



**VIT<sup>®</sup>**  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

**Course Code: BCSE302P**

**Course Name: Database Systems Lab**

**Assessment – 1**

**Name: Shivam Dave**

**Reg. No.: 21BCB0107**

## DDL and DML COMMANDS

**Aim: To study Data Definition and Data Manipulation commands.**

### Exercise-I:

Create the student schema as follows

STUDENT (Regno, Fname, Lname, Phone, Address)

OUTPUT:

```
Table created.
SQL> desc STUDENT
Name                               Null?    Type
-----
REGNO                             NOT NULL VARCHAR2(20)
FNAME                             VARCHAR2(10)
LNAME                             VARCHAR2(10)
PHONE                             NUMBER(38)
ADDRESS                           VARCHAR2(20)
```

1. Add a column Email\_id

Query:

```
SQL> alter table STUDENT add(Email_ID varchar(20));
Table altered.
```

Output:

```
SQL> desc STUDENT
Name                               Null?    Type
-----
REGNO                             NOT NULL VARCHAR2(20)
FNAME                             VARCHAR2(10)
LNAME                             VARCHAR2(10)
PHONE                             NUMBER(38)
ADDRESS                           VARCHAR2(20)
EMAIL_ID                           VARCHAR2(20)
```

2. Remove the column Address

Query:

```
SQL> alter table STUDENT drop(ADDRESS);
Table altered.
```

Output:

```
SQL> desc STUDENT
Name                               Null?    Type
-----
REGNO                             NOT NULL VARCHAR2(20)
FNAME                             VARCHAR2(10)
LNAME                             VARCHAR2(10)
PHONE                             NUMBER(38)
EMAIL_ID                           VARCHAR2(20)
SQL> |
```

3. Modify the column Phone with size 10.

Query:

```
SQL> alter table STUDENT modify(phone number(10));
Table altered.
```

Output:

```
SQL> desc STUDENT
Name                               Null?    Type
-----
REGNO                             NOT NULL VARCHAR2(20)
FNAME                             VARCHAR2(10)
LNAME                             VARCHAR2(10)
PHONE                             NUMBER(10)
EMAIL_ID                           VARCHAR2(20)
SQL> |
```

4. Rename a column Phone by Mobile\_no.

Query:

```
SQL> alter table STUDENT Rename column PHONE to MOBILE_NO;
Table altered.
```

Output:

```
SQL> desc STUDENT
Name                               Null?    Type
-----
REGNO                             NOT NULL VARCHAR2(20)
FNAME                             VARCHAR2(10)
LNAME                             VARCHAR2(10)
MOBILE_NO                          NUMBER(10)
EMAIL_ID                           VARCHAR2(20)
SQL>
```

5. Rename the table name STUDENT by STUDENTS

Query:

```
SQL> rename STUDENT TO STUDENTS;
Table renamed.
```

Output:

21BCB0107

```
SQL> desc STUDENTS
Name                                         Null?   Type
-----
REGNO                                         NOT NULL VARCHAR2(20)
FNAME                                         VARCHAR2(10)
LNAME                                         VARCHAR2(10)
MOBILE_NO                                     NUMBER(10)
EMAIL_ID                                     VARCHAR2(20)
```

## 6. Insert 10 students records in STUDENTS table

Query:

```
SQL> insert into STUDENTS
2 values('21BCE0001','Shivan','Dave',9551212,'shivan@gmail.com');
1 row created.
```

OUTPUT:

```
REGNO      FNAME      LNAME      MOBILE_NO EMAIL_ID
-----
21BCE0001   Shivan     Dave       9551212   shivan@gmail.com
21BCE0002   Samir      Dave       9912121   samir@gmail.com
21BCE0018   Virat      Kohli      19191918  vkohli@gmail.com
21BCE003     Ms         Dhoni      120207    msd@gmail.com
21BCB0107   Lionel    Messi      100101010 mleo@gmail.com
21BCB0000   Cris       Ronaldo    7070707   cris@gmail.com
21BCD0104   Xavi       Hernandez  10203910  xavi@gmail.com
21BDS0010   Neymar     Jr         1010111   njr@gmail.com
21BCT0001   Gavi       Hernandez  3030303   gavi@gmail.com
19BCE0012   Kan        Ye         123401    kanye@gmail.com

10 rows selected.
```

SQL>

## 7. Show the table description

```
SQL> desc STUDENTS
Name                                         Null?   Type
-----
REGNO                                         NOT NULL VARCHAR2(20)
FNAME                                         VARCHAR2(10)
LNAME                                         VARCHAR2(10)
MOBILE_NO                                     NUMBER(10)
EMAIL_ID                                     VARCHAR2(20)

SQL>
```

## 8. Display all the records

```
SQL> Select * from STUDENTS
2 ;

REGNO      FNAME      LNAME      MOBILE_NO EMAIL_ID
-----
21BCE0001   Shivan     Dave       9551212   shivan@gmail.com
21BCE0002   Samir      Dave       9912121   samir@gmail.com
21BCE0018   Virat      Kohli      19191918  vkohli@gmail.com
21BCE003     Ms         Dhoni      120207    msd@gmail.com
21BCB0107   Lionel    Messi      100101010 mleo@gmail.com
21BCB0000   Cris       Ronaldo    7070707   cris@gmail.com
21BCD0104   Xavi       Hernandez  10203910  xavi@gmail.com
21BDS0010   Neymar     Jr         1010111   njr@gmail.com
21BCT0001   Gavi       Hernandez  3030303   gavi@gmail.com
19BCE0012   Kan        Ye         123401    kanye@gmail.com

10 rows selected.
```

