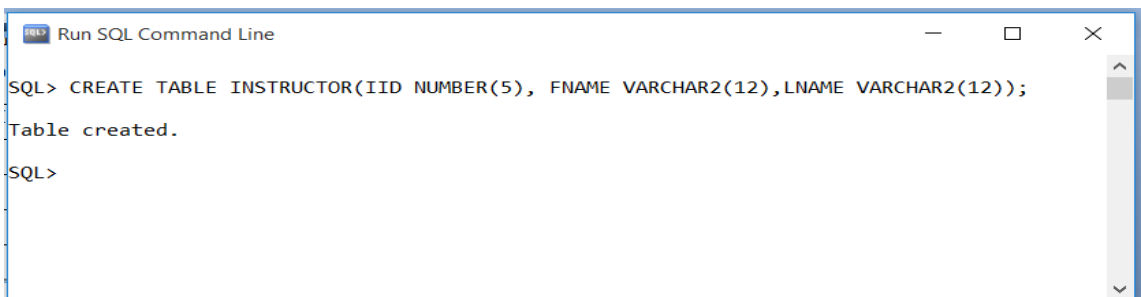
COLUMN NAME	DATA TYPE
SID	NUMBER(5)
FNAME	VARCHAR2(12)
LNAME	VARCHAR2(12)



```
Run SQL Command Line
SQL> CREATE TABLE STUDENT (SID NUMBER(5),FNAME VARCHAR2(12),LNAME VARCHAR2(12));
Table created.
SQL>
```

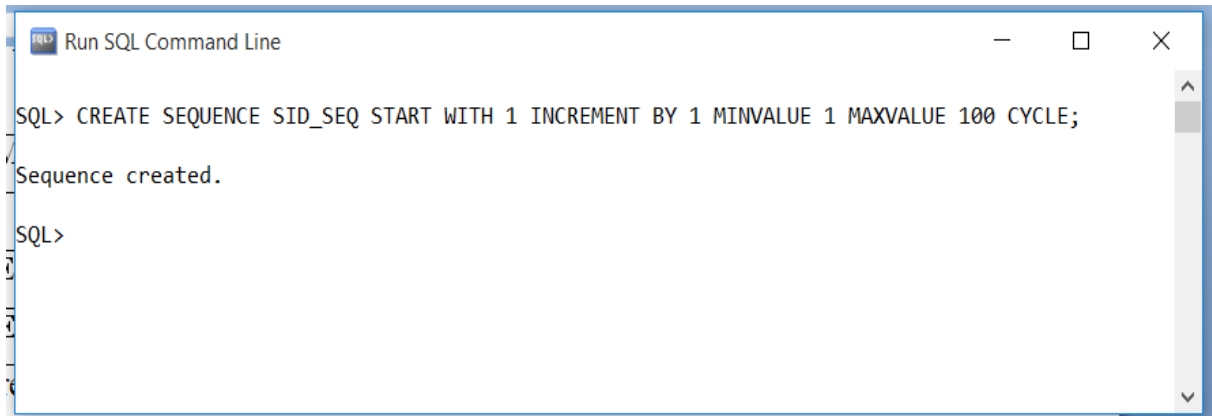
2) CREATE A TABLE CALLED 'INSTRUCTOR' WITH THE FOLLOWING STRUCTURE

COLUMN NAME	DATA TYPE
IID	NUMBER(5)
FNAME	VARCHAR2(12)
LNAME	VARCHAR2(12)



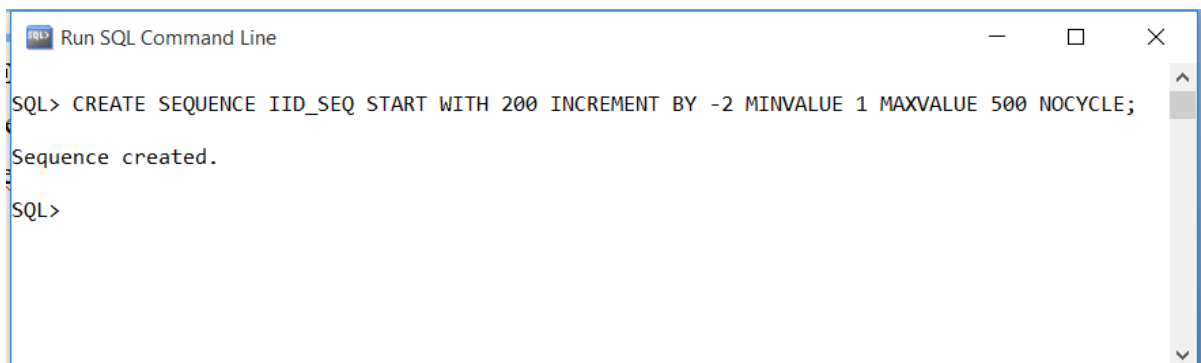
```
Run SQL Command Line
SQL> CREATE TABLE INSTRUCTOR(IID NUMBER(5), FNAME VARCHAR2(12),LNAME VARCHAR2(12));
Table created.
SQL>
```

3) CREATE A SEQUENCE CALLED 'SID_SEQ' WHICH WILL GENERATE NUMBERS FROM 1 TO 500 IN ASCENDING ORDER WITH AN INTERVAL OF '1' THE SEQUENCE MUST RESTART FROM THE NUMBER AFTER GENERATED NUMBER 500.



```
Run SQL Command Line
SQL> CREATE SEQUENCE SID_SEQ START WITH 1 INCREMENT BY 1 MINVALUE 1 MAXVALUE 100 CYCLE;
Sequence created.
SQL>
```

