executed

Name: Qais Mohammed Haneef Warekar

Duration: 1 hour   **UCID:** 2023510066

Marks: 25 Marks

General Instructions:

Viva will be taken at the time of practical as well as after the practical if required.
The figures to the right indicate full marks.
If you are using any additional information, state it clearly.
Once you finish with the code show it to the examiner for testing. Write your answer in Word file and upload it on Moodle.

Q.			
A)	1	Consider the bank database given below. The primary keys are underlined and the data types are specified: BRANCH(<u>branch_name</u>:varchar2(30),branch_city:varchar2(30),assets:number(10)) CUSTOMER(<u>customer_name</u>:varchar2(30),customer_street:varchar2(30), customer_city:varchar2(30)) ACCOUNT(<u>account_number</u>:number(10),customer_name:varchar2(30), branch_name:varchar2(30), balance:number(10)) a) Create the above tables by properly specifying the primary keys and the foreign keys and named constraints. b) Enter atleast five tuples for each relation. c) List the details of all customers who belong to the 'P K Road' branch. d) List the customers who live in Mumbai and have their branch in Nagpur city . e) Give the details of all customers who have a balance above Rs.100000	10
B)		Create a trigger, which will display a message 'Balance below minimum!', whenever an updation of the ACCOUNT table causes the balance to fall below Rs. 1000.	10
C)		List the customers who have the same living city and branch city	5

1.a) create table branch (branch_name varchar (30) primary key, branch_city varchar(30), assets int);

create table customer (customer_name varchar(30) primary key, customer_street varchar(30), customer_city varchar(30));

create table account (account_number int primary key, customer_name varchar(30), branch_name varchar(30), balance int);

alter table account add constraint cust_fk foreign key (customer_name) references customer (customer_name) on update cascade;

alter table account add constraint branch_fk foreign key (branch_name) references branch (branch_name) on update cascade.

1.b) insert into branch values ('PK Road', 'Mumbai', 100000), ('M G Road', 'Nagpur', 200000), ('Nehru Road', 'Nagpur', 150000), ('James Road', 'Pune', 100000), ('Frank Road', 'Nagpur', 120000);

insert into customer values ('Harsh', 'Frank St', 'Mumbai'), ('Tim', 'Gilbert St', 'Mumbai'), ('Jim', 'Kernel St', 'Nagpur'), ('Ram', 'Paint St', 'Pune'), ('kumar', 'Jacob st', 'Mumbai');

```sql
insert into account values (101, 'Harsh', 'P K Road', 15000), (102, 'Tim',
'M G Road', 40000), (103, 'Jim', 'James Road', 20000), (104, 'Ram', 'Frank
Road', 25000), (105, 'Kumar', 'Nehru Road', 12000);
```

1.c)
```sql
select * from account where branch-name = 'P K Road';
```

1.d)
```sql
select distinct customer-name from customer
inner join {account} using (customer-name)
inner join branch using (branch-name)
where customer.customer_city = 'Mumbai' and branch.branch_city = 'Nagpur'
```

1.e)
```sql
select * from account where balance > 100000;
```

B)
```sql
create definer = 'root'@'localhost' trigger 'account_after_update'
after update on 'account'
for each row
begin
    if (new.balance < 1000)
    then signal sqlstate '45000' set message.txt = 'Balance
    below Minimum!';
    end if;
END;
```

c)
```sql
select customer-name, customer_city as Common_city from
customer
inner join {account}
using (customer-name)
inner join branch
using (branch-name)
where customer.customer_city = branch.branch-city;
```