**STRUCTURED QUERY LANGUAGE**

**DATA DEFINITION LANGUAGE (DDL):**

**1.Create command:**

SQL> create table amma(sno number(2),sname varchar2(10),age number(2));

Table created.

SQL> desc amma

Name Null? Type

----------------------------------------- -------- ----------------------------

SNO NUMBER(2)

SNAME VARCHAR2(10)

AGE NUMBER(2)

**2.Alter command:**

**A.Alter add:**

SQL> alter table amma add(phonenum varchar2(10));

Table altered.

SQL> desc amma

Name Null? Type

----------------------------------------- -------- ---------------------------

SNO NUMBER(2)

SNAME VARCHAR2(10)

AGE NUMBER(2)

PHONENUM VARCHAR2(10)

**B.Alter modify:**

SQL> alter table amma modify(sno varchar2(10));

Table altered.

SQL> desc amma;

Name Null? Type

----------------------------------------- -------- ----------------------------

SNO VARCHAR2(10)

SNAME VARCHAR2(10)

AGE NUMBER(2)

PHONENUM VARCHAR2(10)

**C.Alter drop:**

SQL> alter table amma drop column phonenum;

Table altered.

SQL> desc amma

Name Null? Type

----------------------------------------- -------- ----------------------------

SNO VARCHAR2(10)

SNAME VARCHAR2(10)

AGE NUMBER(2)

**D.Alter rename:**

SQL> alter table amma rename column sno to sid;

Table altered.

SQL> desc amma;

Name Null? Type

----------------------------------------- -------- --------------

SID VARCHAR2(10)

SNAME VARCHAR2(10)

AGE NUMBER(2)

**3.Drop command:**

SQL> drop table std\_per;

Table dropped.

SQL> desc std\_per;

ERROR:ORA-04043: object std\_per does not exist

**4.Truncate command:**

SQL> select \* from amma;

SNO SNAME AGE PHONENUM

---------- ---------- ---------- ----------

s01 madhav 20 9966589691

s02 madhava 21 7981449762

SQL> truncate table amma;

Table truncated.

SQL> select \* from amma;

no rows selected

**DATA MANIPULATION LANGUAGE (DML):**

**Insert command:**

**A.Static insertion:**

SQL>insert into amma values('s01','madhav',20,'987654321');

1 row created.

SNO SNAME AGE PHONENUM

---------- ---------- ---------- ----------

s01 madhav 20 9966589691

**B.Dynamic insertion:**

SQL> insert into amma values('&sno','&sname',&age,'&phonenum');

Enter value for sno: s02

Enter value for sname: madhava

Enter value for age: 21

Enter value for phonenum: 7981449762

old 1: insert into amma values('&sno','&sname',&age,'&phonenum')

new 1: insert into amma values('s02','madhava',21,'7981449762')

1 row created.

SQL> select \* from amma;

SNO SNAME AGE PHONENUM

---------- ---------- ---------- ----------

s01 madhav 20 9966589691

s02 madhava 21 7981449762

**2.Update command:**

SQL> update amma set phno='9966589691' where sno='s01';

1 row updated.

SQL> select \* from amma;

SNO SNAME AGE PHNO

---------- ---------- ---------- ----------

s01 madhav 20 9966589691

s02 madhava 21 7981449762

**3.Delete command:**

SQL> delete from amma where sno='s01';

1 row deleted.

SQL> select \* from amma;

SNO SNAME AGE PHNO

---------- ---------- ---------- ----------

s02 madhava 21 7981449762

**4.Rename command:**

SQL> rename amma to stud\_7;

Table renamed.

SQL> select \* from stud\_7;

SNO SNAME AGE PHNO

---------- ---------- ---------- ----------

s02 madhava 21 7981449762

**TRANSACTION CONTROL LANGUAGE (TCL COMMANDS):**

SQL> select \* from amma7;

EMPID ENAME SAL

---------- ---------- ----------

e01 madhav 15000

e02 rao 25000

**1.Savepoint command:**

SQL> savepoint tsp;

Savepoint created.

SQL>insert into amma7 values (3,'madhava',35000);

1 row created.

SQL> update amma7 set sal=15000;

2 rows updated.

**Rollback command:**

SQL> rollback to tsp;

Rollback complete.

SQL> select \* from amma7;

EMPID ENAME SAL

---------- ---------- ----------

e01 madhav 15000

e02 rao 25000

**Commit command:**

SQL> select \*from amma7;

ENO ENAME DESG SAL

---------- ----- ---------- ----------

1 asd manager 25000.05

SQL> select \*from stud8;

SNO SNAME PHNO

---------- ---------- ----------

1 xyz 9999988888

2 abc 7778889992

SQL> select \* from amma7;

ENO ENAME DESG SAL

---------- ----- ---------- ----------

1 efgh manager 23000.75

2 opqrs clerk 24000

SQL> savepoint spt;

Savepoint created.

SQL> update stud8 set phno='9966589691' where sno=1;

1 row updated.

SQL> update amma7 set sal=25000.50 where eno=1;

1 row updated.

**3.Commit command:**

SQL> commit;

Commit complete.

