create database movie\_db;

use movie\_db;

Create table CUSTOMER\_MASTER

(

CUSTOMER\_ID Varchar(10),

CUSTOMER\_NAME Varchar(30) NOT NULL,

CONTACT\_NO BIGINT(10),

CONTACT\_ADD Varchar(20),

DATE\_OF\_REGISTRATION Date NOT NULL,

AGE Varchar(15)NOT NULL,

Constraint MT\_cts1 PRIMARY KEY(CUSTOMER\_ID)

);

Create table LIBRARY\_CARD\_MASTER

(

CARD\_ID Varchar(10),

DESCRIPTION Varchar(30) NOT NULL,

AMOUNT BIGINT(50),

NUMBER\_OF\_YEARS bigint(10) NOT NULL,

Constraint MT\_cts2 PRIMARY KEY(CARD\_ID)

);

Create table MOVIES\_MASTER

(

MOVIE\_ID Varchar(10),

MOVIE\_NAME Varchar(50) NOT NULL,

RELEASE\_DATE Varchar(30) NOT NULL,

LANGUAGE Varchar(30),

RATING int(2),

DURATION VARCHAR(10) NOT NULL,

MOVIE\_TYPE Varchar(3),

MOVIE\_CATEGORY VARCHAR(20) NOT NULL,

DIRECTOR VARCHAR(20) NOT NULL,

LEAD\_ROLE\_1 Varchar(3) NOT NULL,

LEAD\_ROLE\_2 VARCHAR(4) NOT NULL,

RENT\_COST BIGINT(10),

Constraint MT\_cts4 PRIMARY KEY(MOVIE\_ID)

);

Create table CUSTOMER\_CARD\_DETAILS

(

CUSTOMER\_ID Varchar(10),

CARD\_ID VARCHAR(10),

ISSUE\_DATE DATE NOT NULL,

Constraint MT\_cts3 PRIMARY KEY(CUSTOMER\_ID),

Constraint MT\_CTS41 FOREIGN KEY(CUSTOMER\_ID) References CUSTOMER\_MASTER(CUSTOMER\_ID),

Constraint MT\_CTS42 FOREIGN KEY(CARD\_ID) References LIBRARY\_CARD\_MASTER(CARD\_ID)

);

Create table CUSTOMER\_ISSUE\_DETAILS

(

ISSUE\_ID Varchar(10) NOT NULL,

CUSTOMER\_ID Varchar(10) NOT NULL,

MOVIE\_ID VARCHAR(10),

ISSUE\_DATE Date NOT NULL,

RETURN\_DATE Date NOT NULL,

ACTUAL\_DATE\_RETURN Date NOT NULL,

Constraint MT\_cts5 PRIMARY KEY(ISSUE\_ID),

Constraint MT\_Mem FOREIGN KEY(CUSTOMER\_ID) References CUSTOMER\_MASTER(CUSTOMER\_ID),

Constraint MT\_Mem1 FOREIGN KEY(MOVIE\_ID) References MOVIES\_MASTER(MOVIE\_ID)

);

/\*Insert records \*/

Insert into CUSTOMER\_MASTER Values('CUS001', 'AMIT', 9876543210,'ADD1', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS002', 'ABDHUL', 8765432109,'ADD2', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS003', 'GAYAN', 7654321098,'ADD3', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS004', 'RADHA', 6543210987,'ADD4', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS005', 'GURU', NULL,'ADD5', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS006', 'MOHAN', 4321098765,'ADD6', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS007', 'NAME7', 3210987654,'ADD7', '2012-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS008', 'NAME8', 2109876543,'ADD8', '2013-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS009', 'NAME9', NULL,'ADD9', '2013-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS010', 'NAM10', 9934567890,'ADD10', '2013-02-12', '21');

Insert into CUSTOMER\_MASTER Values('CUS011', 'NAM11', 9875678910,'ADD11', '2013-02-12', '21');

Insert into LIBRARY\_CARD\_MASTER Values('CR001', 'DES1', 200, 5);

Insert into LIBRARY\_CARD\_MASTER Values('CR002', 'DES2', 400, 9);

Insert into LIBRARY\_CARD\_MASTER Values('CR003', 'DES3', 600, 8);

Insert into LIBRARY\_CARD\_MASTER Values('CR004', 'DES4', 800, 7);

Insert into LIBRARY\_CARD\_MASTER Values('CR005', 'DES5', 1200, 6);