SQL>

## 9. Delete all the records

Query:

21BCB0107

```
SQL> delete from STUDENTS  
2 ;  
10 rows deleted.  
SQL> |
```

## 10.Drop the table

```
SQL> drop table STUDENTS;  
Table dropped.  
SQL> |
```

## Exercise 2:

1. Insert the data given above in both employee, department and project tables.

### Queries:

```

Oracle SQL*Plus
File Edit Search Options Help

SQL> alter table employee modify(salary number(7));
Table altered.

SQL> insert into Employee
  2  values('Doug','E','Gilbert','123','09-jun-1968','Chennai','M',80000,NULL,1);
1 row created.

SQL> insert into Employee
  2  values('Joyce','','PAN','124','07-FEB-1973','Vellore','F',70000,NULL,1);
1 row created.

SQL> insert into EMPLOYEE
  2  values('Franklin','T','Wong','125','08-DEU-1972','Delhi','M',40000,123,2);
values('Franklin','T','Wong','125','08-DEU-1972','Delhi','M',40000,123,2)
*
ERROR at line 2:
ORA-01843: not a valid month

SQL> insert into EMPLOYEE
  2  values('Franklin','T','Wong','125','08-DEC-1972','Delhi','M',40000,123,2);
1 row created.

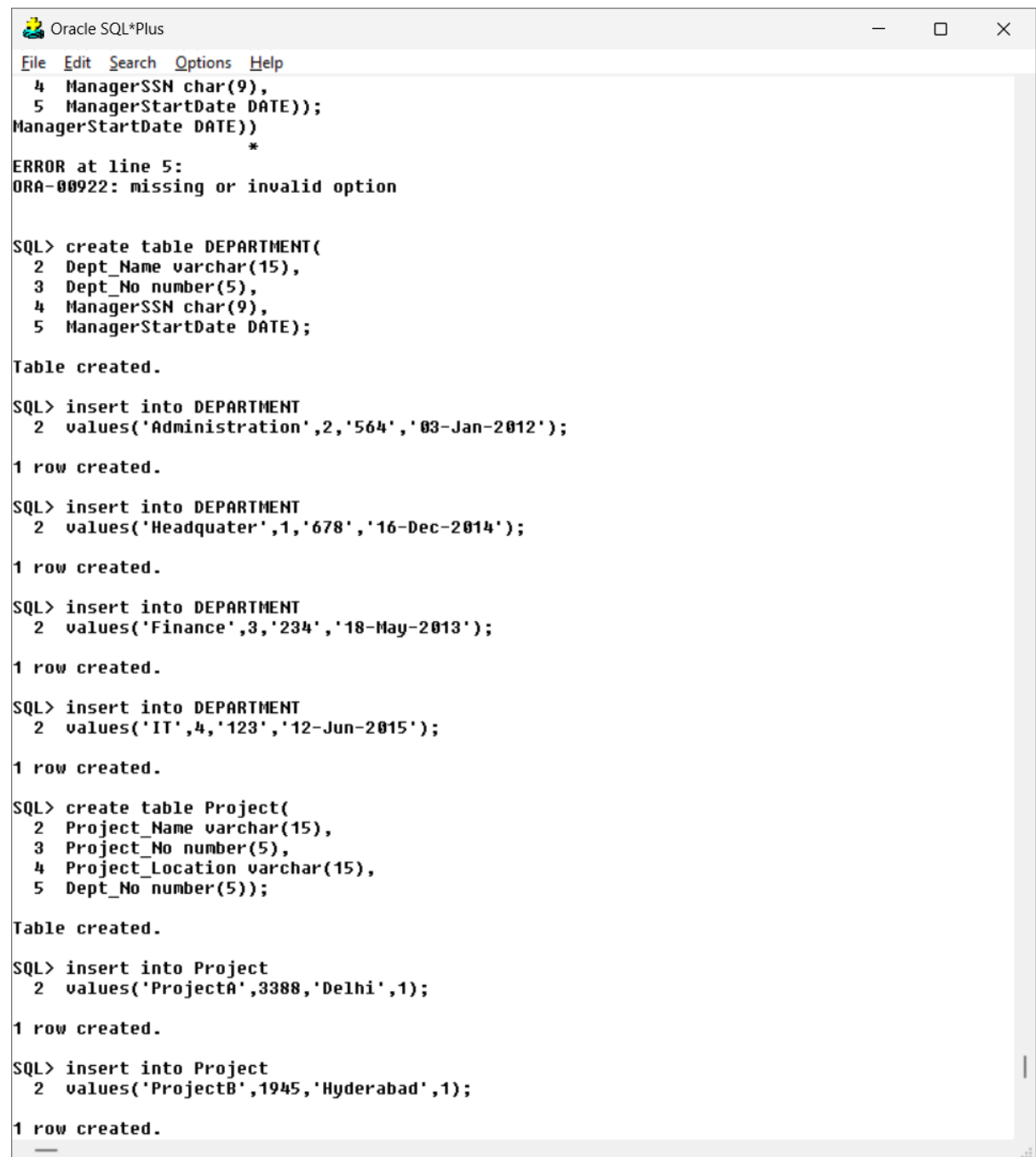
SQL> insert into EMPLOYEE
  2  values('Jennifer','S','Wallace','564','20-JUN-1983','Chennai','F',43000,123,2);
1 row created.

SQL> insert into EMPLOYEE
  2  values('John','B','Smith','678','09-JAN-1987','Madurai','M',30000,124,1);
1 row created.

SQL> INSERT INTO EMPLOYEE
  2  values('Ramesh','K','Narayan','234','15-SEP-1985','Bangalore','M',38000,124,3);
1 row created.

SQL> create table DEPARTMENT(
  2  Dept_Name varchar(15),
  3  Dept_No number(5),
  4  ManagerSSN char(9),
  5  ManagerStartDate DATE));
ManagerStartDate DATE))
*
ERROR at line 5:
ORA-00922: missing or invalid option