SQL> select \*from amma7;

ENO ENAME DESG SAL

---------- ----- ---------- ----------

1 efgh manager 25000.5

2 opqrs clerk 24000

SQL> select \*from stud8;

SNO SNAME PHNO

---------- ---------- ----------

1 xyz 9966589691

2 abc 7778889992

**DATA QUERRY LANGUAGE (DQL COMMANDS):**

**Select command:**

SQL> select \* from emp08;

EID ENAME SAL DOJ DEPTNO

---------- ---------- ---------- ---------- ----------

1 x 20000 2010 1

2 y 30000 2015 2

3 z 40000 2007 1

4 w 35000 2011 3

**A.Using AS:**

SQL> select eid as empid,doj as dateofbirth from emp08;

EMPID DATEOFBIRTH

---------- -----------

1 2010

2 2015

3 2007

4 2011

**B.Using||:**

SQL> select ename||' has this '||doj from emp08;

ENAME||'HASTHIS'||DOJ

------------------------------------------------------------

x has this 2010

y has this 2015

z has this 2007

w has this 2011

**C.Using where:**

SQL> select \* from emp08 where deptno=1;

EID ENAME SAL DOJ DEPTNO

---------- ---------- ---------- ---------- ----------

1 x 20000 2010 1

3 z 40000 2007 1

**CONSTRAINTS:**

**NOTNULL & PRIMARY KEY CONSTRAINT:**

SQL> create table student8(sno number(2) primary key,sname varchar2(10) not null,phno varchar2(10));

Table created.

SQL> desc student8;

Name Null? Type

----------------------------------------- -------- ------------------

SNO NOT NULL NUMBER(2)

SNAME NOT NULL VARCHAR2(10)

PHNO VARCHAR2(10)

**UNIQUE KEY CONSTRAINT:**

SQL> create table emp\_8(eno number(10),ename varchar2(10),phno varchar2(10) unique);

Table created.

SQL> desc emp\_8;

Name Null? Type

----------------------------------------- -------- -------------

ENO NUMBER(10)

ENAME VARCHAR2(10)

PHNO VARCHAR2(10)

**FORIEGN KEY & CHECK KEY COSTRAINT:**

SQL> create table stud\_per(sno number(2),age number(2),foreign key(sno) references student8(sno),check (age>=20));

Table created.

SQL> desc stud\_7;

Name Null? Type

----------------------------------------- -------- -------------------

SNO VARCHAR2(10)

SNAME VARCHAR2(10)

AGE NUMBER(2)

PHNO VARCHAR2(10)

**BUILT IN FUNCTIONS**

**1.CHARACTER /STRING FUNCTIONS:**

**A.CASE CHARACTER FUNCTIONS:**

**LOWER():**

SQL> select lower('MADHAVA') from dual;

LOWER(

------

madhava

**UPPER():**

SQL> select upper ('college') from dual;

UPPER('

-------

COLLEGE

**INITCAP():**

SQL> select initcap('madhav') from dual;

INITCA

------

Madhav

**B.CHARACTER MANIPULATION FUNCTIONS:**

**LENGTH():**

SQL> select length('aditya pg college') from dual;

LENGTH('ADITYAPGCOLLEGE')

-------------------------

17

**LTRIM():**

SQL> select ltrim('aditya','a') from dual;

LTRIM

-----

ditya

**RTRIM():**

SQL> select rtrim('aditya','a') from dual;

RTRIM

-----

adity

**LPAD():**

SQL> select lpad('aditya',10,'\*') from dual;

LPAD('ADIT

----------

\*\*\*\*aditya

**RPAD():**

SQL> select rpad('aditya',10,'&') from dual;

RPAD('ADIT

----------

aditya&&&&

**REPLACE():**

SQL> select replace('madhavi','i','a') from dual;

REPLACE

-------

madhava

**CONCAT():**

SQL> select concat('aditya',' college') from dual;

CONCAT('ADITYA

--------------

aditya college

**SUBSTR():**

SQL> select substr('aditya engineering college',7,5) from dual;

SUBST

-----

engi

**INSTR():**

SQL> select instr('aditya engineering college','engineering') from dual;

INSTR('ADITYAENGINEERINGCOLLEGE','ENGINEERING')

-----------------------------------------------

8

**NUMERIC/ARITHEMATIC FUNCTIONS**:

**ABS():**

SQL> select abs(-5) from dual;

ABS(-5)

----------

5

**CEIL():**

SQL> select ceil(24.52) from dual;

CEIL(24.52)

-----------

25

**FLOOR():**

SQL> select floor(24.52) from dual;

FLOOR(24.52)

------------

24

**TRUNC():**

SQL> select trunc(24.567,2) from dual;

TRUNC(24.567,2)

---------------

24.56

**ROUND():**

SQL> select round(24.567,2) from dual;

ROUND(24.567,2)

---------------

24.57

**POWER():**

SQL> select power(5,2) from dual;

POWER(5,2)

----------

25

**SQRT():**

SQL> select sqrt(25) from dual;

SQRT(25)

----------

5

**MOD():**

SQL> select mod(64,5) from dual;

MOD(64,5)

----------

4

**SIGN():**

SQL> select sign(-5) from dual;

SIGN(-5)

----------

-1

**SIN():**

SQL> select sin(45) from dual;

SIN(45)

----------

.850903525

**COS():**

SQL> select cos(45) from dual;

COS(45)

----------

.525321989

**TAN():**

SQL> select tan(45) from dual;

TAN(45)

----------

1.61977519

**DATE &TIME FUNCTONS:**

**SYSDATE:**

SQL> select sysdate from dual;

SYSDATE

---------

05-JUL-17

**MONTHS\_BETWEEN():**

SQL> select months\_between(sysdate,'13-may-1996') from dual;

MONTHS\_BETWEEN(SYSDATE,'13-MAY-1996')

-------------------------------------

253.764802

**ADD\_MONTHS():**

SQL> select add\_months(sysdate,10) from dual;

ADD\_MONTH

---------

05-MAY-18

**NEXT\_DAY():**

SQL> select next\_day('06-jul-17',2) from dual;

NEXT\_DAY(

---------

10-JUL-17

**LAST\_DAY():**

SQL> select last\_day('07-feb-14') from dual;

