ASSIGNMENT – 8

1.A   Problem Statement:

Query: Create a table whose structure will be as follows:

Table Name: Prime_Entry

Column Name	Data Type	Attributes
Num_id	Number(3)	Primary Key
Prime_num	Number(3)	Not Null

```
8)1.a

create table Prime_entry(num_id number(3) primary key,prime_num number(3) not NULL);
create sequence s
start with 1
increment by 1
/
```

Output:

```
SQL> create table Prime_entry(num_id number(3) primary key,prime_num number(3) not NULL);

Table created.

SQL> @D:\sidd57\ass_8_1.txt

Sequence created.

SQL>
```

**1.b Problem Statement: Write a PL/SQL block of code that will take a number from user and test
whether the numberis prime or not. If the number is prime, then enter into above
table by generating NUMID automatically.**

Query:

```
SET SERVEROUTPUT ON
DECLARE
    num NUMBER;
    i NUMBER;
    n NUMBER;
    flag NUMBER;
    g NUMBER;
BEGIN
    num := &num;
    n := TRUNC(num / 2);

    FOR i IN 2..n LOOP
        IF MOD(num, i) = 0 THEN
            flag := 1;
            EXIT;
        ELSE
            flag := 0;
        END IF;
    END LOOP;

    IF flag = 1 THEN
        DBMS_OUTPUT.PUT_LINE(num || ' is not prime!');
    ELSE
        SELECT seq.nextval INTO g FROM dual;
        INSERT INTO Prime_entry VALUES (g, num);
    END IF;
END;
/
```

Output:

```
Enter value for num: 7
old     8:          num := &num;
new     8:          num := 7;

PL/SQL procedure successfully completed.

SQL> select * from prime_entry;

    NUM_ID   PRIME_NUM
---------- ----------
         1           7

SQL>
```

1.C Problem Statement:

Now add a checking for same prime number entry. It will show - 'Number already exists in database' for same prime number entry. Write a function to test whether given number exist or not.

Query:

```
SET SERVEROUTPUT ON
create or replace function prime_test(id number) return number IS
num number(20);
begin
select num_id into num from Prime_entry where prime_num = id;
return 1;
exception
when no_data_found then
return 0;
end;
/
SET SERVEROUTPUT ON
declare
num number;
i number;
j number;
n number;
flag number;
x number;
begin
num:=&n;
n:=TRUNC(num/2);
for i in 2..n
loop
if(mod(num,i)=0) then
flag:=1;
exit;
else
flag:=0;
end if;
end loop;
if(flag=1) then
dbms_output.put_line(num||'is not prime no');
else
x:=prime_test(num);
if(x=0) then
insert into prime_entry values(seq.nextval,num);
else
dbms_output.put_line('Allready exist in table');
end if;
end if;
end;
/
```

Output:

```
SQL> @D:\sidd57\ass_8_3.txt

Function created.

Enter value for n: 4
old     9: num:=&n;
new     9: num:=4;
4is not prime no

PL/SQL procedure successfully completed.

SQL> select * from prime_entry;

    NUM_ID  PRIME_NUM
---------- ----------
         1          7

SQL> select * from prime_entry;

    NUM_ID  PRIME_NUM
---------- ----------
         1          7

SQL> |
```

2. **Problem Statement:** Create the following table:

2. Create the following table:

Table Name: Acc_details

Column_Name	Data type	Size	Attributes
Acc_no	Varchar2	8	Primary Key
Name	Varchar2	20	Not Null
Address	Varchar2	20	Not Null
DOB	Date		Not Null
Sex	Char	1	Not Null, Values ('M', 'F')
Contact_no	Number	10	Not Null
Last_trans_date	Date		Not Null
Total_amt	Number	12,4	Not Null
Acc_status	Char	1	Not Null, Values ('A', 'I')

Table Name: Transactions_Acc

Column_Name	Data type	Size	Attributes
Transaction_id	Number	8	Primary Key
Acc_no	Number	8	References Acc_details.Acc_no
Deposit_amt	Number	12,4	
Withdraw_amt	Number	12,4	
Mode_trans	Char	5	Not Null
Cheque_no	Number	6	Default 0
Trans_date	Date		Not Null

Query:

```
create table Transaction_Acc
(
Transaction_ID number(8) primary key,
Acc_no varchar2(8)  references Acc_details on DELETE CASCADE,
Deposit_amt number(12,4),
Withdraw_amt number(12,4),
Mode_trans char(5) not null,
check_no number(6) default 0,
Trans_date date not null
);


create table Acc_details
(
Acc_no varchar2(8) primary key,
Name varchar2(20) not null,
Address varchar2(50) not null,
Dob  date not null,
sex char(1) check (sex in ('M','F')),
contact_no number(10) not null,
last_trans_date date not null,
total_cost number(14,2) not null,
Acc_status char(1) not null check (Acc_status IN ('A','I'))
);
```

Output:

```
SQL> create table Acc_details
  2  (
  3  Acc_no varchar2(8) primary key,
  4  Name varchar2(20) not null,
  5  Address varchar2(50) not null,
  6  Dob  date not null,
  7  sex char(1) check (sex in ('M','F')),
  8  contact_no number(10) not null,
  9  last_trans_date date not null,
 10  total_cost number(14,2) not null,
 11  Acc_status char(1) not null check (Acc_status IN ('A','I'))
 12  );

Table created.

SQL>
SQL> create table Transaction_Acc
  2  (
  3  Transaction_ID number(8) primary key,
  4  Acc_no varchar2(8)  references Acc_details on DELETE CASCADE,
  5  Deposit_amt number(12,4),
  6  Withdraw_amt number(12,4),
  7  Mode_trans char(5) not null,
  8  check_no number(6) default 0,
  9  Trans_date date not null
 10  );

Table created.

SQL>
```

Problem Statement:

When a specific account will be deleted then all the transaction details from Transactions_acc will be deleted for that account number.

Query:

insert into Acc_details values('001','AMIT','CK-256','12-JAN-2012','M',9836793258,'13-JUN-2012',12000,'A');

insert into Transaction_Acc values('002','001',11000,5000,'A',101,'12-JUN-2012');
insert into Transaction_Acc values('003','001',12000,6000,'B',102,'12-JULY-2012');

Output:

```
SQL> insert into Acc_details values('001','AMIT','CK-256','12-JAN-2012','M',9836793258,'13-JUN-2012',12000,'A');

1 row created.

SQL>
SQL> insert into Transaction_Acc values('002','001',11000,5000,'A',101,'12-JUN-2012');

1 row created.

SQL> insert into Transaction_Acc values('003','001',12000,6000,'B',102,'12-JULY-2012');

1 row created.

SQL>
```