4) CREATE SEQUENCE CALLED 'IID_SEQ' WHICH SHOULD CONTAIN NUMBERS FROM 500 TO 1 WITH AN INTERVAL OF -2.



```
Run SQL Command Line
SQL> CREATE SEQUENCE IID_SEQ START WITH 200 INCREMENT BY -2 MINVALUE 1 MAXVALUE 500 NOCYCLE;
Sequence created.
SQL>
```

5) POPULATE DATA IN 'STUDENT' TABLES USING 'SID_SEQ'

```

Select Run SQL Command Line
SQL> DESC STUDENT;
Name                                     Null?      Type
-----
SID                                       NUMBER(5)
FNAME                                     VARCHAR2(12)
LNAME                                     VARCHAR2(12)

SQL> INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, '&FNAME', '&LNAME');
Enter value for fname: SUSAN
Enter value for lname: YAO
old 1: INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, '&FNAME', '&LNAME')
new 1: INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, 'SUSAN', 'YAO')

1 row created.

SQL> /
Enter value for fname: RAMESH
Enter value for lname: SHAH
old 1: INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, '&FNAME', '&LNAME')
new 1: INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, 'RAMESH', 'SHAH')

1 row created.

SQL> /
Enter value for fname: JOHNNY
Enter value for lname: KOHLER
old 1: INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, '&FNAME', '&LNAME')
new 1: INSERT INTO STUDENT VALUES(SID_SEQ.NEXTVAL, 'JOHNNY', 'KOHLER')

1 row created.
    
```

6) POPULATE DATA IN 'INSTRUCTOR' TABLES USING 'IID_SEQ'

```

Run SQL Command Line
SQL> DESC INSTRUCTOR
Name                                     Null?      Type
-----
IID                                       NUMBER(5)
FNAME                                     VARCHAR2(12)
LNAME                                     VARCHAR2(12)

SQL> INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, '&FNAME', '&LANE');
Enter value for fname: JOHN
Enter value for lane: SMITH
old 1: INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, '&FNAME', '&LANE')
new 1: INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, 'JOHN', 'SMITH')

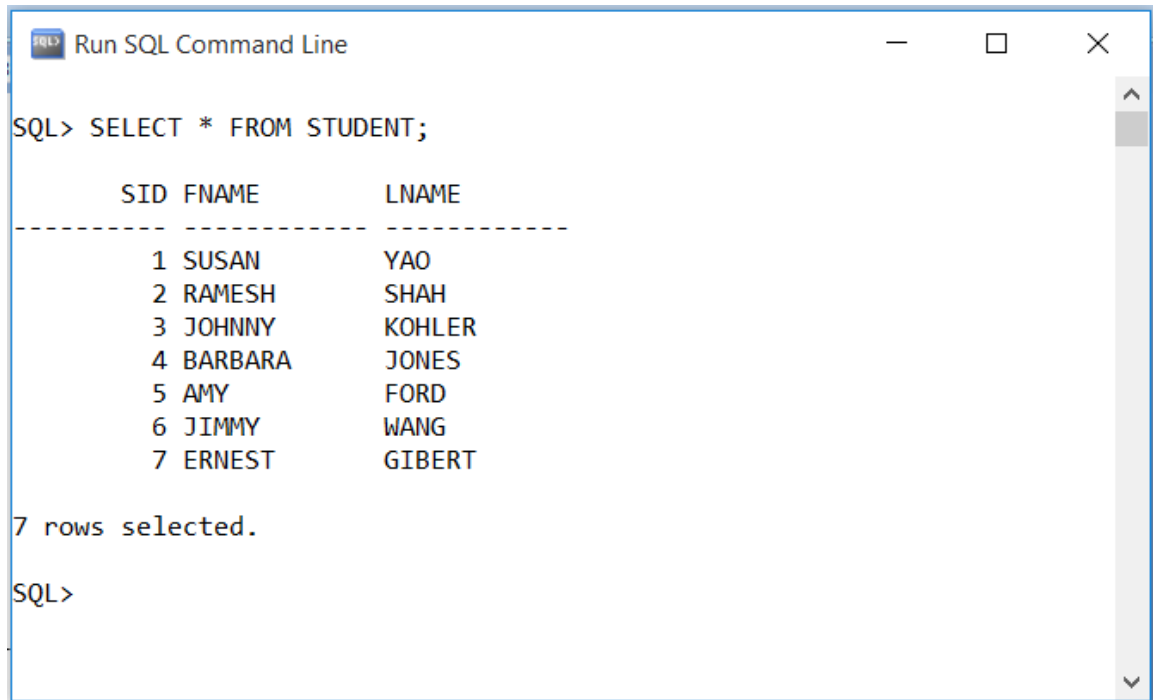
1 row created.

SQL> /
Enter value for fname: RICARDO
Enter value for lane: BROWNE
old 1: INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, '&FNAME', '&LANE')
new 1: INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, 'RICARDO', 'BROWNE')

1 row created.

SQL> /
Enter value for fname: SUSAN
Enter value for lane: YAO
old 1: INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, '&FNAME', '&LANE')
new 1: INSERT INTO INSTRUCTOR VALUES(IID_SEQ.NEXTVAL, 'SUSAN', 'YAO')

1 row created.
    
```