LAST\_DAY(

---------

28-FEB-14

**Conversion function:**

**TO\_CHAR():**

SQL> select to\_char(sysdate,'dd-mon-yyyy hh:mi:ss:am') from dual;

TO\_CHAR(SYSDATE,'DD-MON

-----------------------

17-jul-2017 09:48:47:am

**AGGREGATE FUNCTIONS:**

SQL> select \*from students8;

SNO SNAME DBMS UNIX

--------- ---------- ---------- ----------

1 jai 45 44

2 madhav37 43

3 sri56 54

4ram53 56

5 rama 43 54

**SUM():**

SQL> select sum(dbms) from students8;

SUM(DBMS)

----------

234

**COUNT():**

SQL> select count(dbms) from students8;

COUNT(DBMS)

-----------

5

SQL> select count(\*) from students8;

COUNT(\*)

----------

5

**AVG():**

SQL> select avg(dbms) from students8;

AVG(DBMS)

----------

46.8

**MIN():**

SQL> select min(dbms) from students8;

MIN(DBMS)

----------

37

**MAX():**

SQL> select max(dbms) from students8;

MAX(DBMS)

----------

56

**GROUP BY FUNCTIONS:**

**1.GROUPBY clause:**

SQL> select rating,max(age) from sailors group by rating;

RATING MAX(AGE)

--------- ----------

1 33

6 24

5 24

8 56

7 45

3 64

9 35

10 35

8 rows selected.

**2.HAVING clause:**

SQL> select s.rating,min(age) from sailors8 s where age>=18 group by rating having count(\*)>1;

RATING MIN(AGE)

---------- ----------

8 25.5

7 35

3 25.5

**3.ORDER BY clause:**

SQL> select \* from sailors8 order by age desc;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

95 bob 3 63.5

31 lubber 8 55.5

22 dustin 7 45

58 rusty 10 35

74 horatio 9 35

64 horatio 7 35

29 brutus 1 33

85 art 3 25.5

32 andy 8 25.5

71 zorba 10 16

10 rows selected.

**SPECIAL OPERATORS**

**1.COMPARISION OPERATORS:**

**BETWEEN-AND**:

SQL> select \* from sailors8 where age between 25 and 33;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

29 brutus 1 33

32 andy 8 25.5

85 art 3 25.5

**IN:**

SQL> select \* from sailors where age in(25,30,35,40);

SID SNAME RATING AGE

---------- ---------- ---------- ----------

58 rusty 10 35

64 horatio 7 35

74 horatio 9 35

**LIKE:**

SQL> select \*from sailors8 where sname like 'bob';

SID SNAME RATING AGE

---------- ---------- ---------- ----------

95 bob 3 63.5

**2.LOGICAL OPERATORS:**

**AND:**

SQL> select \*from sailors8 where sname='bob' and rating=3;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

95 bob 3 63.5

**OR:**

SQL> select \*from sailors8 where sname like 'bob' or rating=5;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

95 bob 3 63.5

**NOT:**

SQL> select \*from sailors8 where rating not in(1,3,7);

SID SNAME RATING AGE

---------- ---------- ---------- ----------

31 lubber 8 55.5

32 andy 8 25.5

58 rusty 10 35

71 zorba 10 16

74 horatio 9 35

**3.RELATIONAL OPERATORS:**

SQL> select \* from sailors8 where sid>=60 and rating >5;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

64 horatio 7 35

71 zorba 10 16

74 horatio 9 35

SQL> select \*from sailors8 where age<35 and rating<=7;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

29 brutus 1 33

85 art 3 25.5

SQL> select \*from sailors8 where rating=3 or age!=35;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

22 dustin 7 45

29 brutus 1 33

31 lubber 8 55.5

32 andy 8 25.5

71 zorba 10 16

85 art 3 25.5

95 bob 3 63.5

7 rows selected.

SQL> select \*from reserves8 where sid>31;

SID BID DAY

---------- ---------- ---------

64 101 09-MAY-98

64 102 09-AUG-98

74 103 09-AUG-98

**4.SET OPERATORS:**

**UNION:**

SQL> select \* from sailors8 where age>25 union select \* from sailors8 where rating>5;

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 35

85 art 3 25.5

95 bob 3 63.5

10 rows selected.

**INTERSECT:**

SQL> select \*from sailors8 where age>25 intersect select \*from sailors8 where rating>5;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

22 dustin 7 45

31 lubber 8 55.5

32 andy 8 25.5

58 rusty 10 35

64 horatio 7 35

74 horatio 9 35

6 rows selected.

**EXCEPT:**

SQL> select \*from sailors8 where age>25 minus select \* from sailors8 where rating>5;

SID SNAME RATING AGE

---------- ---------- ---------- ----------

29 brutus 1 33

85 art 3 25.5

95 bob 3 63.5

**CROSS JOIN:**

SQL> select \*from dept8;

DEPTNO DEPTNAME LOC

---------- ---------- ----------

1 inventory pune

2 research hyd

3 develop delhi

4 design Chennai

SQL> select \*from emp08;

EID ENAME SAL DOJ DEPTNO

---------- ---------- ---------- ---------- ----------

1 x 20000 2010 1

2 y 30000 2015 2

3 z 40000 2007 1

4 w 35000 2011 3

SQL> select \* from emp08 cross join dept8;

EID ENAME SAL DOJ DEPTNO DEPTNO DEPTNAME LOC

---------- ---------- ---------- ---------- ---------- ---------- ----------------

1 x 20000 2010 1 1 inventory pune

1 x 20000 2010 1 2 researchhyd

1 x 20000 2010 1 3 developdelhi

1 x 20000 2010 1 4 designchennai

2 y 30000 2015 2 1 inventorypune

2 y 30000 2015 2 2 research hyd

2 y 30000 2015 2 3 developdelhi

2 y 30000 2015 2 4 design chennai

3 z 40000 2007 1 1 inventory pune

3 z 40000 2007 1 2 research hyd

3 z 40000 2007 1 3 develop delhi

3 z 40000 2007 1 4 design chennai

4 w 35000 2011 3 1 inventory pune

4 w 35000 2011 3 2 research hyd

4 w 35000 2011 3 3 develop delhi

4 w 35000 2011 3 4 design chennai

16 rows selected.

