DBMS LAB

MANUAL

NAME - DEEPAK KANAUJIA

ROLL NO. – 205119029

**QUESTION 1**

**1.1-1.3**

***ans*** :- create table EMP(enmpno integer PRIMARY KEY,enamevarchar(20) NOT NULL,jobvarchar(10) Not NULL,mgrinteger,deptnointeger,sal integer);

desc EMP;

alter table EMP add comission integer(8);

desc EMP;

alter table EMP modify job varchar(8);

desc EMP;

**1.4**

ans :- create table EMP(

enmpno integer(6) PRIMARY KEY,

enamevarchar(20) NOT NULL,

jobvarchar(10) Not NULL,

mgr integer(4),

deptno integer(3),

sal integer(7)

);

desc EMP;

alter table EMP add comission integer(8);

desc EMP;

alter table EMP modify job varchar(8);

desc EMP;

create table dept(deptno integer PRIMARY KEY,dnamevarchar(10),locvarchar(10));

descdept;

alter table EMP add foreign key(deptno) references dept(deptno);

descdept;

desc EMP;

alter table EMP add check(empno> 100);

alter table EMP alter sal set default 5000;

selectsal from EMP;

alter table EMP add dob varchar(10);

**QUESTION 2**

**2.1**

insert into dept values(10,'management','main block');

insert into dept values(20,'development','manufacturing');

insert into dept values(30,'maintainence','main block');

**2.2**

insert into emp values(29,'deepak','clerk','7566',20,800,'11-dec-20',0);

insert into emp values(36,'harsh','worker','7586',10,1000,'12-nov-98',10);

insert into emp values(16,'ankur','coder','7644',18,1200,'26-sep-98',10);

insert into emp values(51,'lakshay','asst prof','7294',28,1500,'28-jan-98',10);

insert into emp values(57,'mrinal','developer','6863',30,2500,'02-sep-98',10);

insert into emp values(64,'pankaj','developer','9893',30,2500,'31-aug-98',10);

insert into emp values(103,'sunil','psycho','9633',30,1500,'28-feb-99',10);

insert into emp values(109,'vishesh','teacher','4433',30,2500,'28-mar-96',10);

insert into emp values(49,'kunal','master','2489',30,500,'18-jul-97',10);

insert into emp values(73,'rahul','maid','8889',30,500,'16-jun-97',10);

**2.3**

updateemp set comm=1000 where job='developer';

**2.4**

create table employee as select \* from emp;

**2.5**

delete from employee where job='clerk';

**2.6**

delete from employee where empno=51;

**2.7**

select \* from employee order by sal;

**2.8**

select \* from employee order by saldesc;

**2.9**

select \* from employee where deptno=30;

**2.10**

select distinct deptno from employee;

**2.11**

select \* from employee order by ename;

**2.12**

create table developer as select \* from emp where job='developer';

**2.13**

select \* from emp where comm=null;

**2.14**

selectename,dname from emp,dept where emp.deptno=dept.deptno;

**QUESTION 3**

**3.1**

**select \* from employee where deptno in(10,30);**

**3.2**

**select \* from employee where empname like ‘s%’;**

**3.3**

**select \* from employee where empname NOT like ‘s%’;**

**QUESTION 4**

**4.1**

select \* from emp,dept where emp.deptno=dept.deptno and(dname='maintanence' or dname='developmt');

**4.2**

selectename,sal from emp where sal>(select min(sal) from emp where job like 'm%');

**4.3**

selectename from emp where job = (select job from emp where ename='jones');

**4.4**

select \* from emp where sal>(select max(sal) from emp where deptno=30);

**4.5**

selectename from emp where job = (select job from emp where ename='jones') and sal>=(select sal from emp where ename='ford');

**4.6**

selectename,job from emp where deptno=20;

**4.7**

select \* from emp where sal> all(select avg(sal) from emp group by deptno);

**4.8**

selectename,job,dname from emp,dept where emp.deptno=dept.deptno;

**4.9**

select \* from emp where job=any(select distinct job from emp,dept where emp.deptno=dept.deptno and loc='main block');