```



```
Oracle SQL*Plus
File Edit Search Options Help
  4 ManagerSSN char(9),
  5 ManagerStartDate DATE));
ManagerStartDate DATE))
*
ERROR at line 5:
ORA-00922: missing or invalid option

SQL> create table DEPARTMENT(
  2 Dept_Name varchar(15),
  3 Dept_No number(5),
  4 ManagerSSN char(9),
  5 ManagerStartDate DATE);

Table created.

SQL> insert into DEPARTMENT
  2 values('Administration',2,'564','03-Jan-2012');

1 row created.

SQL> insert into DEPARTMENT
  2 values('Headquater',1,'678','16-Dec-2014');

1 row created.

SQL> insert into DEPARTMENT
  2 values('Finance',3,'234','18-May-2013');

1 row created.

SQL> insert into DEPARTMENT
  2 values('IT',4,'123','12-Jun-2015');

1 row created.

SQL> create table Project(
  2 Project_Name varchar(15),
  3 Project_No number(5),
  4 Project_Location varchar(15),
  5 Dept_No number(5));

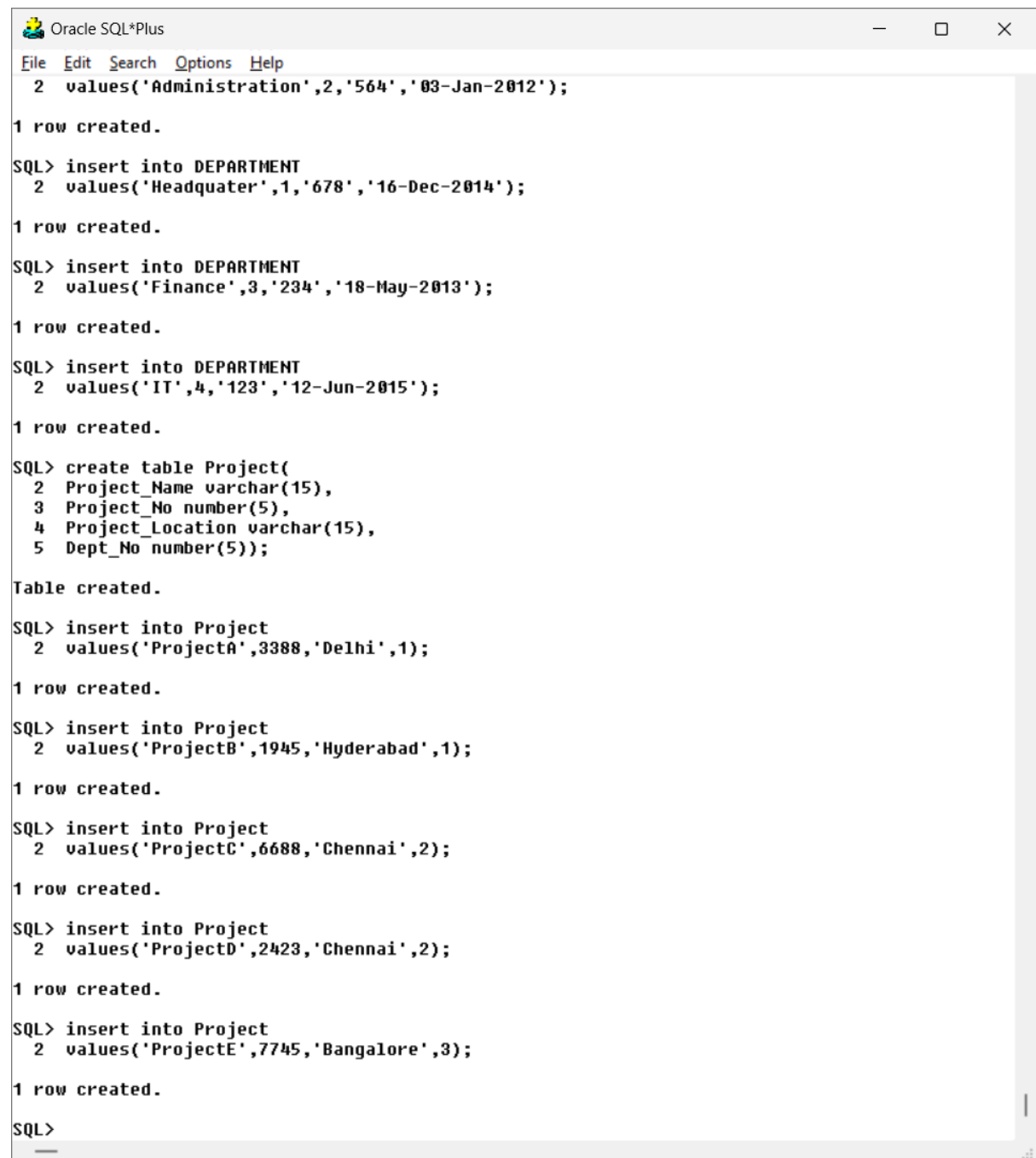
Table created.

SQL> insert into Project
  2 values('ProjectA',3388,'Delhi',1);

1 row created.

SQL> insert into Project
  2 values('ProjectB',1945,'Hyderabad',1);

1 row created.
```



```

Oracle SQL*Plus
File Edit Search Options Help
2 values('Administration',2,'564','03-Jan-2012');
1 row created.
SQL> insert into DEPARTMENT
2 values('Headquater',1,'678','16-Dec-2014');
1 row created.
SQL> insert into DEPARTMENT
2 values('Finance',3,'234','18-May-2013');
1 row created.
SQL> insert into DEPARTMENT
2 values('IT',4,'123','12-Jun-2015');
1 row created.
SQL> create table Project(
2 Project_Name varchar(15),
3 Project_No number(5),
4 Project_Location varchar(15),
5 Dept_No number(5));
Table created.
SQL> insert into Project
2 values('ProjectA',3388,'Delhi',1);
1 row created.
SQL> insert into Project
2 values('ProjectB',1945,'Hyderabad',1);
1 row created.
SQL> insert into Project
2 values('ProjectC',6688,'Chennai',2);
1 row created.
SQL> insert into Project
2 values('ProjectD',2423,'Chennai',2);
1 row created.
SQL> insert into Project
2 values('ProjectE',7745,'Bangalore',3);
1 row created.
SQL>