**JOINS:**

SQL> select\* from department8;

DEPTNO DNAME LOC

---------- ---------- ----------

10 inventory hyd

20 finanace bglr

30 hr Mumbai

SQL> select \* from employee8;

EMPNO ENAME JOB MGR DEP

---------- ---------- ---------- ---------- ---

111 saketh analyst 444 10

222 sudha clerk 333 20

333 jagan manager 111 10

444 madhu engineer 222 40

**EQUI-JOIN:**

SQL> select e.empno,e.ename,e.job,d.dname,d.loc from employee8 e,department8 d where e.deptno=d.deptno;

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

**USING CLAUSE:**

SQL> select empno,ename,job,dname,loc from employee8 e join department8 d using(deptno);

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

**ON CLAUSE:**

SQL> select empno,ename,job,dname,loc from employee8 e join department8 d on (e.deptno=d.deptno);

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

**NON EQUI-JOIN:**

SQL> select empno,ename,job,dname,loc from employee8 e,department8 d where e.deptno>d.deptno;

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

222 sudha clerk inventory hyd

444 madhu engineer inventory hyd

444 madhu engineer finanace bglr

444 madhu engineer hr mumbai

**SELF JOIN:**

SQL> select e1.empno,e2.empno,e1.ename,e1.job,e2.deptno from employee8 e1,employee8 e2 where e1.empno=e2.mgr;

EMPNO EMPNO ENAME JOB DEP

---------- ---------- ---------- ---------- ---

444 111 madhu engineer 10

333 222 jagan manager 20

111 333 saketh analyst 10

222 444 sudha clerk 40

**NATURAL JOIN:**

SQL> select empno,ename,job,dname,loc from employee8 natural join department8;

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

**CROSS JOIN:**

SQL> select empno,ename,job,dname,loc from employee8 cross join department8;

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

111 saketh analyst finanace bglr

111 saketh analyst hr mumbai

222 sudha clerk inventory hyd

222 sudha clerk finanace bglr

222 sudha clerk hr mumbai

333 jagan manager inventory hyd

333 jagan manager finanace bglr

333 jagan manager hr mumbai

444 madhu engineer inventory hyd

444 madhu engineer inanace bglr

444 madhu engineer hr mumbai

12 rows selected.

**OUTER JOIN:**

**A.Left outer join:**

SQL> select empno,ename,job,dname,loc from employee8 e, department8 d where e.deptno=d.deptno(+);

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

444 madhu engineer

**B.Right outer join:**

SQL> select empno,ename,job,dname,loc from employee8 e, department8 d where e.deptno(+)=d.deptno;

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

333 jagan manager inventory hyd

222 sudha clerk finanace bglr

hr mumbai

**Full outer join:**

SQL> select empno,ename,job,dname,loc from employee8 e full outer join department8 d on(e.deptno=d.deptno);

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

444 madhu engineer

hr mumbai

**INNER JOIN:**

SQL> select empno,ename,job,dname,loc from employee8 inner join department8 using(deptno);

EMPNO ENAME JOB DNAME LOC

---------- ---------- ---------- ---------- ----------

111 saketh analyst inventory hyd

222 sudha clerk finanace bglr

333 jagan manager inventory hyd

**CASE STUDY OF SAILORS DATABASE**

**SAILORS TABLE:**

SQL> create table sailors8(sid number(3)primary key,sname varchar2(10),rating number(2),

age number(4,2));

Table created.

SQL> desc sailors8;

Name Null? Type

----------------------------------------- -------- ----------------------------

SID NOT NULL NUMBER(3)

SNAME VARCHAR2(10)

RATING NUMBER(2)

AGE NUMBER(4,2)

**BOATS TABLE:**

SQL> create table boats8(bid number(3) primary key,bname varchar2(10),color varchar2(10));

Table created.

SQL> desc boats8;

Name Null? Type

----------------------------------------- -------- ----------------------------

BID NOT NULL NUMBER(3)

BNAME VARCHAR2(10)

COLOR VARCHAR2(10)

**RESERVES TABLE:**

SQL> create table reserves8(sid number(3),bid number(3),day date,foreign key(sid) references sailors8(sid),foreign key(bid) references boats(bid));

Table created.

SQL> desc reserves8;

Name Null? Type

--------------------------- -------- ----------------------------

SID NUMBER(3)

BID NUMBER(3)

DAY DATE

**After inserting values into tables the output is:**

SQL> select \*from sailors8;

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 35

85 art 3 25.5

95 bob 3 63.5

10 rows selected.

SQL> select \* from boats8;

BID BNAME COLOR

---------- ---------- ----------

101 interlake blue

102 interlake red

103 clipper green

104 marine red

SQL> select \* from reserves8;

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.