**4.10**

select \* from emp where deptno=10 and job=any(select job from emp,dept where emp.deptno=dept.deptno and dname='developmt');

**4.11**

select \* from emp where job=(select job from emp where ename='ford') and sal=(select sal from emp where ename='ford');

**4.12**

selectdname from dept where deptno=any(select deptno from (select count(job) as no,deptno from emp where job='saleman' group by deptno) where no>1) ;

**4.13**

select \* from emp where deptno=20 and job=any(select job from emp where deptno=30);

**4.14**

selectename from emp where sal>any(select max(sal) from emp where deptno in(20,30));

**4.15**

select max(sal) from emp having max(sal)>9000 group by deptno;

**4.16**

select max(sal) from emp having (min(sal)>1000 and min(sal)<5000) group by deptno;

create table AccDept(deptnointeger,dname varchar2(20),dcity varchar2(20));

insert into AccDept values(10,'management','main block');

insert into AccDept values(20,'development','manufacturing');

insert into AccDept values(30,'maintainance','unit main block');

**4.17**

selectdept.dname from dept join AccDept on dept.deptno=AccDept.deptno;

**4.18**

selectename from emp join AccDept on emp.deptno=AccDept.deptno where emp.deptno!=AccDept.deptno;

**4.19**

selectename,dname from emp left join dept on emp.deptno=dept.deptno;

**4.20**

selectename,dname from emp right join dept on emp.deptno=dept.deptno;

**4.21**

selectename,dname from emp full join dept on emp.deptno=dept.deptno;

**4.22**

select e1.ename as empname,e2.ename as mgrname from emp e1, emp e2 where e1.mgr=e2.enmpno;

**QUESTION 5**

**5.1**

selectdeptno from dept union select deptno from accdept;

**5.2**

selectdeptno from dept union all select deptno from accdept;

**5.3**

selectdeptno from dept intersect select deptno from accdept;

**5.4**

selectdeptno from dept minus select deptno from accdept;

**5.5**

create view managers as select ename from emp where job='manager';

select \* from managers;

**5.6**

create view general as select enmpno,ename,emp.deptno,dname from emp,dept where emp.deptno=dept.deptno;

select \* from general;

**5.7**

create view allv as select enmpno,ename,emp.deptno,dname from emp,dept where emp.deptno=dept.deptno and job!='CEO' and job!='HOD';

select \* from allv;

**5.9**

select \* from general;

**5.10**

drop view managers;

**QUESTION 6**

**6.1**

declare

a number(10);

b number(10);

begin

a:=&a;

b:=&b;

dbms\_output.put\_line('THE PREV VALUES OF A AND B WERE');

dbms\_output.put\_line(a);

dbms\_output.put\_line(b);

a:=a+b;

b:=a-b;

a:=a-b;

dbms\_output.put\_line('THE VALUES OF A AND B ARE');

dbms\_output.put\_line(a);

dbms\_output.put\_line(b);

end;

**6.2**

declare

a number(10);

b number(10);

c number(10);

begin

dbms\_output.put\_line('THE PREV VALUES OF A AND B WERE');

dbms\_output.put\_line(a);

dbms\_output.put\_line(b);

a:=&a;

b:=&b;

c:=a;

a:=b;

b:=c;

dbms\_output.put\_line('THE VALUES OF A AND B ARE');

dbms\_output.put\_line(a);

dbms\_output.put\_line(b);

end;

6.3

declare

a number; b number; begin a:=&a; b:=&b;

if a=b then

dbms\_output.put\_line('BOTH ARE EQUAL');

elsif a>b then

dbms\_output.put\_line('A IS GREATER');

else

dbms\_output.put\_line('B IS GREATER');

end if;

end;

