|  |
| --- |
| 1.count the members who has gold cards |
|  |
| ans:select count(customer\_id) from customer\_card\_details where card\_id in  (select card\_id from library\_card\_master where description='gold card'); |
|  |
| 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 |
|  |
| ans:select customer\_name,count(movie\_id) count  from customer\_issue\_details a,customer\_master b  where a.customer\_id=b.customer\_id  group by b.customer\_id union  select customer\_name,0 as 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' |
| ans: |
| select distinct a.customer\_name  from customer\_master a,customer\_issue\_details b,movies\_master c  where a.customer\_id=b.customer\_id and b.movie\_id=c.movie\_id and c.movie\_category='comedy'  and a.customer\_name like 'r%'; |
|  |
|  |
| 4.display id,name & total rent of customers for movie issued |
| ans:select a.customer\_id,customer\_name,count(a.movie\_id)\*c.rent\_cost rent  from customer\_issue\_details a,customer\_master b ,(select movie\_id,rent\_cost from movies\_master) c  where a.customer\_id=b.customer\_id and a.movie\_id=c.movie\_id  group by b.customer\_id; |
|  |
|  |
| 5.display id,name,card id,amount in $(amount/54.42) upto 0 decimals |
| ans: |
| select a.customer\_id,customer\_name,b.card\_id,round(amount/54.42) Amount  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; |
|  |
| 6.display id,name of customers who dont have library card but still have issued the movie |
| ans:  select distinct b.customer\_id,customer\_name  from customer\_issue\_details a,customer\_master b  where a.customer\_id=b.customer\_id  and a.customer\_id not in (select customer\_id from customer\_card\_details); |
|  |
| 7.display the no.of customers with first letter 'r' and have paid fine  i.e actual return date is greater than return date |
| ans:  select count(b.customer\_id) count from (select customer\_id  from customer\_issue\_details  where return\_date>actual\_date\_return  group by customer\_id) b,customer\_master c  where b.customer\_id=c.customer\_id and c.customer\_name like 'r%'; |
|  |
| 8.display customer name,customer id who have issued max and min no.of |
| movies issued |
| ans:  select customer\_name,a.customer\_id  from customer\_issue\_details a,customer\_master b  where a.customer\_id=b.customer\_id  group by a.customer\_id  having count(a.movie\_id)=(  select min(a.count) min from (select customer\_id,count(movie\_id) count  from customer\_issue\_details  group by customer\_id) a)  or count(a.movie\_id)=(  select max(b.count) min from (select customer\_id,count(movie\_id) count  from customer\_issue\_details  group by customer\_id) b); |
|  |
|  |
| 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 |
|  |
| ans:  select a.customer\_id,a.customer\_name,coalesce(a.contact\_no,a.contact\_add) contact,description  from customer\_master a,library\_card\_master b,customer\_card\_details c  where a.customer\_id=c.customer\_id and c.card\_id=b.card\_id union  select a.customer\_id,a.customer\_name,coalesce(a.contact\_no,a.contact\_add) contact,null as description  from customer\_master a  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 |
| ans:select m.customer\_id,customer\_name,movie\_id,count(i.movie\_id) as count\_of\_movie  from customer\_master m join customer\_issue\_details i  on m.customer\_id = i.customer\_id  group by movie\_id  having count(movie\_id)>1; |
|  |
|  |
|  |
|  |
| 11.display customer information those who has library cards |
| ans:  select a.customer\_id,customer\_name  from customer\_card\_details a,customer\_master b,library\_card\_master c  where a.customer\_id=b.customer\_id and a.card\_id=c.card\_id; |
|  |
| 12.display the members who watch the movie but doesnt have card |
| ans:select distinct b.customer\_id,customer\_name  from customer\_issue\_details a,customer\_master b  where a.customer\_id=b.customer\_id  and a.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 |