```

2. Display all the employees' information.

Output:

FIRST_NAME	HI LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS	S	SALARY	SUPERVISO	DEPT_NO
Doug	E Gilbert	123	09-JUN-68	Chennai	M	80000		1
Joyce	PAN	124	07-FEB-73	Vellore	F	70000		1
Franklin	T Wong	125	08-DEC-72	Delhi	H	40000	123	2
Jennifer	S Wallace	504	20-JUN-80	Chennai	F	40000	123	2
John	B Smith	678	09-JAN-87	Madurai	H	30000	124	1
Ramesh	K Narayan	234	15-SEP-85	Bangalore	H	38000	124	3



### 3. Display Employee name along with his SSN and Supervisor SSN.

Query:

```
SQL> select First_Name, SSN_Number, Supervisor_SSN from Employee;
```

Output:

FIRST_NAME	SSN_NUMBE	SUPERVISO
Doug	123	
Joyce	124	
Franklin	125	123
Jennifer	564	123
John	678	124
Ramesh	234	124

6 rows selected.

### 4. Display the employee names whose bdate is '20-JUN-1983'.

Query:

```
SQL> select First_name from employee where birthday='20-JUNE-1983';
```

Output:

FIRST_NAME
Jennifer

SQL> |

### 5. Display salary of the employees without duplications.

## Query:

```
SQL> select DISTINCT SALARY, First_name from EMPLOYEE;
```

## Output:

```

SALARY FIRST_NAME
-----
80000 Doug
70000 Joyce
30000 John
43000 Jennifer
40000 Franklin
38000 Ramesh

6 rows selected.

SQL>
```

6. Display the MgrSSN, MgrStartDate of the manager of 'Finance' department.

## Query:

```
SQL> select MANAGERSSN,MANAGERSTARTDATE from DEPARTMENT where dept_name='Finance';
```

## Output:

```

MANAGERSS MANAGERST
-----
234      18-MAY-13

SQL>
```

7. Modify the department number of an employee having fname as 'Joyce' to 5

## Query:

```
SQL> update employee set dept_no=5 where first_name='Joyce';

1 row updated.
```

Output:

```

FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY  ADDRESS
-----
Doug            E  Gilbert        123       09-JUN-68 Chennai
Joyce           PAN          124       07-FEB-73 Vellore
Franklin        T  Wong           125       08-DEC-72 Delhi
Jennifer        S  Wallace        564       20-JUN-83 Chennai
John            B  Smith          678       09-JAN-87 Madurai
Ramesh          K  Narayan        234       15-SEP-85 Bangalore

6 rows selected.

SQL>

```

8. Alter Table department add column  
DepartmentPhoneNum of NUMBER data

Query:

type and insert values into this column only.

```

SQL> ALTER TABLE DEPARTMENT ADD CONTACT_NUMBER NUMBER(10);
Table altered.

SQL> UPDATE DEPARTMENT SET CONTACT_NUMBER=1234567890 WHERE DEPT_NO=1;
1 row updated.

SQL> UPDATE DEPARTMENT SET CONTACT_NUMBER=1234567890 WHERE DEPT_NO=2;
1 row updated.

SQL> UPDATE DEPARTMENT SET CONTACT_NUMBER=1234567890 WHERE DEPT_NO=3;
1 row updated.

SQL> UPDATE DEPARTMENT SET CONTACT_NUMBER=1234567890 WHERE DEPT_NO=4;
1 row updated.

SQL> UPDATE DEPARTMENT SET CONTACT_NUMBER=1234567890 WHERE DEPT_NO=5;|

```

9. Alter table department to modify the size of  
DepartmentPhoneNum.

Query:

```
SQL> ALTER TABLE DEPARTMENT MODIFY CONTACT_NUMBER NUMBER(11);
Table altered.
SQL>
```

OUTPUT:

```
SQL> select * from department;
```

DEPT_NAME	DEPT_NO	MANAGERSS	MANAGERST	CONTACT_NUMBER
Administration	2	564	03-JAN-12	1234567890
Headquater	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

```
SQL> |
```

10. Modify the field name DepartmentPhoneNum of departments table to PhNo.

Query:

```
SQL> ALTER TABLE DEPARTMENT RENAME COLUMN CONTACT_NUMBER TO PhNo;
Table altered.
```