Insert into MOVIES\_MASTER Values('MV001', 'DIEHARD', '2012-05-13','ENGLISH', 4 , '2HRS', 'U/A','ACTION','DIR1','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV002', 'THE MATRIX', '2012-05-13','ENGLISH', 4 , '2HRS', 'A','ACTION','DIR2','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV003', 'INCEPTION', '2012-05-13','ENGLISH', 4 , '2HRS', 'U/A','ACTION','DIR3','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV004', 'DARK KNIGHT', '2012-05-13','ENGLISH', 4 , '2HRS', 'A','ACTION','DIR4','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV005', 'OFFICE S', '2012-05-13','ENGLISH', 4 , '2HRS', 'U/A','COMEDY','DIR5','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV006', 'SHAWN OF DEAD', '2012-05-13','ENGLISH', 4 , '2HRS', 'U/A','COMEDY','DIR6','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV007', 'YOUNG FRANKEN', '2012-05-13','ENGLISH', 4 , '2HRS', 'U/A','COMEDY','DIR7','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV008', 'CAS', '2012-05-13','ENGLISH', 4 , '2HRS', 'A','ROMANCE','DIR8','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV009', 'GWW', '2012-05-13','ENGLISH', 4 , '2HRS', 'A','ROMANCE','DIR9','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV010', 'TITANIC', '2012-05-13','ENGLISH', 4 , '2HRS', 'A','ROMANCE','DIR10','L1','L2',100);

Insert into MOVIES\_MASTER Values('MV011', 'THE NOTE BOOK', '2012-05-13','ENGLISH', 4 , '2HRS', 'A','ROMANCE','DIR11','L1','L2',100);

Insert into CUSTOMER\_CARD\_DETAILS Values('CUS001', 'CR001', '2012-05-13');

Insert into CUSTOMER\_CARD\_DETAILS Values('CUS002', 'CR002', '2012-05-13');

Insert into CUSTOMER\_CARD\_DETAILS Values('CUS003', 'CR002', '2013-05-13');

Insert into CUSTOMER\_CARD\_DETAILS Values('CUS004', 'CR003', '2013-05-13');

Insert into CUSTOMER\_CARD\_DETAILS Values('CUS005', 'CR003', '2012-05-13');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS001', 'CUS001', 'MV001', '2012-05-13', '2012-05-13','2012-05-13');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS002', 'CUS001', 'MV001', '2012-05-01', '2012-05-16','2012-05-16');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS003', 'CUS002', 'MV004', '2012-05-02', '2012-05-06','2012-05-16');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS004', 'CUS002', 'MV004', '2012-04-03', '2012-04-16','2012-04-20');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS005', 'CUS002', 'MV009', '2012-04-04', '2012-04-16','2012-04-20');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS006', 'CUS003', 'MV002', '2012-03-30', '2012-04-15','2012-04-20');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS007', 'CUS003', 'MV003', '2012-04-20', '2012-05-05','2012-05-05');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS008', 'CUS003', 'MV005', '2012-04-21', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS009', 'CUS003', 'MV001', '2012-04-22', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS010', 'CUS003', 'MV009', '2012-04-22', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS011', 'CUS003', 'MV010', '2012-04-23', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS012', 'CUS003', 'MV010', '2012-04-24', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS013', 'CUS003', 'MV008', '2012-04-25', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS014', 'CUS004', 'MV007', '2012-04-26', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS015', 'CUS004', 'MV006', '2012-04-27', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS016', 'CUS004', 'MV006', '2012-04-28', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS017', 'CUS004', 'MV001', '2012-04-29', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS018', 'CUS010', 'MV008', '2012-04-24', '2012-05-07','2012-05-25');

Insert into CUSTOMER\_ISSUE\_DETAILS Values ('IS019', 'CUS011', 'MV009', '2012-04-27', '2012-05-07','2012-05-25');

/\*1.count the members who has gold cards\*/

select count(customer\_id) from customer\_card\_details

where card\_id IN (

select card\_id from library\_card\_master where description='des1'

);

/\* 2.display the name of member who issued movie and the count of the

movies issued and display 0 for the member who have not issued any

movie

\*/

select cm.customer\_id,customer\_name,IF((NULLIF(issue\_id,0)),0,count(issue\_id))

FROM customer\_master cm LEFT OUTER JOIN customer\_issue\_details ci

ON cm.customer\_id=ci.customer\_id

group BY cm.customer\_id;