**Queries:**

1.SQL> select avg(age) from sailors8;

AVG(AGE)

----------

36.9

2.SQL> select avg(age) from sailors8 where rating=10;

AVG(AGE)

----------

25.5

3.SQL> select sname,age from sailors8 where age=(select max(age) from sailors8);

SNAME AGE

---------- ----------

bob 63.5

4.SQL> select count(\*) from sailors8;

COUNT(\*)

----------

10

5.SQL> select count(distinct sname) from sailors8;

COUNT(DISTINCTSNAME)

--------------------

9

6.SQL> select s2.sname from sailors8 s2 where s2.age>(select max(s1.age) from sailors8 s1 where

s1.rating=10);

SNAME

----------

dustin

lubber

bob

7.SQL> select rating,min(age) from sailors8 group by rating;

RATING MIN(AGE)

---------- ----------

1 33

8 25.5

7 35

3 25.5

10 16

9 35

6 rows selected.

8.SQL> select rating,min(age) from sailors8 where age>=18 group by rating having count(\*)>1;

RATING MIN(AGE)

---------- ----------

8 25.5

7 35

3 25.5

9.SQL> select b.bid,count(\*) from boats8 b,reserves8 r where b.bid=r.bid and b.color='red' group by b.bid;

BID COUNT(\*)

---------- ----------

102 3

104 2

10.SQL> select rating,avg(age) from sailors8 group by rating having count(\*)>1;

RATING AVG(AGE)

---------- ----------

8 40.5

7 40

3 44.5

10 25.5

11.SQL> select rating,avg(age) from sailors8 where age>=18 group by rating having count(\*)>1;

RATING AVG(AGE)

---------- ----------

8 40.5

7 40

3 44.5

12.SQL> select s.sname from sailors8 s,reserves8 r where s.sid=r.sid and r.bid=103;

SNAME

----------

dustin

lubber

horatio

13.SQL> select r.sid from reserves8 r,boats8 b where r.bid=b.bid and b.color='red';

SID

----------

22

31

64

31

22

14.SQL> select s.sname from sailors8 s,reserves8 r,boats8 b where s.sid=r.sid and r.bid=b.bid

and b.color='red';

SNAME

----------

dustin

dustin

lubber

lubber

horatio

15.SQL> select b.color from sailors8 s,reserves8 r,boats8 b where b.bid=r.bid and r.sid=s.sid

and s.sname='lubber';

COLOR

----------

red

green

red

16.SQL> select s.sname from sailors8 s,reserves8 r where s.sid=r.sid;

SNAME

----------

dustin

dustin

dustin

dustin

lubber

lubber

lubber

horatio

horatio

horatio

10 rows selected.

17.SQL> select s.sname,s.rating+1 as incrating from sailors8 s,reserves8 r1,reserves8 r2 where

s.sid=r1.sid and s.sid=r2.sid and r1.day=r2.day and r1.bid<>r2.bid;

SNAME INCRATING

---------- ----------

dustin 8

dustin 8

18.SQL> select s.sname from sailors8 s,reserves8 r,boats8 b where s.sid=r.sid and r.bid=b.bid and

(b.color='red' or b.color='green');

SNAME

----------

dustin

dustin

dustin

lubber

lubber

lubber

horatio

horatio

8 rows selected.

19.SQL> select s.sname from sailors8 s,reserves8 r1,reserves8 r2,boats8 b1,boats8 b2 where s.sid=r1.sid and s.sid=r2.sid and r1.bid=b1.bid and r2.bid=b2.bid and b1.color='red' and b2.color='green';

SNAME

----------

lubber

lubber

dustin

dustin

20.SQL> select s.sid from sailors8 s,reserves8 r where s.sid=r.sid and (s.rating=10 or r.bid=104);

SID

----------

22

31

Without using joins above querry:

SQL> select s.sid from sailors8 s where s.rating=10

union

select r.sid from reserves8 r where r.bid=104;

SID

----------

22

31

58

71

21.SQL> select s.sname from sailors8 s where s.sid in(select r.sid from reserves8 r where r.bid=103);

SNAME

----------

dustin

lubber

horatio

22.SQL> select s.sname from sailors8 s where s.sid in(select r.sid from reserves8 r where r.bid in(select b.bid from boats8 b where b.color='red'));

SNAME

----------

dustin

lubber

horatio

23.SQL> select s.sname from sailors8 s where s.sid not in(select r.sid from reserves8 r where r.bid in(select b.bid from boats8 b where b.color='red'));

SNAME

----------

brutus

andy

rusty

zorba

horatio

art

bob

7 rows selected.

24.SQL> select s.sname from sailors8 s where exists (select \* from reserves8 r where

r.sid=s.sid and r.bid=103);

SNAME

----------

dustin

lubber

horatio

25.SQL> select \*from sailors8 s1 where s1.rating>any(select s.rating from sailors8 s where s.sname like

'horatio');

SID SNAME RATING AGE

---------- ---------- ---------- ----------

58 rusty 10 35

71 zorba 10 16

74 horatio 9 35

31 lubber 8 55.5

32 andy 8 25.5

26.SQL> select \* from sailors8 s1 where s1.rating>=all(select s.rating from sailors8 s);

SID SNAME RATING AGE

---------- ---------- ---------- ----------

58 rusty 10 35

71 zorba 10 16

**PL/SQL PROGRAMS**

1.PL/SQL program for arithmetic operations:

declare

a number:=&a;

b number:=&b;

begin

dbms\_output.put\_line('Addition:'||(a+b));

dbms\_output.put\_line('Substraction:'||(a-b));

dbms\_output.put\_line('Multiplication:'||(a\*b));

dbms\_output.put\_line('Division:'||(a/b));

end;

**Output:**

SQL> set line 100;

SQL> set serveroutput on;

SQL> @ Z:\madhavdbms\final\prg(1).sql.tx

10 /

Enter value for a: 08

old 2: a number:=&a;

new 2: a number:=08;

Enter value for b: 08

old 3: b number:=&b;

new 3: b number:=08;

Addition:16

Substraction:0

Multiplication:64

Division:1

PL/SQL procedure successfully completed.