7) LIST THE RECORDS OF STUDENT TABLE

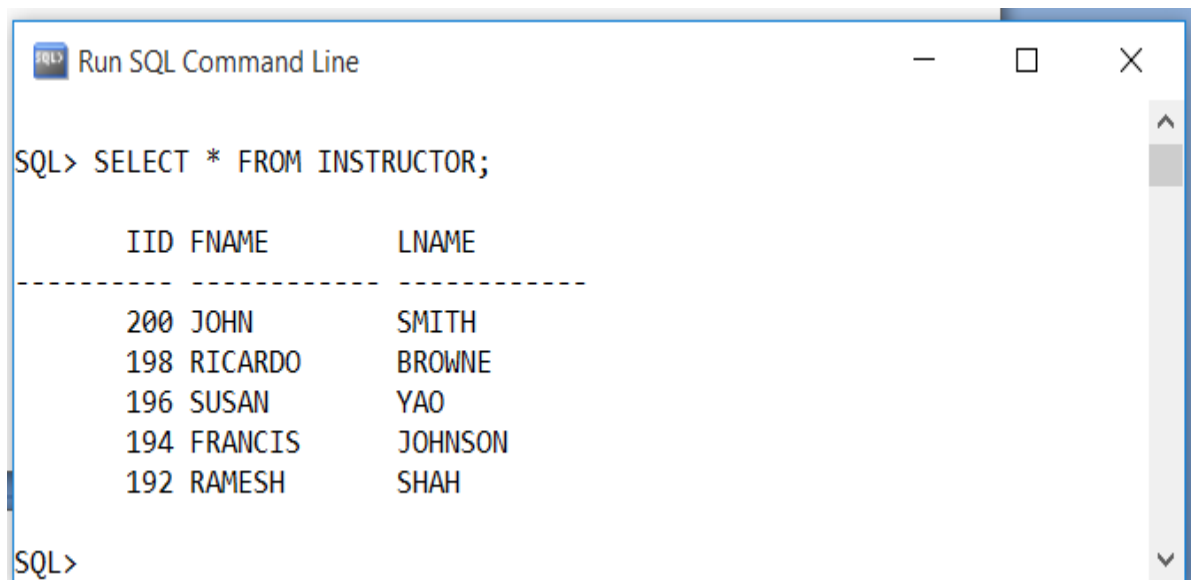
```
Run SQL Command Line

SQL> SELECT * FROM STUDENT;

  SID FNAME      LNAME
-----
   1 SUSAN       YAO
   2 RAMESH      SHAH
   3 JOHNNY      KOHLER
   4 BARBARA     JONES
   5 AMY         FORD
   6 JIMMY       WANG
   7 ERNEST     GIBERT

7 rows selected.

SQL>
```

8) LIST THE RECORDS OF INSTRUCTOR TABLE

```
Run SQL Command Line

SQL> SELECT * FROM INSTRUCTOR;

  IID FNAME      LNAME
-----
 200 JOHN        SMITH
 198 RICARDO    BROWNE
 196 SUSAN      YAO
 194 FRANCIS    JOHNSON
 192 RAMESH     SHAH

SQL>
```

9) LIST THE FIRSTNAMES OF ALL STUDENTS & INSTRUCTORS WITHOUT ELIMINATING DUPLICATE RECORDS.

```

Run SQL Command Line
-----
SUSAN
RAMESH
JOHNNY
BARBARA
AMY
JIMMY
ERNEST
7 rows selected.

SQL> SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
JOHN
RICARDO
SUSAN
FRANCIS
RAMESH

SQL> SELECT FNAME FROM STUDENT
  2 UNION ALL
  3 SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
SUSAN
RAMESH
JOHNNY
BARBARA
AMY
JIMMY
ERNEST
JOHN
RICARDO
SUSAN
FRANCIS
RAMESH
12 rows selected.
    
```

10) LIST THE FIRSTNAMES OF ALL STUDENTS & INSTRUCTORS BY ELIMINATING DUPLICATE RECORDS.

```

Run SQL Command Line
-----
SQL> SELECT FNAME FROM STUDENT;
FNAME
-----
SUSAN
RAMESH
JOHNNY
BARBARA
AMY
JIMMY
ERNEST
7 rows selected.

SQL> SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
JOHN
RICARDO
SUSAN
FRANCIS
RAMESH

SQL> SELECT FNAME FROM STUDENT
  2 UNION
  3 SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
AMY
BARBARA
ERNEST
FRANCIS
JIMMY
JOHN
JOHNNY
RAMESH
ERICARDO
SUSAN
11 rows selected.
    
```

