MINI PROJECT

VIDEO MANAGEMENT SYSTEM

NAME : SANJEEV N
DATE : 14/07/2025

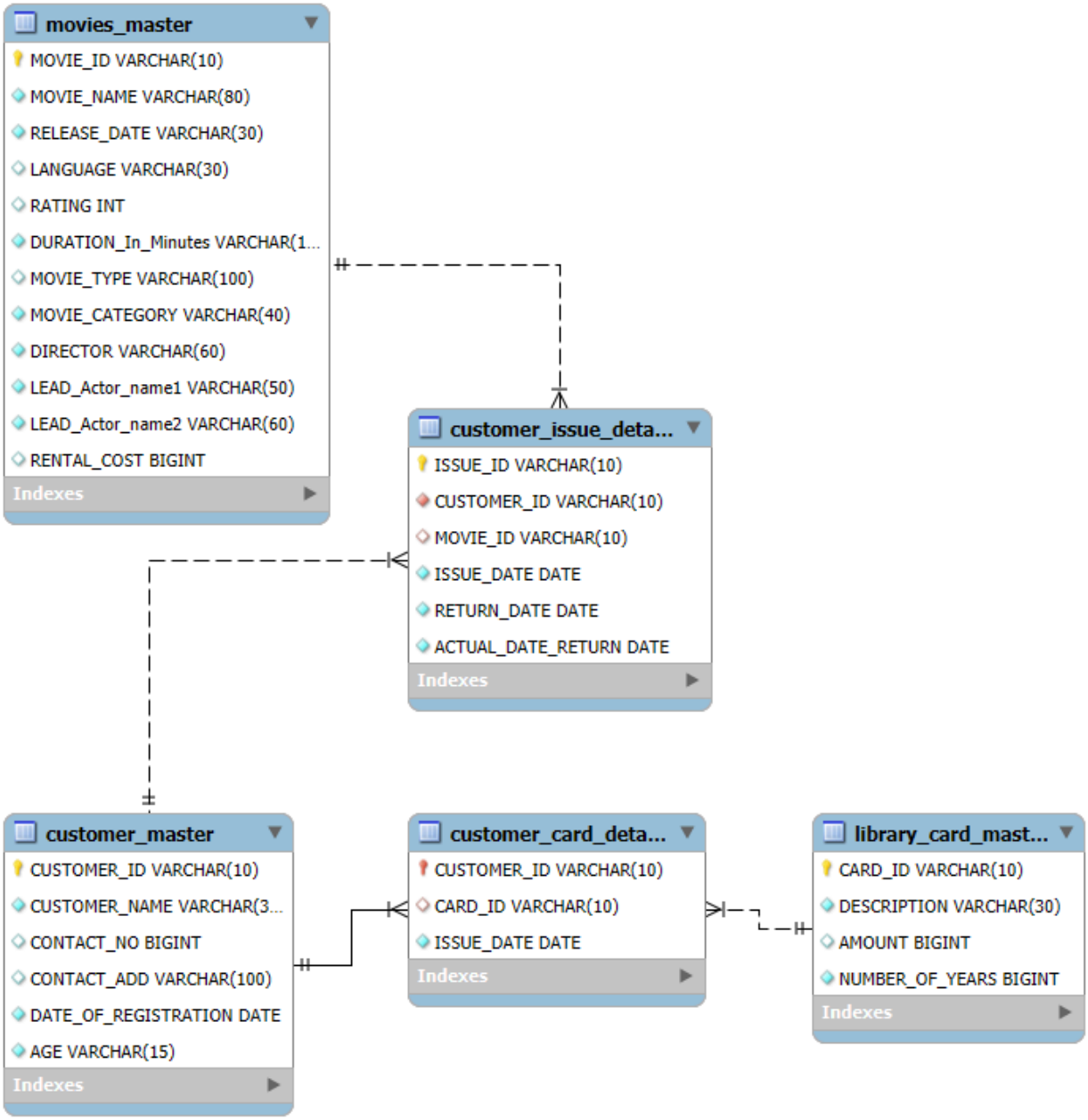
```
 create table CUSTOMER_MASTER
(
        CUSTOMER_ID Varchar(10),
        CUSTOMER_NAME Varchar(30) NOT NULL,
        CONTACT_NO BIGINT(20),
        CONTACT_ADD Varchar(100),
        DATE_OF_REGISTRATION Date NOT NULL,
        AGE Varchar(15)NOT NULL,
        Constraint MT_cts1 PRIMARY KEY(CUSTOMER_ID)
);


 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)
);



 Create table MOVIES_MASTER
(
        MOVIE_ID Varchar(10),
        MOVIE_NAME Varchar(80) NOT NULL,
        RELEASE_DATE Varchar(30) NOT NULL,
        LANGUAGE Varchar(30),
```

```
        RATING int(2),
        DURATION_In_Minutes VARCHAR(10) NOT NULL,
        MOVIE_TYPE Varchar(100),
        MOVIE_CATEGORY VARCHAR(40) NOT NULL,
        DIRECTOR VARCHAR(60) NOT NULL,
        LEAD_Actor_name1 Varchar(50) NOT NULL,
        LEAD_Actor_name2 VARCHAR(60) NOT NULL,
        RENTAL_COST BIGINT(10),
        Constraint MT_cts4 PRIMARY KEY(MOVIE_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 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)
);
```

