**DBMS LAB**

**Assignment No. 4**

Q1: Prepare a database for an e-commerce company containing following entities (Also draw an ER diagram):-

Supplier, Customer, Product, Category, Order and Payment

create table customer2(cname varchar2(15),cid number(4) primary key not null,cstate varchar2(9));

Insert into customer2 values('vaishali',1000,'delhi');

insert into customer2 values('mohit',1001,'bihar');

Insert into customer2 values('sumit',1002,'faridabad');

Insert into customer2 values('mariam',1003,'delhi');

Insert into customer2 values('ifra',1004,'noida');

Insert into customer2 values('sreyasi',1005,'noida');

create table products(pname varchar2(10),product\_id number(4) primary key not null);

Insert into products values('knife',2000);

Insert into products values('jar',2001);

Insert into products values('pan',2002);

Insert into products values('bed',2003);

Insert into products values('bedsheet',2004);

Insert into products values('pillow',2005);

Insert into products values('chair',2006);

create table supplier1(product\_id number(4),sid number(4) primary key not null,sname varchar2(10),foreign key(product\_id) references products(product\_id));

Insert into supplier1 values(2000,3000,'astaha');

Insert into supplier1 values(2001,3004,'tanya');

Insert into supplier1 values(2004,3002,'mariyum');

Insert into supplier1 values(2005,3003,'sahil');

Insert into supplier1 values(2003,3001,'ayush');

Insert into supplier1 values(2002,3005,'aima');

Create table order2(oamount number(7,2),odate date,cid number(4),order\_id number(4) primary key not null,product\_id number(4),sid number(4),qty number(2),delplace varchar2(10),foreign key(product\_id) references products(product\_id),foreign key(sid) references supplier1(sid),foreign key(cid) references customer2(cid));

Insert into order2 values(23000,'13-mar-18',1000,4000,2005,3003,3,'delhi');

Insert into order2 values(16422,'3-jun-18',1001,4005,2004,3001,1,'jaipur');

Insert into order2 values(19876,'29-aug-18',1002,4004,2003,3005,5,'ladakh');

Insert into order2 values(12028,'13-mar-18',1003,4001,2002,3004,2,'goa');

Insert into order2 values(11090,'12-oct-17',1004,4002,2001,3002,6,'shimla');

Insert into order2 values(14090,'6-nov-19',1005,4006,2003,3002,2,'manali');

Insert into order2 values(19590,'25-feb-18',1003,4007,2005,3001,1,'lucknow');

Insert into order2 values(35630,'11-sep-17',1002,4003,2001,3004,2,'banglore');

Insert into order2 values(45630,'11-sep-17',1002,4009,2000,3002,1,'banglore');

Create table payment1(pmode varchar2(10),order\_id number(4),foreign key(order\_id) references order2(order\_id));

Insert into payment1 values('cash',4000);

Insert into payment1 values('cheque',4003);

Insert into payment1 values('online',4002);

Insert into payment1 values('cash',4004);

Insert into payment1 values('online',4003);

Insert into payment1 values('dd',4001);

Insert into payment1 values('cheque',4005);

Insert into payment1 values('online',4007);

Insert into payment1 values('online',4006);

Create table category1(product\_id number(4),cat\_name varchar2(10),foreign key(product\_id) references products(product\_id));

Insert into category1 values(2000,'kitchen');

Insert into category1 values(2001,'kitchen');

Insert into category1 values(2002,'kitchen');

Insert into category1 values(2003,'room');

Insert into category1 values(2004,'room');

Insert into category1 values(2005,'room');

And, Answer the queries that follow:-

1. Give a list of all customers whose name begins with ‘P’?

**Select \* from customer2 where cname like ‘p%’;**

1. Display the names of suppliers who provide kitchen articles?

**Select s.sname,s.product\_id from supplier1 s,category1 c where s.product\_id=c.product\_id and c.cat\_name='kitchen';**

c) What was the total sale of the company in the month of March 2018?

**Select sum(oamount) from order2 where odate like '%MAR-18';**

d) Find out each customer’s minimum and maximum order?

**Select min(oamount),max(oamount),cid from order2 group by cid;**

e) What were the top selling products along with their categories during the sale of JUNE-AUGUST 2018?

**SQL> Select product\_id,(select cat\_name from category1 c where c.product\_id=o.pr**

**oduct\_id) "cat\_name",qty from order2 o where odate between '1-jun-2018' and '31-**

**aug-18' group by product\_id,qty having sum(qty) > (select avg(qty) from order2);**

f) Display a list of customers who bought for more than Rs. 10,000 in a month?

**Select cid,sum(oamount) from order2 group by cid having sum(oamount)>10000;**

g) List all orders with their order details (name of buyer, mode of payment, products bought).