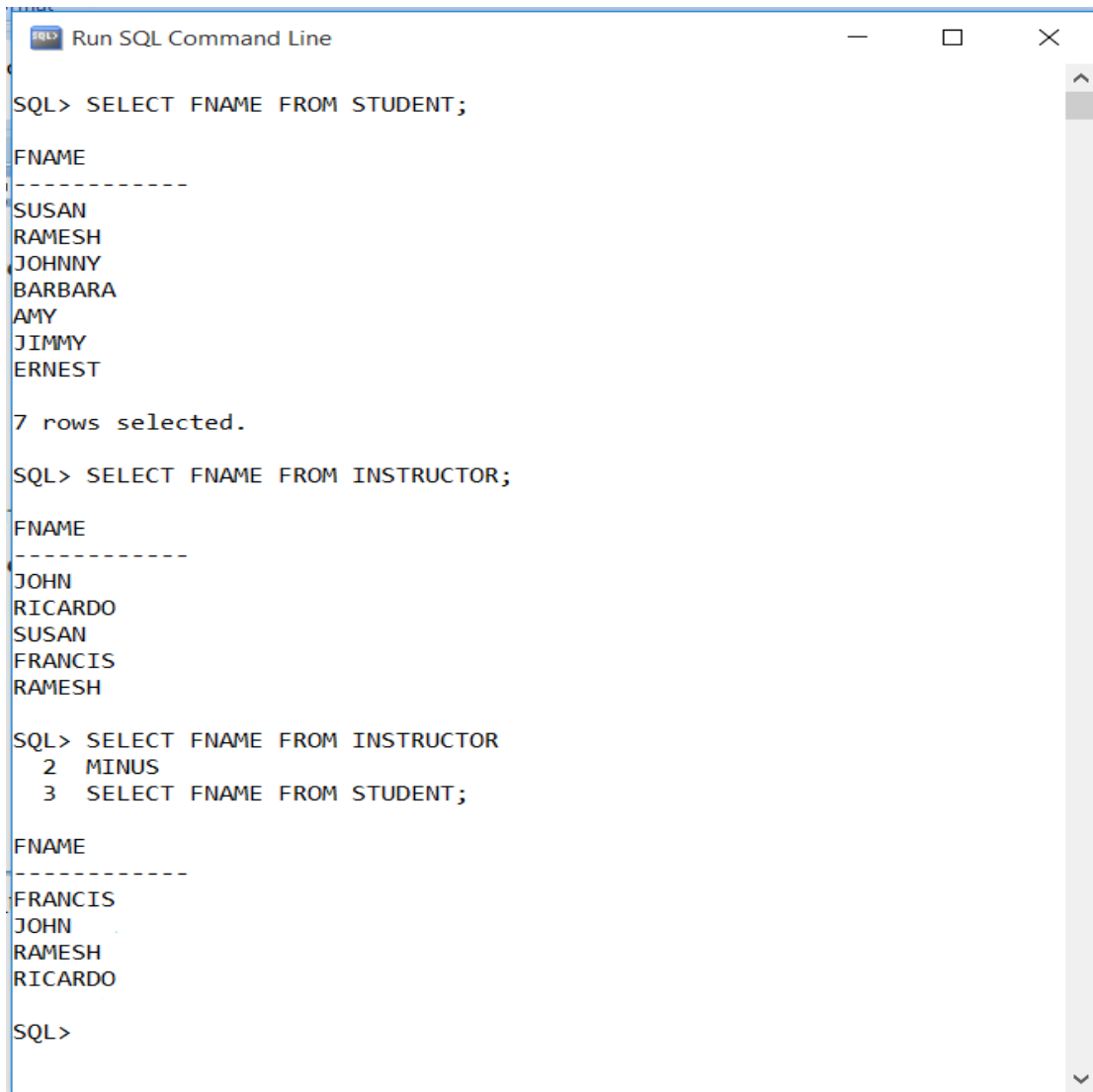
11) LIST ALL THE FIRST NAMES OF STUDENTS WHO IS ALSO AN INSTRUCTOR

```
Run SQL Command Line
SQL> SELECT FNAME FROM STUDENT;
FNAME
-----
SUSAN
RAMESH
JOHNNY
BARBARA
AMY
JIMMY
ERNEST
7 rows selected.
SQL> SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
JOHN
RICARDO
SUSAN
FRANCIS
RAMESH
SQL> SELECT FNAME FROM STUDENT
 2 INTERSECT
 3 SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
SUSAN
SQL>
```

12) LIST THE FIRST NAMES OF ALL STUDENTS WHO ARE NOT INSTRUCTORS

```
Run SQL Command Line
SQL> SELECT FNAME FROM STUDENT;
FNAME
-----
SUSAN
RAMESH
JOHNNY
BARBARA
AMY
JIMMY
ERNEST
7 rows selected.
SQL> SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
JOHN
RICARDO
SUSAN
FRANCIS
RAMESH
SQL> SELECT FNAME FROM STUDENT
 2 MINUS
 3 SELECT FNAME FROM INSTRUCTOR;
FNAME
-----
AMY
BARBARA
ERNEST
JIMMY
JOHNNY
RAMESH
6 rows selected.
```