SQL>.

2. PL/SQL program for biggest of 2 numbers:

SQL> edit prog2.sql

SQL> get Z:\madhavdbms\final\prog2.sql

1 declare

2 x number:=&x;

3 y number:=&y;

4 begin

5 if x>y then

6 dbms\_output.put\_line(x||'is big');

7 end if;

8 if y>x then

9 dbms\_output.put\_line(y||'is big');

10 end if;

11\* end;

SQL> /

Enter value for x: 5

old 2: x number:=&x;

new 2: x number:=5;

Enter value for y: 2

old 3: y number:=&y;

new 3: y number:=2;

PL/SQL procedure successfully completed.

SQL> set serveroutput on;

SQL> /

Enter value for x: 45

old 2: x number:=&x;

new 2: x number:=45;

Enter value for y: 10

old 3: y number:=&y;

new 3: y number:=10;

45is big

PL/SQL procedure successfully completed.

3.PL/SQL program for biggest of 3 numbers:

SQL> get Z:\madhavdbms\final\prog3.sql

1 declare

2 x number:=&x;

3 y number:=&y;

4 z number:=&z;

5 begin

6 if x>y then

7 if z>x then

8 dbms\_output.put\_line(z||'is big');

9 else

10 dbms\_output.put\_line(x||'is big');

11 end if;

12 else

13 if y>z then

14 dbms\_output.put\_line(y||'is big');

15 else

16 dbms\_output.put\_line(z||'is big');

17 end if;

18 end if;

19\* end;

SQL> /

Enter value for x: 12

old 2: x number:=&x;

new 2: x number:=12;

Enter value for y: 12

old 3: y number:=&y;

new 3: y number:=12;

Enter value for z: 12

old 4: z number:=&z;

new 4: z number:=12;

12is big

PL/SQL procedure successfully completed.

SQL> /

Enter value for x: 1

old 2: x number:=&x;

new 2: x number:=1;

Enter value for y: 12

old 3: y number:=&y;

new 3: y number:=12;

Enter value for z: 2

old 4: z number:=&z;

new 4: z number:=2;

12is big

PL/SQL procedure successfully completed.

SQL>

4. pl/sql program for to check odd or even:

SQL> edit prog4.sql

SQL> get Z:\madhavdbms\final\prog4.sql

1 declare

2 n number:=&n;

3 begin

4 if (n mod 2=0) then

5 dbms\_output.put\_line(n||'is even number');

6 else

7 dbms\_output.put\_line(n||'is odd number');

8 end if;

9\* end;

SQL> /

Enter value for n: 3

old 2: n number:=&n;

new 2: n number:=3;

3is odd number

PL/SQL procedure successfully completed.

Output:

SQL> /

Enter value for n: 4

old 2: n number:=&n;

new 2: n number:=4;

4is even number

PL/SQL procedure successfully completed.

SQL>

SQL> /

Enter value for n: 56

old 2: n number:=&n;

new 2: n number:=56;

56is even number

PL/SQL procedure successfully completed.

5. pl/sql program to update salary of an employee using nested if

Before executing pl/sql program emp table:

SQL> select \* from emp08;

EID ENAME SAL DOJ DEPTNO

---------- ---------- ---------- ---------- ----------

1 x 20000 2010 1

2 y 30000 2015 2

3 z 40000 2007 1

4 w 35000 2011 3

5 a 17500 2013 2

6 b 34000 2016 4

6 rows selected.

Pl/sql program:

declare

no emp08.eno%type:=&no;

s emp08.sal%type;

begin

select sal into s from emp08 where eno=no;

if no=1 then

s:=s+1100;

else if no=2 then

s:=s+1000;

else if no=3 then

s:=s+1000;

else if no=4 then

s:=s+1000;

else if no=5 then

s:=s+700;

else if no=6 then

s:=s+700;

else

dbms\_output.put\_line('Eno does not exist');

end if;

end if;

end if;

end if;

end if;

end if;

update emp08 set sal=s where eno=no;

end;

**output:**

SQL> /

Enter value for no: 3

old 2: no emp08.eid%type:=&no;

new 2: no emp08.eid%type:=3;

PL/SQL procedure successfully completed.

SQL> select \* from emp08;

EID ENAME SAL DOJ DEPTNO

---------- ---------- ---------- ---------- ----------

1 x 20000 2010 1

2 y 30000 2015 2

3 z 41000 2007 1

4 w 35000 2011 3

5 a 17500 2013 2

6 b 34000 2016 4

6 rows selected.

SQL> /

Enter value for no: 7

old 2: no emp08.eid%type:=&no;

new 2: no emp08.eid%type:=7;

declare

\*

ERROR at line 1:

ORA-01403: no data found

ORA-06512: at line 5

**Loop statements:**

6. pl/sql program to check whether given no. Is Amstrong or not:

**Program:**

SQL> get Z:\madhavdbms\final\prog6.sql

1 declare

2 s number:=0;

3 n number:=&n;

4 t number:=n;

5 r number:=n;

6 begin

7 while (n>0)

8 loop

9 r:=n mod 10;

10 s:=s+r\*r\*r;

11 n:=trunc(n/10);

12 end loop;

13 if (t=s) then

14 dbms\_output.put\_line(t||'is a armstrong');

15 else

16 dbms\_output.put\_line(t||'is not armstrong')

17 end if;

18\* end;

SQL> /

**output:**

Enter value for n: 153

old 3: n number:=&n;

new 3: n number:=153;

153is a armstrong

PL/SQL procedure successfully completed.