/\* OR \*/

select cm.customer\_id,customer\_name,count(issue\_id) count

FROM customer\_master cm LEFT OUTER JOIN customer\_issue\_details ci

ON cm.customer\_id=ci.customer\_id

group BY cm.customer\_id

UNION

select customer\_id,customer\_name,0 count

FROM customer\_master

WHERE customer\_id NOT IN(select customer\_id from customer\_issue\_details);

/\* 3.display the name of the person starting with letter 'r' and category is 'comedy'\*/

select customer\_name

FROM customer\_master

where customer\_name like 'r%'

and

customer\_id IN

(select customer\_id from movies\_master m JOIN customer\_issue\_details c ON m.movie\_id=c.movie\_id

where movie\_category='comedy'

);

/\* 4.display id,name & total rent of customers for movie issued \*/

select a.customer\_id,customer\_name , count(c.movie\_id)\*rent\_cost amt

FROM customer\_master a JOIN customer\_issue\_details b JOIN (select movie\_id,rent\_cost from movies\_master) c

ON a.customer\_id=b.customer\_id

and b.movie\_id=c.movie\_id

group by a.customer\_id;

/\*5.display id,name,card id,amount in $(amount/54.42) upto 0 decimals\*/

select a.customer\_id,customer\_name,c.card\_id,ROUND(amount/54.52)

FROM customer\_master a,customer\_card\_details b,library\_card\_master c

where a.customer\_id=b.customer\_id

and b.card\_id=c.card\_id group by a.customer\_id;

/\* 6.display id,name of customers who dont have library card but still

have issued the movie\*/

select customer\_id,customer\_name FROM customer\_master

WHERE customer\_id NOT IN

(select customer\_id from customer\_card\_details a,library\_card\_master b

where a.card\_id=b.card\_id

)

and

customer\_id IN

(select distinct a.customer\_id

FROM customer\_master a,customer\_issue\_details b

where a.customer\_id=b.customer\_id

);

/\* 7.display the no.of customers with first letter 'r' and have paid fine

i.e actual return date is greater than return date

\*/

select count(a.customer\_id) cnt

FROM customer\_master a JOIN customer\_issue\_details b

ON a.customer\_id = b.customer\_id

where customer\_name like 'a%'

and return\_date > actual\_date\_return

;

/\* 8.display customer name,customer id who have issued max and min no.of

movies issued \*/

select customer\_id,customer\_name

from customer\_master

where customer\_id IN(

select customer\_id

from customer\_issue\_details

group by customer\_id

having count(issue\_id)=(

select count(issue\_id)

from customer\_issue\_details

group by customer\_id order by count(issue\_id) desc limit 1

)

union

select customer\_id

from customer\_issue\_details

group by customer\_id

having count(issue\_id)=(

select count(issue\_id)

from customer\_issue\_details

group by customer\_id order by count(issue\_id) asc limit 1

)

);

/\* 9.display id,name,mobile num and description of all customers.if

mobile num is not available then display address as alias contact,

for

those who does't have library cards display null as description \*/

select cm.customer\_id,customer\_name,coalesce(CAST(contact\_no AS CHAR),CAST(contact\_add AS CHAR)) datail,description

from customer\_master cm,

customer\_card\_details cd

,library\_card\_master lc

where cm.customer\_id=cd.customer\_id

and cd.card\_id=lc.card\_id

union

select customer\_id,customer\_name,coalesce(CAST(contact\_no AS CHAR),CAST(contact\_add AS CHAR)) detail ,null description

from customer\_master

where customer\_id NOT IN

(

select customer\_id from customer\_card\_details

)

;

/\* 10.display customer details and movie id for those who issued same movie more than one time OR

//dislay customer details who watched same movie more than once

\*/

select a.customer\_id,a.customer\_name,a. age,c.movie\_id,c.movie\_name,count(c.movie\_id)

from customer\_master a JOIN customer\_issue\_details b JOIN movies\_master c

ON a.customer\_id = b.customer\_id

and b.movie\_id = c.movie\_id

group by a.customer\_id,c.movie\_id

having count(c.movie\_id)>1;

/\*11.display customer information those who has library cards \*/

select cm.customer\_id ,customer\_name

from customer\_master cm JOIN customer\_card\_details cc

ON cm.customer\_id=cc.customer\_id;