Output:

```
SQL> select*from department;
```

DEPT_NAME	DEPT_NO	MANAGERSS	MANAGERST	PHNO
Administration	2	564	03-JAN-12	1234567890
Headquater	1	678	16-DEC-14	1234567890
Finance	3	234	18-MAY-13	1234567890
IT	4	123	12-JUN-15	1234567890

```
SQL> |
```

11. Rename Table Department as DEPT.

Query:

```
SP2-0734: unknown command beginning "ORA-00955:..." - rest of line ignored.  
SQL> rename department to deptt;  
  
Table renamed.
```

12. Alter Table department remove column PhNo.

Query:

```
SQL> ALTER TABLE DEPTT DROP COLUMN Phno;  
  
Table altered.  
  
SQL>
```

13. Create a table COPYOFDEPT as a copy of the table DEPT.

Query:

```
SQL> create table copyof_Dept as  
2  Select *  
3  from Deptt;  
  
Table created.
```

14. Delete all the rows from COPYOF DEPT table.

Query:

```
SQL> delete from copyof_Dept;  
  
4 rows deleted.
```

15. Remove COPYOF DEPT table.

Query:

```
SQL> drop table copyof_Dept;  
  
Table dropped.  
  
SQL> |
```

**Aim: To know how the constraints are used to make table is consistent.**

### Exercise – 3:

Execute the following Query on the Db to display and discuss the integrity constraints violated by any of the following operations

1. Insert ('Robert', 'F', 'Scott', '235', '21-JUN-1990', 'Bangalore', M, 58000, '100', 1 ) into EMPLOYEE.

Output:

```
SQL> insert into EMPLOYEE values ('Robert', 'F', 'Scott', '235', '21-JUN-1990', 'Bangalore', M,
2 58000, '100', 1 )
3 ;
insert into EMPLOYEE values ('Robert', 'F', 'Scott', '235', '21-JUN-1990', 'Bangalore', M,
*
ERROR at line 1:
ORA-00984: column not allowed here

SQL>
```

*Since department number 1 doesn't exist, the last column is not allowed here.*

2. Insert ( 'ProjectF', null, 'Chennai', 3 ) into Project.

```
SQL> insert into project
2 values('ProjectF',NULL,'Chennai',3);
```

```
ORA-01400: cannot insert NULL into ("SQL_XBONFHTXZGAGPISODQLEDIVQE"."PROJECT"."Project Number") ORA-06512: at "SYS.DBMS_SQL", line 1721
```

*Project name cannot be set as null*

3. Insert ( 'ProjectF', 1234, 'Chennai', 4 ) into Project.

```
SQL> INSERT INTO Project
2 VALUES ('ProjectF', 1234, 'Chennai', 4);

1 row created.

SQL> |
```

## Alter the tables to

1. Add Foreign Keys using Alter Table [if not done earlier].

```
SQL> alter table Employee
2 Add constraint fk_employee_department
3 foreign key(Dept_No) References Deptt(Dept_No)
4 ON DELETE CASCADE;

SQL> alter table project
2 Add constraint fk_project_Department
3 Foreign Key(Dept_No)References Dept(Dept_No)
4 On delete set NULL;

SQL> Alter table Department
2 Add constraint fk_department_employee
3 foreign key(ManagerSSN) References Employee(SSN_No)
4 on delete set NULL;
```

2. Drop Foreign key defined on SuperSSN and add it using Alter table command.

Query:

```
SQL> alter table Employee
2 drop foreign key fk_employee_supervisor;
```

3. Make name of Project as Unique and sex of employee as not null.

Query:

```
SQL> alter table employee
2 add constraint fk_employee_supervisor
3 Foreign key(supervisor_ssn) references Employee(SSN_NO)
4 ON DELETE SET NULL;
```

3. In the copy table add the columns door no, street, city, State, Continent.

Query:

```
SQL> ALTER TABLE PROJECT
2 ADD CONSTRAINT uc project name UNIQUE(Project name);
```

```
SQL> Alter table employee
2 Modify sex Char(1) NOT NULL;
Table altered.
```

```
SQL> Alter table CopyOFDEPT
2 ADD doorno VARCHAR(10),
3 Street Varchar(50),
4 City Varchar(50),
5 State Varchar(50),
6 Continent Varchar(50);
```

4. Make salary of employee to accept real values.

Query:

```
SQL> Alter table Employee
2 Modify SALARY REAL;
Table altered.
```



### **Exercise – 4:**

1. Create the tables described below:

Table Name: Client\_Master

Column Name    DataType

Clientno    Varchar2(6)

Name    Varchar2(20)

Address    Varchar2(30)

City    Varchar2(10)

Pincode    Number(8)

State    Varchar2(15)

Baldue    Number(10,2)

Table Name: Product\_Master

Column Name    DataType

Productno    Varchar2(6)

Description    Varchar2(15)

Um(Unitmeasure)    Varchar2(10)

QOH(Qty on hand)    Number(10)

Reorderlvl    Number(5)

Sellprice    Number(8,2)