13) LIST THE FIRST NAMES OF ALL INSTRUCTORS WHO ARE NOT STUDENTS



```
Run SQL Command Line
SQL> SELECT FNAME FROM STUDENT;

FNAME
-----
SUSAN
RAMESH
JOHNNY
BARBARA
AMY
JIMMY
ERNEST

7 rows selected.

SQL> SELECT FNAME FROM INSTRUCTOR;

FNAME
-----
JOHN
RICARDO
SUSAN
FRANCIS
RAMESH

SQL> SELECT FNAME FROM INSTRUCTOR
 2  MINUS
 3  SELECT FNAME FROM STUDENT;

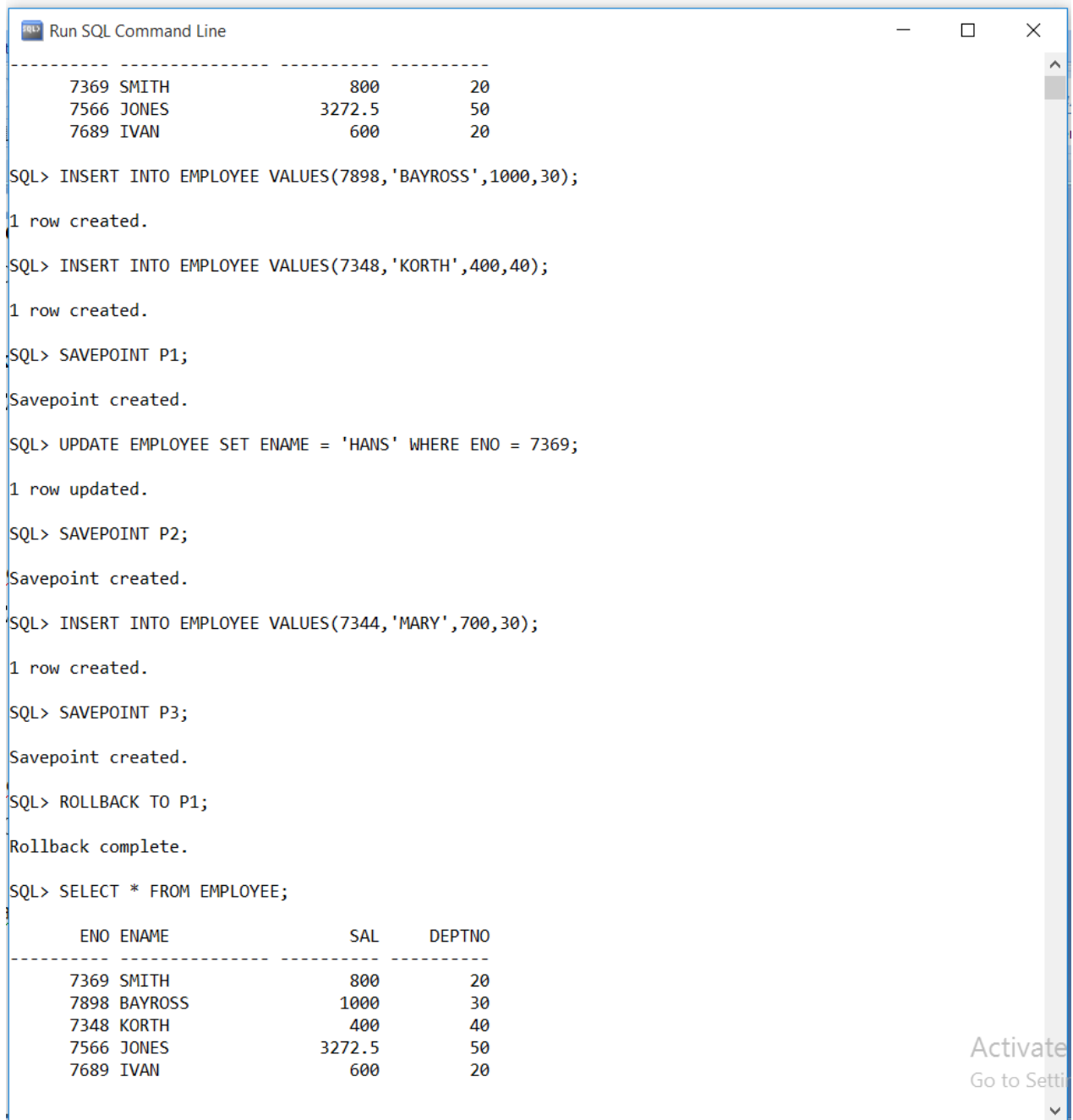
FNAME
-----
FRANCIS
JOHN
RAMESH
RICARDO

SQL>
```

EXERCISE NO:9

AIM : TO ILLUSTRATE TCL COMMANDS

1) TO ILLUSTRATE SAVEPOINT



```
Run SQL Command Line
-----
7369 SMITH          800      20
7566 JONES         3272.5   50
7689 IVAN          600      20

SQL> INSERT INTO EMPLOYEE VALUES(7898,'BAYROSS',1000,30);
1 row created.

SQL> INSERT INTO EMPLOYEE VALUES(7348,'KORTH',400,40);
1 row created.

SQL> SAVEPOINT P1;
Savepoint created.

SQL> UPDATE EMPLOYEE SET ENAME = 'HANS' WHERE ENO = 7369;
1 row updated.

SQL> SAVEPOINT P2;
Savepoint created.

SQL> INSERT INTO EMPLOYEE VALUES(7344,'MARY',700,30);
1 row created.

SQL> SAVEPOINT P3;
Savepoint created.

SQL> ROLLBACK TO P1;
Rollback complete.

SQL> SELECT * FROM EMPLOYEE;
-----
      ENO ENAME          SAL      DEPTNO
-----
7369 SMITH          800      20
7898 BAYROSS       1000     30
7348 KORTH          400      40
7566 JONES         3272.5   50
7689 IVAN          600      20
```