**Select o.cid,p.pmode,o.product\_id,o.order\_id from order2 o,payment1 p where o.order\_id=p.order\_id;**

**Select (select cname from customer2 c where c.cid=o.cid) "name of buyer", p**

**.pmode,(select p.pname from products p where p.product\_id=o.product\_id) "product**

**s bought",o.order\_id from order2 o,payment1 p where o.order\_id=p.order\_id;**

h) List all customers according to their state/union territory of their delivery address.

**Select (select cname from customer2 c where c.cid=o.cid) “customer\_name”,o.delplace from order2 o ;**

i) Create a view containing the names of all the products and their categories.

**SQL> Create view procat1 as select p.pname ,c.cat\_name from products p,category1**

**c where p.product\_id=c.product\_id;**

// j) Find out each customers highest and least chosen mode of payment.

Select cnum from order1 where amount IN(Select max(amount) from order1 group by cnum) ;

Select o.cnum,p.pmode from order1 o,payment p where o.amount IN(Select max(amount) from order1 group by cnum) group by cnum;

Select max(pmode),min(pmode),cname from(select cid,order\_id,(select pmode from payment1 p where o.order\_id=p.order\_id) as pmode,(select cname from customer2 c where c.cid=o.cid)as cname from order2 o) group by cid,cname;

k) List the name of all suppliers who sell more than two categories of products.

**Select sname from (select sid,(select sname from supplier1 s where s.sid=o.sid) as sname,product\_id,(select cat\_name from category1 c where c.product\_id=o.product\_id) as cat\_name from order2 o) group by sname having count(distinct(cat\_name))>2;**

l) Prepare a list of least selling products for the last year.

**Select (select pname from products p where p.product\_id=o.product\_id) “least\_sellingp” from order2 o where odate like ‘%18’ group by product\_id having sum(qty)>(select avg(qty) from order2);**

m) Select the total amount in orders for each customer for whom this total is greater than the amount of the largest order in the table.

**Select sum(oamount),cid from order2 group by cid having sum(oamount) >(Select max(oamount) from order2);**

n) Find out which customers produce largest and smallest orders on each date.

Select max(oamount),min(oamount),odate,cid from order2 group by cid,odate;

Q2: Write a PL/SQL code to check whether a number is prime or not.

Declare

Num number;

Flag boolean:=true;

Begin

num:=&num;

for I in 2….num/2

loop

if(mod(num,i)=0) then

flag:=false;

exit;

end if;

end loop;

if(flag) then

dbms\_output.put\_line('numbr is prime' );

else

dbms\_output.put\_line('numbr is not prime' );

end if;

end;

/

Q3: Write a PL/SQL code to check whether a number is palindrome or not.

Declare

num varchar(4);

inverted\_num varchar(4);

Str\_length number(4);

Begin

num:=&num;

Str\_length:=length(num);

For I in reverse 1….str\_length

Loop

Inverted\_num:=inverted\_num || substr(num,I,1);

End loop;

dbms\_output.put\_line('inverted number is' || inverted\_num);

If(num=inverted\_num) then

Dbms\_output.put\_line('number is palindrome');

Else

Dbms\_output.put\_line('number is not palindrome');

End if;

End;

/

Q4: Write a PL/SQL code to compute factorial of a given number.

Declare

num number;

fact number:=1;

Begin

Num:=&num;

For I in 1….num

loop

fact:=fact\*I;

End loop;

Dbms\_output.put\_line('factorial of a given numbr is ' || fact);

End;

/

Q5: Write a PL/SQL code to print Fibonacci series.

Declare

num Number;

a Number;

b Number;

c Number;

Begin

Dbms\_output.put\_line('enter the value of n upto which you want to

print fibonaaci series');

num:=&num;

a:=0;

b:=1;

dbms\_output.put\_line(a);

dbms\_output.put\_line(b);

For I in 1….num-2

Loop

C:=a+b;

Dbms\_output.put\_line(c);

a:=b;

b:=c;

end loop;

end;

/

Q6: Write a PL/SQL code to display sum of first ten natural numbers.

Declare

sums Number:=0;

Begin

For I in 1..10

loop

sums:=sums+I;

End loop;

Dbms\_output.put\_line('sum of first 10 natural number is ' || sums);

End;

/

Q7: Write a PL/SQL code to compute area and perimeter of a circle.

Declare

radius number;

area number;

perimeter number;

pi constant Number:=3.14;

Begin

radius:=&radius;

Area:=pi\*radius\*radius;

Dbms\_output.put\_line('Area of circle '||area);

Perimeter:=2\*pi\*radius;

Dbms\_output.put\_line('perimeter of circle '||perimeter);

End;

/

Q8: Write a PL/SQL code to find the greatest among three numbers.