| ans:  select substring(issue\_id,4) sr\_no,issue\_id as video\_id,customer\_id,movie\_id  from customer\_issue\_details order by sr\_no; |
|  |
|  |
|  |
| 14.display total revenue spent on videos by each customer |
| ans:select a.customer\_id,count(a.movie\_id)\*c.rent\_cost rent  from customer\_issue\_details a,customer\_master b ,(select  movie\_id,rent\_cost from movies\_master) c  where a.customer\_id=b.customer\_id and a.movie\_id=c.movie\_id  group by b.customer\_id union  select customer\_id,0 as 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 |
| ans:  select concat(ucase(substring(customer\_name,1,1)),lcase(substring(customer\_name,2,5)) )  from customer\_master; |
|  |
|  |
| 16.count how many times a movie issued and arrange them in desc order and display 0 for the movie not issued |
| ans:  select \* from (select a.movie\_id,count(b.movie\_id) count  from movies\_master a,customer\_issue\_details b  where a.movie\_id=b.movie\_id  group by a.movie\_id union  select movie\_id ,0 as count from movies\_master  where movie\_id not in (select movie\_id from customer\_issue\_details)) x order by movie\_id desc; |
|  |
| 17.waq to display cus id and cus name and address as if phone num presents display phone num otherwise address. |
| SELECT CUSTOMER\_ID,CUSTOMER\_NAME,coalesce(CAST(CONTACT\_NO AS CHAR),CAST(CONTACT\_ADD AS CHAR))  from customer\_master;    SELECT CUSTOMER\_ID,CUSTOMER\_NAME,(CASE  WHEN CAST(CONTACT\_NO AS CHAR) IS NULL THEN CAST(CONTACT\_ADD AS CHAR)  ELSE CAST(CONTACT\_NO AS CHAR)  END) AS CONTACT  FROM CUSTOMER\_MASTER; |
| 18.waq that num of customers registered in 2012 year and provided contact num  use NO\_OF\_CUSTOMERS as alias name. |
| ANS:SELECT COUNT(\*) AS NO\_OF\_CUSTOMERS  FROM CUSTOMER\_MASTER  WHERE DATE\_OF\_REGISTRATION LIKE '%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. |
| ANS:SELECT seLect C.CUSTOMER\_ID,C.CUSTOMER\_NAME,year(C.DATE\_OF\_REGISTRATION) REGISTERED\_YEAR,L.CARD\_ID,D.ISSUE\_DATE  FROM CUSTOMER\_MASTER C,LIBRARY\_CARD\_MASTER L,CUSTOMER\_CARD\_DETAILS D  WHERE C.CUSTOMER\_ID=D.CUSTOMER\_ID AND L.CARD\_ID=D.CARD\_ID; |
|  |
| 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 |
| SELECT M.MOVIE\_NAME,COUNT(\*) AS NO\_OF\_TIMES  FROM CUSTOMER\_ISSUE\_DETAILS D,MOVIES\_MASTER M,CUSTOMER\_MASTER C  WHERE M.MOVIE\_ID=D.MOVIE\_ID AND D.CUSTOMER\_ID=C.CUSTOMER\_ID  GROUP BY C.CUSTOMER\_ID  union  SELECT MOVIE\_NAME,0  FROM MOVIES\_MASTER  WHERE MOVIE\_NAME NOT IN (SELECT M.MOVIE\_NAME  FROM CUSTOMER\_ISSUE\_DETAILS D,MOVIES\_MASTER M,CUSTOMER\_MASTER C  WHERE M.MOVIE\_ID=D.MOVIE\_ID AND D.CUSTOMER\_ID=C.CUSTOMER\_ID  GROUP BY C.CUSTOMER\_ID); |
|  |
| 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 |
| ANS:  SELECT M.CUSTOMER\_NAME,M.CUSTOMER\_ID,P.NO\_OF\_TIMES  FROM CUSTOMER\_MASTER M,  (SELECT I.CUSTOMER\_ID,COUNT(\*) AS NO\_OF\_TIMES  FROM CUSTOMER\_ISSUE\_DETAILS I,(  SELECT MOVIE\_ID,MOVIE\_CATEGORY  FROM MOVIES\_MASTER  WHERE MOVIE\_CATEGORY='COMEDY')T  WHERE I.MOVIE\_ID=T.MOVIE\_ID  GROUP BY I.CUSTOMER\_ID  HAVING COUNT(\*)>1)P  WHERE M.CUSTOMER\_ID=P.CUSTOMER\_ID; |
|  |
| 22.(\*\*\*REPEATED)waq to display customerid and total rent paid by them.  use alias name as total\_cost. |
| ANS:select \* from movies\_master;  select c.customer\_id,sum(rent\_cost) as total\_cost  from customer\_master c,customer\_issue\_details d,movies\_master m  where c.customer\_id=d.customer\_id and m.movie\_id=d.movie\_id  group by customer\_id; |
|  |
| 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". |
|  |
| ANS:select m.customer\_id,customer\_name,  concat('+91-',substring(contact\_no,1,3),'-',substring(contact\_no,4,3),'-',substring(contact\_no,7) )as contact\_no,  count(i.movie\_id) as no\_of\_movies,movie\_category  from customer\_master m join customer\_issue\_details i join movies\_master mv  on m.customer\_id = i.customer\_id and i.movie\_id = mv.movie\_id  group by m.customer\_id  having count(\*) >1; |