Activate
Go to Settings

2) TO ILLUSTRATE COMMIT COMMAND

```
Run SQL Command Line
SQL> SELECT * FROM EMPLOYEE;

   ENO ENAME          SAL      DEPTNO
-----
   7369 HANS           800        20
   7898 BAYROSS       1000        30
   7348 KORTH         400        40
   7566 JONES        3272.5      50
   7656 JOHN          870        10
   7689 IVAN          600        20

6 rows selected.

SQL> INSERT INTO EMPLOYEE VALUES(7344, 'ROSY', 700, 30);

1 row created.

SQL> INSERT INTO EMPLOYEE VALUES(7354, 'SMITH', 700, 30);

1 row created.

SQL> SAVEPOINT PP1;

Savepoint created.

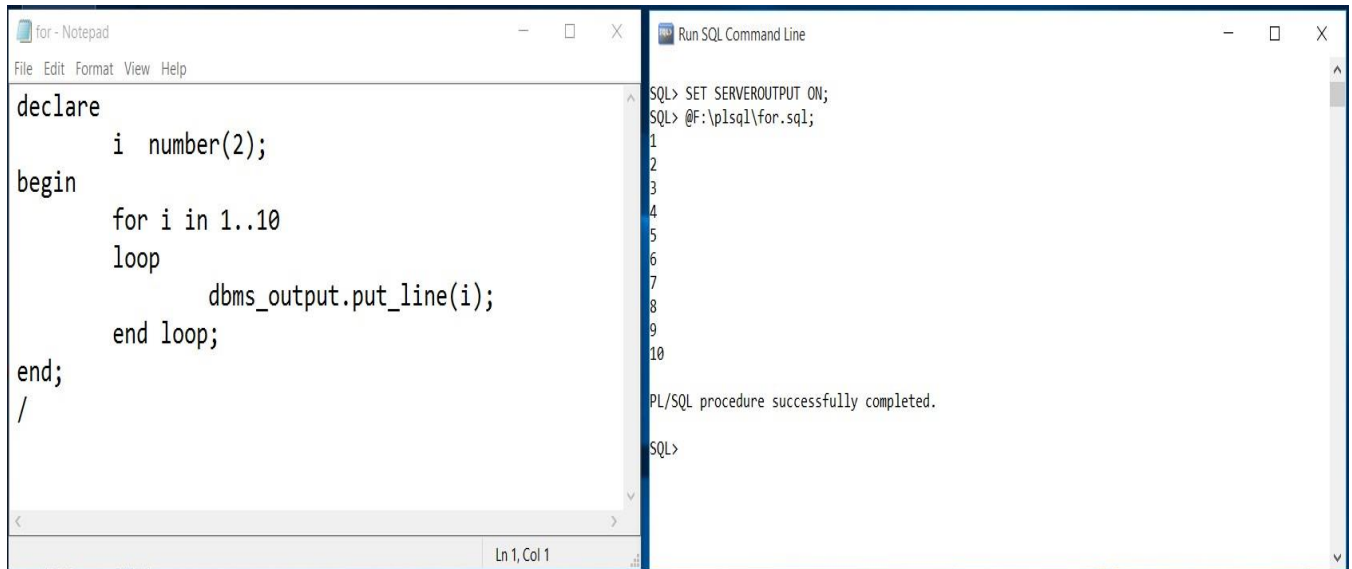
SQL> COMMIT;

Commit complete.

SQL> SELECT * FROM EMPLOYEE;

   ENO ENAME          SAL      DEPTNO
-----
   7369 HANS           800        20
   7898 BAYROSS       1000        30
   7348 KORTH         400        40
   7566 JONES        3272.5      50
   7656 JOHN          870        10
   7344 ROSY          700        30
   7689 IVAN          600        20
   7354 SMITH         700        30

8 rows selected.
```

EXERCISE NO.10**AIM : TO IMPLEMENT PL/SQL CONTROL STATEMENT****1) TO IMPLEMENT 'FOR' STATEMENT**

The screenshot shows two windows side-by-side. The left window is a Notepad editor titled 'for - Notepad' containing the following PL/SQL code:

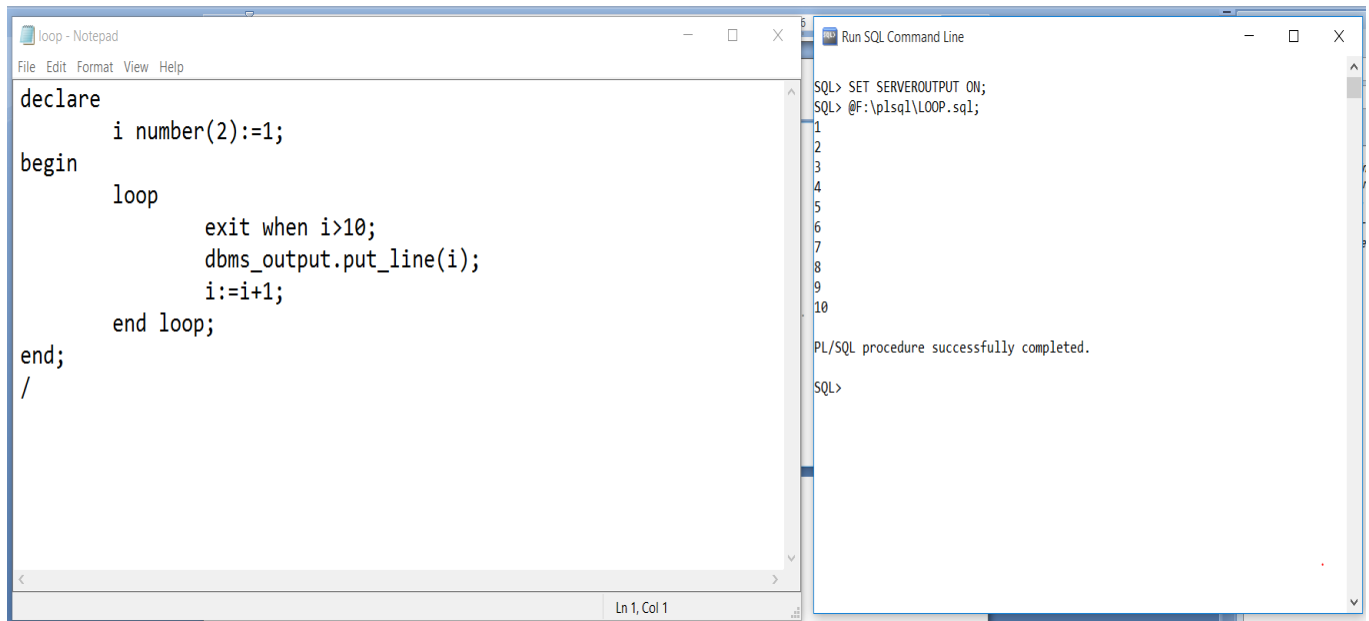
```
declare
    i number(2);
begin
    for i in 1..10
    loop
        dbms_output.put_line(i);
    end loop;
end;
/
```