## Costprice Number(8,2)

### Query and output:

```

Connected to:
Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options

SQL> Create Table Client_Master(Clientno varchar(6),Name varchar(20),Address varchar(30),City(varcha
2

SQL> Create Table Client_Master(Clientno varchar(6),Name varchar(20),Address varchar(30),City varcha
r(10),Pincode Number(8),State varchar(15),Baldue Number(10,2));

Table created.

SQL> desc Client_Master
Name                                         Null?     Type
-----
CLIENTNO                                     VARCHA2(6)
NAME                                         VARCHA2(20)
ADDRESS                                     VARCHA2(30)
CITY                                         VARCHA2(10)
PINCODE                                     NUMBER(8)
STATE                                       VARCHA2(15)
BALDUE                                     NUMBER(10,2)

SQL> |

SQL> Create Table Product_Master(Product_NO varchar(6),Description varchar(15),unit_measure varchar(
10),quan_on_hand number(10),reorder_lvl number(5),sell_price number(8,2),cost_price number(8,2));

Table created.

SQL> desc product_master;
Name                                         Null?     Type
-----
PRODUCT_NO                                 VARCHA2(6)
DESCRIPTION                                VARCHA2(15)
UNIT_MEASURE                              VARCHA2(10)
QUAN_ON_HAND                              NUMBER(10)
REORDER_LVL                               NUMBER(5)
SELL_PRICE                                NUMBER(8,2)
COST_PRICE                                NUMBER(8,2)

SQL>

```

## 2. Insert the following data into their respective tables:

Data for Client\_Master table

Clientno Name Address City Pincode State Baldue

C00001 Asha A/14, worli Mumbai 400054

Maharashtra 15000

C00002 Lakshmi 65, Nariman Bangalore 560001

Karnataka 0

C00003 Puja P-76, Bandra Mumbai 400002  
Maharashtra 10000

C00004 Shankar A/6, Juhu Mangalore 400044  
Karnataka 5000

C00005 Deepak 68, TNagar Chennai 6320018  
Tamilnadu 0

Data for Product\_Master table

Productno Description Um QOH Reorderlvl Sellprice  
Costprice

P00001 T-Shirts Piece 200 40 350 250

P0345 Shirts Piece 150 50 500 350

P06744 Cotton Jeans Piece 50 20 600 450

P0990 Chuddy Piece 100 40 400 250

P08823 Sarries Piece 60 50 500 300

Query and output:

```
SQL> Insert into Client_Master values('C00002','LAKSHMI','65 NARIMAN','BANGLORE',560001,'KARNATAKA',0);
```

```
1 row created.
```

```
SQL> Insert into Client_Master values('C00003','PUJA','P-76,BANDRA','MUMBAI',400002,'MAHARASTRA',10000);
```

```
1 row created.
```

```
SQL> Insert into Client_Master values('C00004','SHANKAR','A/6,JUHU','MANGLORE',400044,'KARNATAKA',5000);
```

```
1 row created.
```

```
SQL> Insert into Client_Master values('C00005','DEEPAK','68,TNAGAR','CHENNAI',6320018,'TAMILNADU',0);
```

```
1 row created.
```

```

SQL> set linesize 200;
SQL> select* from Client_Master;

```

CLIENT NAME	ADDRESS	CITY	PINCODE	STATE	BALDUE
C00001 ASHA	A/14,WORLI	MUMBAI	400054	MAHARASHTRA	15000
C00002 LAKSHMI	65 NARIMAN	BANGLORE	560001	KARNATAKA	0
C00003 PUJA	P-76,BANDRA	MUMBAI	400002	MAHARASTRA	10000
C00004 SHANKAR	A/6,JUHU	MANGLORE	400044	KARNATAKA	5000
C00005 DEEPAK	68,TNAGAR	CHENNAI	6320018	TAMILNADU	0

```

SQL> insert into Product_Master values('P00001','T-SHIRTS','PIECE',200,40,350,250);
1 row created.
SQL> insert into Product_Master values('P0345','SHIRT','PIECE',150,50,500,350);
1 row created.
SQL>
SQL> insert into Product_Master values('P06744','COTTON JEANS','PIECE',50,20,600,450);
1 row created.
SQL> insert into Product_Master values('P0990','CHUDDY','PIECE',100,40,400,250);
1 row created.
SQL> insert into Product_Master values('P08823','SARRIES','PIECE',60,50,500,300);
1 row created.
SQL> SELECT* FROM Product_Master;

```

PRODUC	DESCRIPTION	UNIT_MEASU	QUAN_ON_HAND	REORDER_LVL	SELL_PRICE	COST_PRICE
P00001	T-SHIRTS	PIECE	200	40	350	250
P0345	SHIRT	PIECE	150	50	500	350
P06744	COTTON JEANS	PIECE	50	20	600	450
P0990	CHUDDY	PIECE	100	40	400	250
P08823	SARRIES	PIECE	60	50	500	300

```

SQL> |

```

### 3. Retrieving records from a table

a. Find out the names of all the clients

Query and output:

```

SQL> select distinct name from Client_Master
2 ;

```

```

NAME
-----
DEEPAK
LAKSHMI
ASHA
SHANKAR
PUJA

```

b. Retrieve the entire contents of the Client\_Master table

Query and output:

```
SQL> SELECT * from Client_Master;
```

CLIENT NAME	ADDRESS	CITY	PINCODE	STATE	BALDUE
C00001 ASHA	A/14,WORLI	MUMBAI	400054	MAHARASHTRA	15000
C00002 LAKSHMI	65 NARIMAN	BANGLORE	560001	KARNATAKA	0
C00003 PUJA	P-76,BANDRA	MUMBAI	400002	MAHARASTRA	10000
C00004 SHANKAR	A/6, JUHU	MANGLORE	400044	KARNATAKA	5000
C00005 DEEPAK	68,TNAGAR	CHENNAI	6320018	TAMILNADU	0

```
SQL> |
```

c. Retrieve the list of names, city and the state of all the clients

Query and output:

```
SQL> select name,city,state from Client_Master;
```

NAME	CITY	STATE
ASHA	MUMBAI	MAHARASHTRA
LAKSHMI	BANGLORE	KARNATAKA
PUJA	MUMBAI	MAHARASTRA
SHANKAR	MANGLORE	KARNATAKA
DEEPAK	CHENNAI	TAMILNADU

d. List the various products available from the Product\_Master table.