/\* 12.display the members who watch the movie but doesnt have card\*/

select cm.customer\_id,customer\_name

from customer\_master cm,customer\_issue\_details cd

where cm.customer\_id=cd.customer\_id

and cm.customer\_id NOT IN(

select customer\_id from customer\_card\_details

)

;

/\* 13.display sr no as 2 digits of issue id,emp id,movie watched,video id

and sort by sr no\*/

select substring(issue\_id,4,2) sr\_no,customer\_id,movie\_id

from customer\_issue\_details

order by sr\_no;

/\* 14.display total revenue spent on videos by each customer\*/

select cd.customer\_id,sum(rent\_cost) rent

from customer\_issue\_details cd ,movies\_master mm

where cd.movie\_id = mm.movie\_id

group by cd.customer\_id

union

select customer\_id,0 rent

from customer\_master

where customer\_id NOT IN(select customer\_id from customer\_issue\_details);

/\*15.display customer name in perfect order

i.e 1st letter in ucase remaining lcase

\*/

select customer\_id , concat(ucase(substring(customer\_name,1,1)),lcase(substring(customer\_name,2))) name from customer\_master

order by name;

/\*16.count how many times a movie issued and arrange them in desc order

and display 0 for the movie \*/

select mm.movie\_id ,count(mm.movie\_id) cnt

from customer\_issue\_details cd,movies\_master mm

where cd.movie\_id= mm.movie\_id

group by mm.movie\_id

UNION

select movie\_id,0 cnt

from movies\_master

where movie\_id NOT IN(select movie\_id from customer\_issue\_details);

/\* 17.waq to display cus id and cus name and address as if phone num

presents display phone num otherwise address.\*/

select customer\_id , coalesce(CAST(contact\_no AS CHAR),CAST(contact\_add AS CHAR)) addr

from customer\_master;

/\* 18.waq that num of customers registered in 2012 year and provided contact num

use NO\_OF\_CUSTOMERS as alias name.

\*/

select count(customer\_id) NO\_OF\_CUSTOMER

from customer\_master

where extract(year from date\_of\_registration)='2012'

and contact\_no IS NOT NULL;

/\* 19.display customer id, cus name,year of registration,library card id,card issue date

alias name registered\_year for year of registration.

\*/

select cm.customer\_id,customer\_name,

extract(year from date\_of\_registration) registered\_year, lm.card\_id, CAST(issue\_date AS DATE) i\_date

from customer\_master cm, customer\_card\_details cd, library\_card\_master lm

where cm.customer\_id = cd.customer\_id

and lm.card\_id= cd.card\_id

union

select cm.customer\_id,customer\_name,

extract(year from date\_of\_registration) registered\_year, 0, CAST(0 AS DATE)

from customer\_master cm

where cm.customer\_id NOT IN(select customer\_id from customer\_card\_details);

/\*20. (\*\*\*REPEATED)waq to display movie name and num of times movie issued to

customers..in case of no movie issued

to customers display 0.. use alias name as NO\_OF\_TIMES

\*/

/\*21.waq to display customer id and customer name ,num of times movie issued

to customer in comedy movie category

display only customers who has issude more than once

\*/

select cm.customer\_id,customer\_name,count(mm.movie\_id) cnt

from customer\_master cm, customer\_issue\_details cd ,movies\_master mm

where cm.customer\_id = cd.customer\_id

and cd.movie\_id = mm. movie\_id

and movie\_category='comedy'

group by cm.customer\_id

having cnt>1;

/\*22.(\*\*\*REPEATED)waq to display customerid and total rent paid by them.

use alias name as total\_cost.

\*/

/\*23.waq to display customerid,cusname,contactno,num of movies issued to customer based

on category and category

display the customer who has issude for more than one movie from that caregory.

display phone num as "+91-987-654-3210".

\*/

select cm.customer\_id, contact\_no ,concat('+91 - ',CAST(substring(contact\_no,1,3) AS CHAR),' - ',CAST(substring(contact\_no,4,3) AS CHAR), ' - ',CAST(substring(contact\_no,7,4) AS CHAR)) r\_date , count(mm.movie\_id) cnt

from customer\_master cm, customer\_issue\_details cd , movies\_master mm

where cm.customer\_id= cd.customer\_id

and cd.movie\_id = mm. movie\_id

group by cm.customer\_id , movie\_category;

select \* from CUSTOMER\_MASTER;

select \* from LIBRARY\_CARD\_MASTER;

select \* from MOVIEs\_MASTER;

select \* from customer\_card\_details;

select \* from customer\_issue\_details;