The right window is 'Run SQL Command Line' showing the execution of the code:

```
SQL> SET SERVEROUTPUT ON;
SQL> @F:\plsql\for.sql;
1
2
3
4
5
6
7
8
9
10

PL/SQL procedure successfully completed.

SQL>
```

2) TO IMPLEMENT 'WHILE' STATEMENT

The screenshot shows two windows side-by-side. The left window is a Notepad editor titled 'loop - Notepad' containing the following PL/SQL code:

```
declare
    i number(2):=1;
begin
    loop
        exit when i>10;
        dbms_output.put_line(i);
        i:=i+1;
    end loop;
end;
/
```

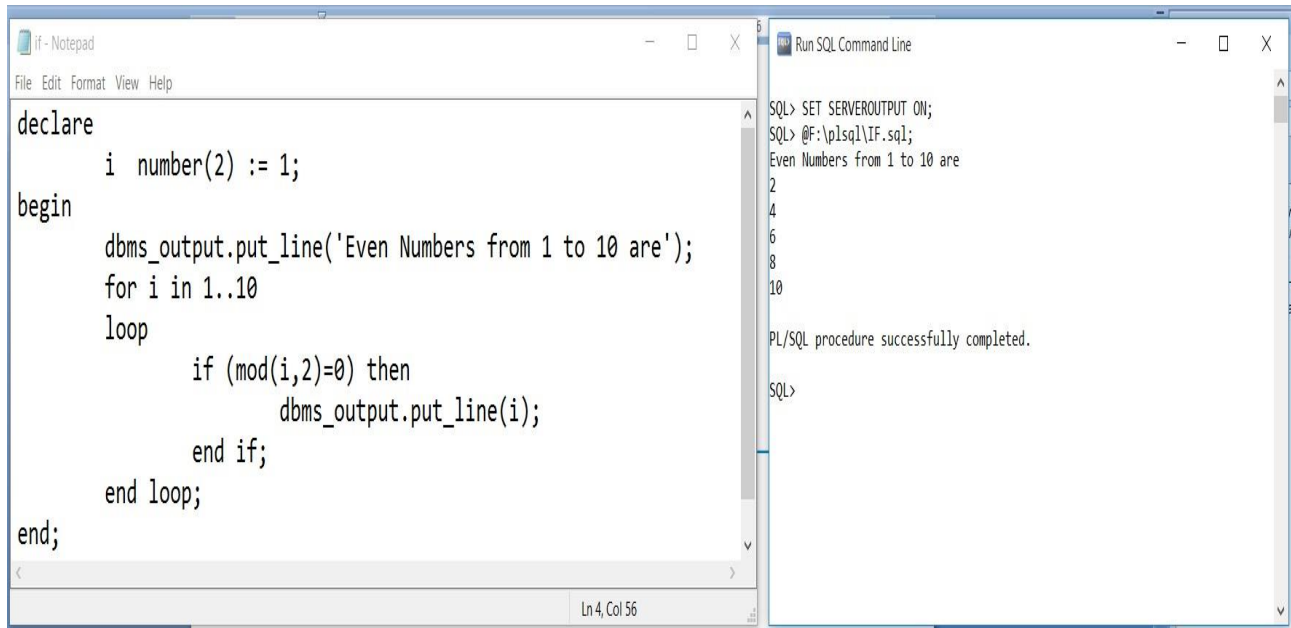
The right window is 'Run SQL Command Line' showing the execution of the code:

```
SQL> SET SERVEROUTPUT ON;
SQL> @F:\plsql\LOOP.sql;
1
2
3
4
5
6
7
8
9
10

PL/SQL procedure successfully completed.

SQL>
```