Query and output:

```
SQL> select distinct description from Product_Master;
```

DESCRIPTION
COTTON JEANS
CHUDDY
T-SHIRTS
SHIRT
SARRIES

```
SQL> |
```

e. List all the clients who are located in Mumbai.

Query and output:

```
SQL> select * from client_master where City = 'MUMBAI';
```

CLIENT NAME	ADDRESS	CITY	PINCODE	STATE	BALDUE
C00001 ASHA	A/14,WORLI	MUMBAI	400054	MAHARASHTRA	15000
C00003 PUJA	P-76,BANDRA	MUMBAI	400002	MAHARASTRA	10000

```
SQL>
```

## 4. Updating records in a table

a. Change the City of Clientno 'C00001' to 'Chennai'

Query and output:

```
SQL> update Client_Master set CITY='CHENNAI' where clientno='C00001';
```

1 row updated.

```
SQL> select * from client_master;
```

CLIENT NAME	ADDRESS	CITY	PINCODE	STATE	BALDUE
C00001 ASHA	A/14,WORLI	CHENNAI	400054	MAHARASHTRA	15000
C00002 LAKSHMI	65 NARIMAN	BANGLORE	560001	KARNATAKA	0
C00003 PUJA	P-76,BANDRA	MUMBAI	400002	MAHARASTRA	10000
C00004 SHANKAR	A/6,JUHU	MANGLORE	400044	KARNATAKA	5000
C00005 DEEPAK	68,THAGAR	CHENNAI	6320018	TAMILNADU	0

SQL>

b. Change the Baldue of Clientno 'C00005' to Rs.1000

Query and output:

```
SQL> update client_master set Baldue=1000 where ClientNo = 'C00005';
```

1 row updated.

c. Change the cost price of 'shirts' to Rs. 400

Query and output:

```
SQL> update Product_Master set quan_on_hand=150 where description='T-SHIRTS';
```

1 row updated.

```
SQL> select * from Product_Master;
```

PRODUC	DESCRIPTION	UNIT_MEASU	QUAN_ON_HAND	REORDER_LVL	SELL_PRICE	COST_PRICE
P00001	T-SHIRTS	PIECE	150	40	350	250
P0345	SHIRT	PIECE	150	50	500	400
P06744	COTTON JEANS	PIECE	50	20	600	450
P0990	CHUDDY	PIECE	100	40	400	250
P08823	SARRIES	PIECE	60	50	500	300

d. Change the QOH of the 'T-shirts' to 150

Query and output:

```
SQL> update Product_Master set quan_on_hand=150 where description='T-SHIRTS';
```

```
1 row updated.
```

```
SQL> select * from Product_Master;
```

PRODUC	DESCRIPTION	UNIT_MEASU	QUAN_ON_HAND	REORDER_LVL	SELL_PRICE	COST_PRICE
P00001	T-SHIRTS	PIECE	150	40	350	250
P0345	SHIRT	PIECE	150	50	500	400
P06744	COTTON JEANS	PIECE	50	20	600	450
P0990	CHUDDY	PIECE	100	40	400	250
P08823	SARRIES	PIECE	60	50	500	300

## 5. Deleting records in a table

a. Delete all products from Product\_Master where the quantity on hand is equal to 100

Query and output:

```
SQL> delete from Product_Master where quan_on_hand=100;
```

```
1 row deleted.
```

```
SQL> select * from product_master;
```

PRODUC	DESCRIPTION	UNIT_MEASU	QUAN_ON_HAND	REORDER_LVL	SELL_PRICE	COST_PRICE
P00001	T-SHIRTS	PIECE	150	40	350	250
P0345	SHIRT	PIECE	150	50	500	400
P06744	COTTON JEANS	PIECE	50	20	600	450
P08823	SARRIES	PIECE	60	50	500	300

```
SQL> |
```

b. Delete from Client\_Master where the column state holds the value “Tamilnadu”

Query and output:

```
SQL> delete from client_master where state='TAMILNADU';
```

```
1 row deleted.
```

```
SQL> delete from client_master where BALDUE=0;
```

```
1 row deleted.
```

```
SQL> SELECT * FROM Client_Master;
```

CLIENT NAME	ADDRESS	CITY	PINCODE	STATE	BALDUE
C00001 ASHA	A/14, WORLI	CHENNAI	400054	MAHARASHTRA	15000
C00003 PUJA	P-76, BANDRA	MUMBAI	400002	MAHARASTRA	10000
C00004 SHANKAR	A/6, JUHU	MANGLORE	400044	KARNATAKA	5000

```
SQL>
```

c. Delete all clients from Client\_Master where the Baldue is equal to 0.

## 6. Altering the table structure

a. Add a column call Profitpercent of datatype number10,2 to the Product\_Master

Query and output:

```
SQL> alter table Product_Master add Profit_percent Number(10,2);  
Table altered.
```

b. Change the size of sellprice column in Product\_Master to 10,2

Query and output:

```
SQL> alter table product_master modify Sell_price Number(10,2);  
Table altered.  
SQL> alter table product_master rename column unit_measure to UNITmeasure;  
Table altered.
```

c. Change the name of the column UM to UnitMeasure in Product\_Master

Query and output:

```
SQL> alter table product_master rename column unit_measure to UNITmeasure;  
Table altered.
```

d. Drop the column UnitMeasure.