6.4

declare

java number(10);

dbms number(10);

co number(10);

se number(10);

es number(10);

ppl number(10);

total number(10);

avgs number(10);

per number(10);

begin

dbms\_output.put\_line('ENTER THE MARKS');

java:=&java;

dbms:=&dbms;

co:=&co;

se:=&se;

es:=&es;

ppl:=&ppl;

total:=(java+dbms+co+se+es+ppl);

per:=(total/600)\*100;

if java<40 or dbms<40 or co<40 or se<40 or es<40 or ppl<40 then dbms\_output.put\_line('FAIL');

if per>75 then dbms\_output.put\_line('GRADE A');

elsif per>65 and per<75 then dbms\_output.put\_line('GRADE B'); elsif per>55 and per<65 then dbms\_output.put\_line('GRADE C'); else

dbms\_output.put\_line('INVALID INPUT');

end if;

dbms\_output.put\_line('PERCENTAGE IS '||per);

dbms\_output.put\_line('TOTAL IS '||total);

end;

**6.5**

declare

a number;

d number:=0;

sum1 number:=0;

begin

a:=&a; while a>0 loop

d:=mod(a,10);

sum1:=sum1+d;

a:=trunc(a/10);

end loop;

dbms\_output.put\_line('sum is'|| sum1);

end;

**6.6**

declare

a number;

rev number;

d number;

begin

a:=&a; rev:=0;

while a>0 loop

d:=mod(a,10);

rev:=(rev\*10)+d;

a:=trunc(a/10);

end loop;

dbms\_output.put\_line('no is'|| rev);

end;

**6.7**

declare

a number;

c number:=0; i number; begin

a:=&a;

fori in 1..a

loop

if mod(a,i)=0 then c:=c+1;

end if;

end loop;

if c=2 then

dbms\_output.put\_line(a ||'is a prime number');

else

dbms\_output.put\_line(a ||'is not a prime number');

end if;

end;

**6.8**

declare

n number;

f number:=1;

begin n:=&n;

fori in 1..n loop

f:=f\*i;

end loop;

dbms\_output.put\_line('the factorial is'|| f);

end;

**6.9**

**TABLE NAME:AREAS**

**RADIUS AREA**

SQL> create table areas(radius number(10),area number(6,2)); Table created.

--PROGRAM

declare

pi constant number(4,2):=3.14;

radius number(5):=3;

area number(6,2);

begin

while radius<7 loop area:=pi\*power(radius,2);

insert into areas values(radius,area);

radius:=radius+1;

end loop;

end;

**6.10**

declare

mano number(5);

mcb number(6,2);

minibal constant number(7,2):=1000.00;

fine number(6,2):=100.00;

beginmano:=&mano;

selectcur\_bal into mcb from acct where acctno=mano;

ifmcb<minibal then

update acct set cur\_bal=cur\_bal-fine where acctno=mano;

end if;

end;

**7.1**

create or replace procedure salary(deptid number) as

begin

updateemp set sal=sal+1000 where sal>5000 AND deptno=deptid;

end;

**7.2**

create or replace procedure salary1(empid number) as

begin

updateemp set sal=sal+sal\*(0.1) where empno=empid; end;

**7.3**

create or replace procedure get\_sal(dept number) as

begin

for s in (select \* from emp where deptno = dept)

loop

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

end loop;

end;

**7.4**

create or replace procedure get\_nature(dept number) as

begin

for s in (select \* from emp where deptno = dept)

loop

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

end loop;

end;

**7.5**

create or replace procedure dep\_name(deptid number) as

begin

selectdept.dname from dept,emp where emp.deptno=dept.deptno;

end;

create or replace trigger trig1 before insert on dept

for each row declare a number;

begin

if(:new.deptno is null)

then

raise\_application\_error(-20001,'error:: deptno cannot be null');

else

select count(\*) into a from deptwhere deptno=:new.deptno;

if(a=1) then

raise\_application\_error(-20002,'error:: cannot have duplicate deptno ');

end if;

end if;

end;

create [or replace] trigger trig2 after delete on deptfor each row

begin

delete fromempwhere emp.deptno=:new.deptno;

end;

create trigger trig3 after delete onempfor each row

begin

insert into log(val1, val2, ...) values(old.val1, old.val2, ...);

end;