Consider the following relations for a transport management system application:

DRIVER (DCODE, DNAME, DOB, GENDER)

CITY (CCODE, CNAME)
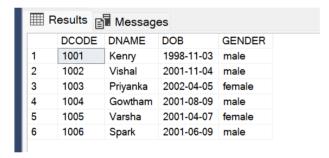
TRAVEL (Tripid, DCODE, DATE, FROM_CITY, TO_CITY)

a) Display the details of female drivers.

b) Display the trip details of trips from Madurai to Chennai.

c) Display the Tripid of the trips made by Kenry.

d) display the trip details of trips planned today.

e) Write a function to display the total number of trips from Mumbai to Delhi.

f) Write a trigger to log the details whenever there is any data change in the travel details

Show the log with user details, time of update in the table, along with the type of operation performed

```
create table DRIVER(DCODE int primary key,DNAME nvarchar(20) not null,DOB date, GENDER
nvarchar(10));
create table CITY(CCODE int ,CNAME nvarchar(20));
create table TRAVEL(Tripid int primary key,DCODE int foreign key references
DRIVER,DATE date,FROM_CITY nvarchar(20),TO_CITY nvarchar(20));
```
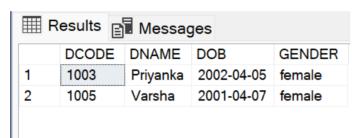
```
select * from driver;
```

Results | Messages

	DCODE	DNAME	DOB	GENDER
1	1001	Kenry	1998-11-03	male
2	1002	Vishal	2001-11-04	male
3	1003	Priyanka	2002-04-05	female
4	1004	Gowtham	2001-08-09	male
5	1005	Varsha	2001-04-07	female
6	1006	Spark	2001-06-09	male

```
select * from city;
```

Results | Messages

	CCODE	CNAME
1	628001	Chennai
2	628002	Madurai
3	628003	Mumbai
4	628004	Delhi
5	628005	Thoothukudi

```sql
select * from TRAVEL;
```

	Tripid	DCODE	DATE	FROM_CITY	TO_CITY
1	101	1001	2023-03-20	Madurai	Chennai
2	102	1003	2023-03-20	Mumbai	Delhi
3	103	1004	2023-03-20	Mumbai	Delhi
4	104	1005	2023-03-22	Chennai	Thoothukudi
5	105	1002	2023-03-21	Trichy	Chennai
6	106	1001	2023-03-24	Mumbai	Delhi
7	107	1006	2023-03-25	Chennai	Delhi

a)
```sql
select * from driver where GENDER = 'female';
```

	DCODE	DNAME	DOB	GENDER
1	1003	Priyanka	2002-04-05	female
2	1005	Varsha	2001-04-07	female

b)
```sql
select * from TRAVEL where FROM_CITY = 'Madurai' and TO_CITY = 'Chennai';
```

	Tripid	DCODE	DATE	FROM_CITY	TO_CITY
1	101	1001	2023-03-20	Madurai	Chennai

c)
```sql
select t.tripid trip_id_of_kenry
from travel t left join driver d on t.dcode = d.dcode
where d.dname = 'Kenry';
```
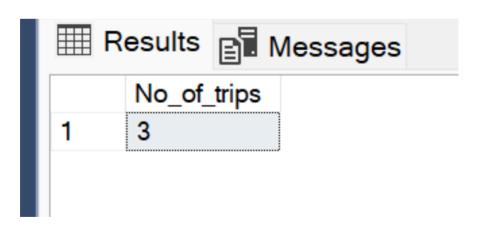
	trip_id_of_kenry
1	101
2	106

```
d) select * from travel
   where date = '2023/03/20';
```

	Tripid	DCODE	DATE	FROM_CITY	TO_CITY
1	101	1001	2023-03-20	Madurai	Chennai
2	102	1003	2023-03-20	Mumbai	Delhi
3	103	1004	2023-03-20	Mumbai	Delhi

```
e) CREATE FUNCTION udf_no_of_Trips_
   (

   )
   RETURNS TABLE
   AS
   RETURN
   (
         -- Add the SELECT statement with parameter references here
         SELECT count(FROM_CITY) No_of_trips from travel
         where from_city = 'Mumbai' and to_city = 'Delhi'
         group by FROM_CITY
   )
   GO
   select * from udf_no_of_trips_();
```

	No_of_trips
1	3

```
f)  create or alter trigger dbo.travelinsert
    on dbo.travel
    for insert
    AS
        INSERT INTO dbo.triggertables(UpdatedDate,Operation)select getdate(),
    'INSERT' from inserted;

    insert into travel values(108,1004,'2023-03-20', 'Chennai','Madurai');

    select name from sys.tables;

    select * from triggertables;

    create or alter trigger dbo.travelupdate
    on dbo.travel
    after update
    as
        INSERT INTO dbo.triggertables(UpdatedDate,Operation)select
    getdate(),'UPDATE' from deleted;
        go
    update travel set FROM_CITY='Cbe' where Tripid=108;

    create or alter trigger dbo.traveldelete
    on dbo.travel
    instead of delete
    AS
        INSERT INTO dbo.triggertables(UpdatedDate, Operation)select getdate(),
    'DELETE' from deleted;

    delete from travel where FROM_CITY='chennai';

    select * from travel;
```

Results / Messages

	LogID	UpdatedDate	Operation
1	1	2023-03-20 14:15:58.533	INSERT
2	2	2023-03-20 14:17:59.937	UPDATE
3	3	2023-03-20 14:19:08.207	DELETE
4	4	2023-03-20 14:19:08.207	DELETE