SQL> /

Enter value for n: 142

old 3: n number:=&n;

new 3: n number:=142;

142is not armstrong

PL/SQL procedure successfully completed.

7.PL/SQL program for palindrome of a number:

SQL> get Z:\madhavdbms\final\prog7.sql

1 declare

2 s number:=0;

3 n number:=&n;

4 t number:=n;

5 r number;

6 begin

7 while (n>0)

8 loop

9 r:=n mod 10;

10 s:=s\*10+r;

11 n:=trunc(n/10);

12 end loop;

13 if (t=s) then

14 dbms\_output.put\_line(t||' is palindrome');

15 else

16 dbms\_output.put\_line(t||' is not palindrome');

17 end if;

18\* end;

**Output:**

SQL> /

Enter value for n: 6

old 3: n number:=&n;

new 3: n number:=6;

6 is palindrome

PL/SQL procedure successfully completed.

SQL> /

Enter value for n: 123

old 3: n number:=&n;

new 3: n number:=123;

123 is not palindrome

PL/SQL procedure successfully completed.

SQL> /

Enter value for n: 121

old 3: n number:=&n;

new 3: n number:=121;

121 is palindrome

PL/SQL procedure successfully completed.

8.program to write for sum of digts:

SQL> edit prog8.sql

SQL> get Z:\madhavdbms\final\prog8.sql

1 declare

2 n number:=&n;

3 s number:=0;

4 r number;

5 begin

6 while n>1 loop

7 r:=trunc(mod(n,10));

8 s:=s+r;

9 n:=n/10;

10 end loop;

11 dbms\_output.put\_line(s);

12\* end ;

**Output:**

SQL> /

Enter value for n: 123

old 2: n number:=&n;

new 2: n number:=123;

6

PL/SQL procedure successfully completed.

SQL> /

Enter value for n: 321

old 2: n number:=&n;

new 2: n number:=321;

6

PL/SQL procedure successfully completed.

SQL>

9.pl/sql program for reverse number:

SQL> edit prog9.sql

SQL> get Z:\madhavdbms\final\prog9.sql

1 declare

2 n number:=&n;

3 r number;

4 s number:=0;

5 begin

6 while n> 1 loop

7 r:=trunc(mod(n,10));

8 s:=(s\*10)+r;

9 n:=n/10;

10 end loop;

11 dbms\_output.put\_line(s);

12\* end;

SQL> /

**Output:**

Enter value for n: 123

old 2: n number:=&n;

new 2: n number:=123;

321

PL/SQL procedure successfully completed.

SQL> /

Enter value for n: 9991

old 2: n number:=&n;

new 2: n number:=9991;

1999

PL/SQL procedure successfully completed.

10. pl/sql program to print fibanocci series:

SQL> edit prog10.sql

SQL> get Z:\madhavdbms\final\prog10.sql

1 declare

2 a number:=0;

3 b number:=1;

4 c number:=1;

5 n number:=&n;

6 begin

7 dbms\_output.put\_line(a);

8 dbms\_output.put\_line(b);

9 while (c<n) loop

10 c:=a+b;

11 dbms\_output.put\_line(c);

12 a:=b;

13 b:=c;

14 end loop;

15\* end;

**output:**

SQL> /

Enter value for n: 3

old 5: n number:=&n;

new 5: n number:=3;

0

1

1

2

3

PL/SQL procedure successfully completed.

11. pl/sql program to print prime numbers in given range:

SQL> edit prog11.sql

SQL> get Z:\madhavdbms\final\prog11.sql

1 declare

2 i number;

3 j number;

4 n number:=&n;

5 cnt number;

6 begin

7 for i in 2..n loop

8 cnt:=0;

9 for j in 1..i loop

10 if i mod j=0 then

11 cnt:=cnt+1;

12 end if;

13 end loop;

14 if cnt=2 then

15 dbms\_output.put\_line(i);

16 end if;

17 end loop;

18\* end;

**output:**

Enter value for n: 10

old 4: n number:=&n;

new 4: n number:=10;

2

3

5

7

PL/SQL procedure successfully completed.

SQL> /

Enter value for n: 9

old 4: n number:=&n;

new 4: n number:=9;

2

3

5

7

PL/SQL procedure successfully completed.

12.pl/sql program to print multiplication table:

**Program:**

SQL> edit prog12.sql

SQL> get Z:\madhavdbms\final\prog12.sql

1 declare

2 n number:=&n;

3 i number;

4 begin

5 for i in 1..12 loop

6 dbms\_output.put\_line(n||'X'||i||'='||n\*i);

7 end loop;

8\* end;

**Output:**

SQL> /

Enter value for n: 4

old 2: n number:=&n;

new 2: n number:=4;

4X1=4

4X2=8

4X3=12

4X4=16

4X5=20

4X6=24

4X7=28

4X8=32

4X9=36

4X10=40

4X11=44

4X12=48

PL/SQL procedure successfully completed.

12. pl/sql program for PRINT NUMBERS BY USING LOOP REVERSE AND LOOP

SQL> DECLARE

2 loop\_start Integer := 1;

3 BEGIN

4 FOR i IN REVERSE loop\_start..5 LOOP

5 DBMS\_OUTPUT.PUT\_LINE('Loop counter is ' || i);

6 END LOOP;

7 END;

8 /

Output:

Loop counter is 5

Loop counter is 4

Loop counter is 3

Loop counter is 2

Loop counter is 1

PL/SQL procedure successfully completed.