Query and output:



```
SQL> alter table Product_Master DROP COLUMN UNITMEASURE;
```

```
Table altered.
```

```
SQL> select * from product_master;
```

PRODUC	DESCRIPTION	QUAN_ON_HAND	REORDER_LVL	SELL_PRICE	COST_PRICE	PROFIT_PERCENT
P00001	T-SHIRTS	150	40	350	250	
P0345	SHIRT	150	50	500	400	
P06744	COTTON JEANS	50	20	600	450	
P08823	SARRIES	60	50	500	300	

```
SQL> |
```

e. Drop the table Client\_Master along with its data

Query and output:

```
SQL> drop table Client_Master;
```

```
Table dropped.
```

```
SQL> SELECT * from client_Master;
```

```
SELECT * from client_Master
```

```
*
```

```
ERROR at line 1:
```

```
ORA-00942: table or view does not exist
```

```
SQL> |
```

## Exercise 5:

**Aim: To understand different operators in SQL**

1. Find the employee names having salary greater than Rs.45000.

Query:

```
SQL> select First_name from Employee where SALARY>45000;
```

Output:

```
FIRST_NAME
```

```
Doug  
Joyce  
Rahul
```

```
SQL> |
```

2. Find the employee names whose salary lies in the range between 35000 and 75000.

Query:

```
SQL> select First_name from Employee where SALARY>35000 and SALARY<75000;
```

Output:

```
FIRST_NAME
-----
Joyce
Franklin
Jennifer
Ramesh
Rahul
SQL> |
```

2. Find the employees who have no supervisor.

Query:

```
SQL> select * from employee where Supervisor_ssn IS NULL;
```

Output:

```
FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY  ADDRESS
-----
Doug            E  Gilbert        123      09-JUN-68 Chennai
Joyce          PAN          124      07-FEB-73 Uellore
SQL> |
```

3. Display the employee names having 'salt lake' in their address.

Query:

```
SQL> select first_name from employee where Address='Salt lake';
no rows selected
```

No output.

5.Display the department name that starts with 'H'.

Query:

```
SQL> select dept_name from deptt where dept_name like 'H%';
```

Output:

```
DEPT_NAME
-----
Headquater
SQL>
```

6. Display the project numbers along project name that ends with 'i' in project location.

Query:

```
SQL> select project_name, project_no from project where project_location LIKE '%i';
```

Output:

```
PROJECT_NAME  PROJECT_NO
-----
ProjectA      3388
ProjectC      6688
ProjectD      2423
SQL> |
```

7. Display the names of all the employees having supervisor with any of the following SSN 123, 533.

Query:

```
SQL> select first_name from employee where SUPERVISOR_SSN='123' or SUPERVISOR_SSN='533';
```

Output:

```
FIRST_NAME
-----
Franklin
Jennifer
Rahul
SQL>
```

8. Display all the employee details based on the salary in descending order.

Query:

```
SQL> select * from employee order by salary desc;
```

Output:

```
FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY  ADDRESS
-----
Doug            E  Gilbert         123       09-JUN-68 Chennai
Joyce           PAN         124       07-FEB-73 Vellore
Rahul           S  Ram             555       10-OCT-73 Vellore
Jennifer        S  Wallace         564       20-JUN-83 Chennai
Franklin        T  Wong            125       08-DEC-72 Delhi
Ramesh          K  Narayan         234       15-SEP-85 Bangalore
John            B  Smith           678       09-JAN-87 Madurai

7 rows selected.

SQL> |
```

9. Display the employees belongs to Chennai or vellore.

Query:

```
SQL> select * from employee where address='Chennai' or address='Vellore';
```

Output:

FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
Doug	E	Gilbert	123	09-JUN-68	Chennai
Joyce		PAN	124	07-FEB-73	Vellore
Jennifer	S	Wallace	564	20-JUN-83	Chennai
Rahul	S	Ram	555	10-OCT-73	Vellore

```
SQL> |
```

10.Display the department details in ascending order of department number.

Query:

```
SQL> select * from deptt order by dept_no asc;
```

Output:

DEPT_NAME	DEPT_NO	MANAGERSS	MANAGERST
Headquater	1	678	16-DEC-14
Administration	2	564	03-JAN-12
Finance	3	234	18-MAY-13
IT	4	123	12-JUN-15

```
SQL> |
```

11.Display all the MgrSSN and MgrStartDate as ManagerSSN and Manager\_DOJ.

## Query:

```
SQL> ALTER TABLE deptt
  2  RENAME COLUMN MGRSSN TO MANAGERSSN;

Table altered.

SQL> ALTER TABLE deptt
  2  RENAME COLUMN Managerstartdate to Manager_DOJ;

Table altered.
```

## Output:

```
SQL> desc deptt;
Name
-----
DEPT_NAME
DEPT_NO
MANAGERSSN
MANAGER_DOJ

SQL> |
```

12.Display all the male employees with SSN and salary

## Query:

```
SQL> select first_name, ssn_number, salary from employee where sex='M';
```

## Output:

FIRST_NAME	SSN_NUMBE	SALARY
Doug	123	80000
Franklin	125	40000
John	678	30000
Ramesh	234	38000
Rahul	555	50000

```
SQL> |
```