Declare

a Number;

b Number;

c Number;

maximum number;

Begin

a:=&a;

b:=&b;

c:=&c;

If(a>b) then

If(a>c)then

Maximum:=a;

Else

Maximum:=c;

End if;

Elsif(b>c) then

Maximum:=b;

Else

Maximum:=c;

End if;

Dbms\_output.put\_line('maximum of three numbers is ' || maximum);

End;

/

Q9: Write a PL/SQL code to display whether a number is even or odd.

Declare

Num number;

Begin

num:=&num;

If(mod(num,2)=0) then

dbms\_output.put\_line('numbr is even' );

Else

Dbms\_output.put\_line('numbr is odd' );

End if;

end;

/

Q10: Write a PL-SQL script to compare three given numbers and display them in ascending order.

Declare

a Number;

b Number;

c Number;

maximum number;

Begin

a:=&a;

b:=&b;

c:=&c;

If(a>b) then

If(a>c)then

maximum:=a;

Else

maximum:=c;

End if;

Elsif(b>c) then

maximum:=b;

Else

maximum:=c;

End if;

If(a=maximum) then

If(b>c) then

Dbms\_output.put\_line('numbers is ' ||maximum ||' '||b || ' '||c );

Else

Dbms\_output.put\_line('numbers is ' ||maximum||' '|| c || ' '||b );

end if;

Elsif(b=maximum) then

If(a>c) then

Dbms\_output.put\_line('numbers is ' ||maximum||' '|| a || ' ' ||c);

Else

Dbms\_output.put\_line('numbers is '||maximum||' '|| c ||' '|| a);

End if;

Else

If(a>b) then

Dbms\_output.put\_line('numbers is ' || maximum||' '||a ||' '||b);

Else

Dbms\_output.put\_line('numbers is ' ||maximum||' '|| b||' '|| a);

End if;

end if;

End;

/

Q11: Create a table ‘Emp’ with attributes ‘ename’,’ecity’,’salary’,’enumber’,’eaddress’,’depttname’.

Create another table ‘Company’ with attributes ‘cname’, ccity’,’empnumber’ in the database ‘Employee’

SQL> create table emp2(enumber number(3) primary key not null,ename varchar2(10)

not null,ecity varchar(15),salary number(8),eaddress varchar2(10),depttname varc

har2(8));

SQL> create table company(cname varchar2(10) primary key not null,ccity varchar2

(6),empnumber number(3) , foreign key(empnumber) references emp2(enumber));

insert into emp2 values('100','john','noida',5000,'delhi','it');

insert into emp2 values('200','mohit','delhi',20000,'delhi','hr');

insert into emp2 values('300','rahul','faridabad',8000,'noida','technical');

insert into emp2 values('400','sreyasi','noida',25000,'faridabad','it');

insert into company values('mahindra','delhi',100);

insert into company values('applift','noida',200);

insert into company values('naggaro','delhi',300);

insert into company values('gigstart','bihar',200);

Execute the following queries on above tables:-

* Create a view having ename and ecity.

create view emp2\_view as select ename,ecity from emp2;

select \* from emp2\_view;

* In the above view change the ecity to ‘Delhi’ where ename is ‘John’.

update emp2\_view set ecity='delhi' where ename='john';

select \* from emp2\_view;

* Create a view having attributes from both the tables.

SQL> create view emp2\_view3 as select \* from emp2 e,company c where e.enumber=c.

empnumber;

select \* from emp2\_view3;

* Update the above view and increase the salary of all employees of IT department by Rs.1000.

update emp2\_view3 set salary=salary+10000 where depttname='it';

Now solve the following queries using PL/SQL:-

* Calculate the average salary from table ‘Emp’ and print increase the salary if the average salary is less than 10,000.

declare

sal number;

begin

select avg(salary) into sal from emp2;

if(sal<10000) then

dbms\_output.put\_line('increase the salary');

else

dbms\_output.put\_line('no need to increase salary');

end if;

end;

/

* Print the deptno from the employee table using the case statement if the deptname is ‘Technical’ then deptno is 1, if the deptname is ‘HR’ then the deptno is 2 else deptno is 3.

declare

deptno number(1);

dname varchar(12);

begin

select depttname into dname from emp2 where enumber=100;

case(dname)

when 'tecnical' then deptno:=1; dbms\_output.put\_line(dname||' '||deptno);

when 'hr' then deptno:=2; dbms\_output.put\_line(dname||' '||deptno);

else deptno:=3; dbms\_output.put\_line(dname||' '||deptno);

end case;

end;

/