SQL>

DECLARE

a NUMBER=1;

BEGIN

dbms\_output.put\_line('Program started.');

LOOP

dbms\_output.put\_line(a);

a:=a+l;

EXIT WHEN a>5;

END LOOP;

dbms\_output.put\_lme('Program completed');

END:

**Functions:**

13. pl/sql program for factorial of a number using functions:

**Function:**

create or replace function factfun7(n number) return number is

i number:=1;

s number:=1;

begin

for i in 1..n loop

s:=s\*i;

end loop;

return s;

end;

**output:**

SQL> @ Z:\mca7r\dbms\factfun7.sql;

10 /

Function created.

**Main program**:

declare

n number;

fact number;

begin

fact:=factfun7(n);

dbms\_output.put\_line('factorial is:'||fact);

end;

**output:**

SQL> @ Z:\mca7r\dbms\factorial7.sql;

8 /

Enter value for n: 5

old 2: n number:=&n;

new 2: n number:=5;

factorial is:120

PL/SQL procedure successfully completed.

**Procedures:**

14.pl/sql program for swaping two numbers using procedures:

**Procedure:**

SQL> edit prog14.sql

SQL> get Z:\madhavdbms\final\prog14.sql

1 create or replace procedure swap7(a in out number,b in out number) is

2 c number;

3 begin

4 dbms\_output.put\_line('numbers before swapping');

5 dbms\_output.put\_line('a='||a||'b='||b);

6 c:=a;

7 a:=b;

8 b:=c;

9 dbms\_output.put\_line('numbers after swapping');

10 dbms\_output.put\_line('a='||a||'b='||b);

11\* end;

SQL> /

Procedure created.

**Main program:**

SQL> edit prog14a.sql

SQL> get Z:\madhavdbms\final\prog14a.sql

1 declare

2 x number:=&x;

3 y number:=&y;

4 begin

5 swap7(x,y);

6\* end;

**Output:**

SQL> /

Enter value for x: 2

old 2: x number:=&x;

new 2: x number:=2;

Enter value for y: 3

old 3: y number:=&y;

new 3: y number:=3;

numbers before swapping

a=2b=3

numbers after swapping

a=3b=2

PL/SQL procedure successfully completed.

15.pl/sql program to display salary of an employee using exceptions:

**Program:**

declare

no emp08.eid%type:=&no;

s emp08.sal%type;

begin

select sal into s from emp08 where eid=no;

dbms\_output.put\_line('salary is:'||s);

exception

when no\_data\_found then

dbms\_output.put\_line('Eno does not exist');

end;

**Output:**

SQL> /

Enter value for no: 3

old 2: no emp08.eid%type:=&no;

new 2: no emp08.eid%type:=3;

salary is:40000

PL/SQL procedure successfully completed.

16.pl/sql program for user defined exception:

**Program:**

SQL> edit 16.sql

SQL> get Z:\madhavdbms\final\prog16.sql

1 declare

2 x number:=&x;

3 y number:=&y;

4 a exception;

5 b exception;

6 begin

7 if x>y then

8 raise a;

9 else

10 raise b;

11 end if;

12 exception

13 when a then

14 dbms\_output.put\_line('x is big');

15 when b then

16 dbms\_output.put\_line('y is big');

17\* end;

18 /

**Output:**

Enter value for x: 22

old 2: x number:=&x;

new 2: x number:=22;

Enter value for y: 12

old 3: y number:=&y;

new 3: y number:=12;

x is big

PL/SQL procedure successfully completed..

**17. write a pl/sql program to implement implicit cursors**

SQL> declare

no emp.eno%type;

s emp.salary%type;

dn emp.deptno%type:=&dept\_no;

cursor c is select eno,salary,deptno fr

begin

open c;

loop

fetch c into no,s,dn;

dbms\_output.put\_line('employee number:'

dbms\_output.put\_line('employee salary:'

dbms\_outpur.put\_line('employee dno:'||d

end loop;

close c;

end;

/

OUTPUT:

Enter value for dept\_no: 21

old 4: dn emp.deptno%type:=&dept\_no;

new 4: dn emp.deptno%type:=21;

employee number:7369

employee salary:800

employee dno:20

employee number:7566

employee salary:2925

employee dno:20

employee number:7879

employee salary:3000

employee dno:20

employee number:7889

employee salary:1100

employee dno:20

employee number:7902

employee salary:3000

employee dno:20

PL/SQL procedure successfully completed.

**18. write a pl/sql program to implement explicit cursors**

SQL> declare

no emp.eno%type;

s emp.salary%type;

dgn emp.ejob%type;

cursor c is select eno,salary,ejob from emp;

begin

open c;

loop

fetch c into no,s,dgn;

exit when c%notfound;

if dgn=’manager’ then

s:=s+1500;

else if dgn=’clerk’ then

s:=s+1000;

else if dgn=’asm’ then

s:=s+1200;

else if dgn=’typist’ then

s:=s+500;

else if dgn=’md’ then

s:=s+3000;

end if;

end if;

end if;

end if;

end if;

update emp set salary=s where ejob=dgn;

end loop;

close c;

end;

/

**TRIGGERS:**

19.Program to make a table for read only using triggers.

SQL> edit 18.sql

SQL> declare

2 a number:=&a;

3 b number:=&b;

4 begin

5 dbms\_output.put\_line('addition is:'||(a+b));

6 dbms\_output.put\_line('substraction is:'||(a-b));

7 dbms\_output.put\_line('multiplication is:'||(a\*b));

8 dbms\_output.put\_line('division is:'||(a/b));

9 end;

10 /

**Output:**

Enter value for a: 2

old 2: a number:=&a;

new 2: a number:=2;

Enter value for b: 23

old 3: b number:=&b;

new 3: b number:=23;

addition is:25

substraction is:-21

multiplication is:46

division is:.0869565217391304347826086956521739130435

PL/SQL procedure successfully completed.

SQL>.