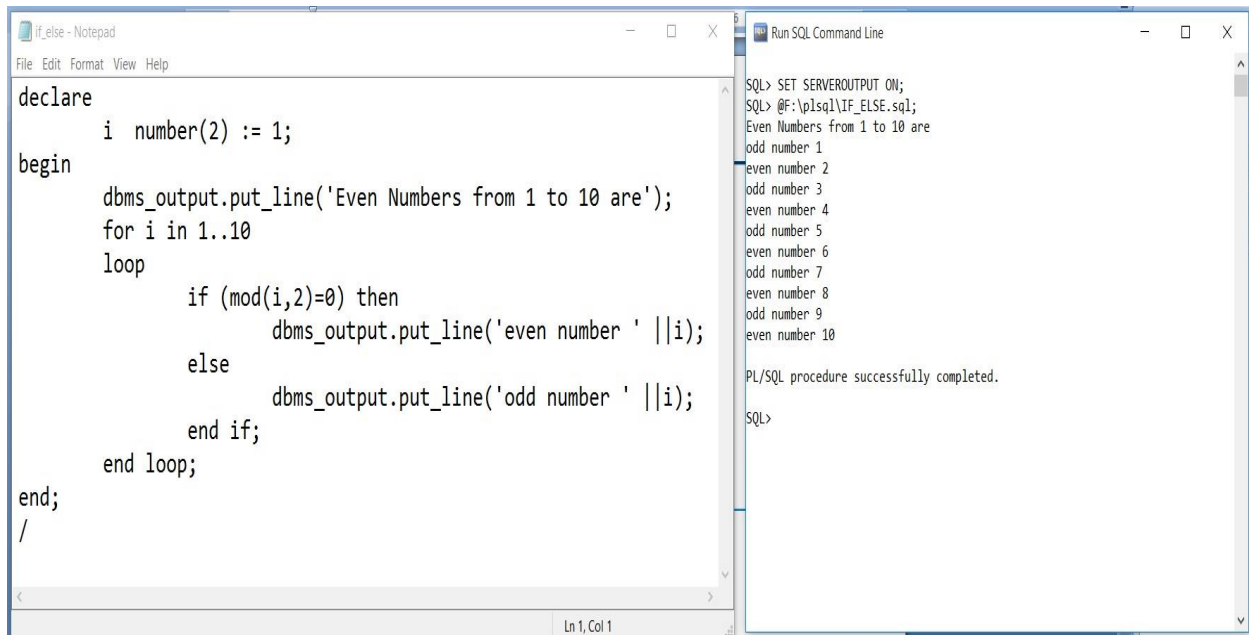
3) TO IMPLEMENT 'IF' STATEMENT



```
if - Notepad
File Edit Format View Help
declare
  i number(2) := 1;
begin
  dbms_output.put_line('Even Numbers from 1 to 10 are');
  for i in 1..10
  loop
    if (mod(i,2)=0) then
      dbms_output.put_line(i);
    end if;
  end loop;
end;
```

```
Run SQL Command Line
SQL> SET SERVEROUTPUT ON;
SQL> @F:\plsql\IF.sql;
Even Numbers from 1 to 10 are
2
4
6
8
10
PL/SQL procedure successfully completed.
SQL>
```

4) TO IMPLEMENT 'IF.ELSE' STATEMENT



```
if_else - Notepad
File Edit Format View Help
declare
  i number(2) := 1;
begin
  dbms_output.put_line('Even Numbers from 1 to 10 are');
  for i in 1..10
  loop
    if (mod(i,2)=0) then
      dbms_output.put_line('even number ' ||i);
    else
      dbms_output.put_line('odd number ' ||i);
    end if;
  end loop;
end;
/
```

```
Run SQL Command Line
SQL> SET SERVEROUTPUT ON;
SQL> @F:\plsql\IF_ELSE.sql;
Even Numbers from 1 to 10 are
odd number 1
even number 2
odd number 3
even number 4
odd number 5
even number 6
odd number 7
even number 8
odd number 9
even number 10
PL/SQL procedure successfully completed.
SQL>
```

5) TO IMPLEMENT 'IF..ELSIF' STATEMENT

The screenshot shows two windows. The left window is a Notepad editor titled 'if_elseif - Notepad' containing the following PL/SQL code:

```

declare
    a number(2) := &a;
    b number(2) := &b;
    c number(2) := &c;
begin
    if(a>b)and(a>c) then
        dbms_output.put_line(a||'is biggest');
    elsif(b>c)then
        dbms_output.put_line(b||'is biggest');
    else
        dbms_output.put_line(c||'is biggest');
    end if;
end;
/
    
```

The right window is titled 'Run SQL Command Line' and shows the execution of the code:

```

SQL> SET SERVEROUTPUT ON;
SQL> @F:\plsql\IF_ELSEIF.sql;
Enter value for a: 5
old 2:      a number(2) := &a;
new 2:      a number(2) := 5;
Enter value for b: 2
old 3:      b number(2) := &b;
new 3:      b number(2) := 2;
Enter value for c: 3
old 4:      c number(2) := &c;
new 4:      c number(2) := 3;
5is biggest

PL/SQL procedure successfully completed.

SQL>
    
```

6) TO IMPLEMENT 'CASE' STATEMENT

The screenshot shows two windows. The left window is a Notepad editor titled 'case - Notepad' containing the following PL/SQL code:

```

declare
    cgpa number(2) := &cgpa;
begin
    case cgpa
        when 6 then dbms_output.put_line('grade is D');
        when 7 then dbms_output.put_line('grade is C');
        when 8 then dbms_output.put_line('grade is B');
        when 9 then dbms_output.put_line('grade is A');
        when 10 then dbms_output.put_line('grade is S');
        else dbms_output.put_line('failed');
    end case;
end;
/
    
```

The right window is titled 'Run SQL Command Line' and shows the execution of the code:

```

SQL> SET SERVEROUTPUT ON;
SQL> @F:\plsql\CASE.sql;
Enter value for cgpa: 9
old 2:      cgpa number(2) := &cgpa;
new 2:      cgpa number(2) := 9;
grade is A

PL/SQL procedure successfully completed.

SQL>
    
```