ER DIAGRAM


movies_master
MOVIE_ID VARCHAR(10)
MOVIE_NAME VARCHAR(80)
RELEASE_DATE VARCHAR(30)
LANGUAGE VARCHAR(30)
RATING INT
DURATION_In_Minutes VARCHAR(1...
MOVIE_TYPE VARCHAR(100)
MOVIE_CATEGORY VARCHAR(40)
DIRECTOR VARCHAR(60)
LEAD_Actor_name1 VARCHAR(50)
LEAD_Actor_name2 VARCHAR(60)
RENTAL_COST BIGINT
Indexes
customer_issue_deta...
ISSUE_ID VARCHAR(10)
CUSTOMER_ID VARCHAR(10)
MOVIE_ID VARCHAR(10)
ISSUE_DATE DATE
RETURN_DATE DATE
ACTUAL_DATE_RETURN DATE
Indexes
customer_master
CUSTOMER_ID VARCHAR(10)
CUSTOMER_NAME VARCHAR(3...
CONTACT_NO BIGINT
CONTACT_ADD VARCHAR(100)
DATE_OF_REGISTRATION DATE
AGE VARCHAR(15)
Indexes
customer_card_deta...
CUSTOMER_ID VARCHAR(10)
CARD_ID VARCHAR(10)
ISSUE_DATE DATE
Indexes
library_card_mast...
CARD_ID VARCHAR(10)
DESCRIPTION VARCHAR(30)
AMOUNT BIGINT
NUMBER_OF_YEARS BIGINT
Indexes

PROBLEM 1: Write a query to display movie names and number of times that movie is issued to customers. In case movies are never issued to customers display number of times as 0.

```
select m.movie_name,  count(c.issue_id) as ISSUE_COUNT from movies_master m left join
customer_issue_details c on m.movie_id = c.movie_id
group by m.movie_name order by issue_count desc,  m.movie_name asc ;
```

movie_name	ISSUE_COUNT
DIE HARD	4
GONE WITH THE WIND	3
CASABLANCA	2
SHAUN OF THE DEAD	2
THE DARK KNIGHT	2
TITANIC	2
INCEPTION	1
THE MATRIX	1
OFFICE SPACE	1
YOUNG FRANKENSTEIN	1
THE NOTEBOOK	0

PROBLEM 2 : Write a query to display id, name, age, contact no of customers whose age is greater than 25 and who have registered in the year 2012. Display contact no in the below format +91-XXX-XXX-XXXX example +91-987-678-3434 and use the alias name as "CONTACT_ISD". If the contact no is null then display as 'N/A' Sort all the records in ascending order based on age and then by name.

```
select  customer_id, customer_name,age,
case when contact_no is not null then concat('+91-', substring(contact_no,1,3),'-',
substring(contact_no,4,3),'-',substring(contact_no,7,4))
 ELSE 'N/A'
END AS CONTACT_ISD from customer_master where age > 25 and year(date_of_registration)
= 2012;
```

customer_id	customer_name	age	CONTACT_ISD
C00002	AGNESH	35	+91-892-315-6781
C00004	RAJIB MITRA	45	+91-983-035-6781
C00005	SHIV PRASAD	30	N/A
C00007	GEETHA REDDY	30	+91-897-616-7890

PROBLEM 3: Write a query to display the movie category and number of movies in that category. Display records based on number of movies from higher to lower order and then by movie category in ascending order.

```
select movie_category, count(*) as no_of_movies from movies_master
group by movie_category order by no_of_movies desc, movie_category asc;
```

movie_category	no_of_movies
ACTION	3
ROMANCE	2
COMEDY	1
ROMANCE	1
ACTION	1
COMEDY	1
COMEDY	1
ROMANCE	1

PROBLEM 4 : Write a query to display the number of customers having card with description "Gold card". Use CUSTOMER_COUNT as alias name for number of customers.

```
select count(*) as customer_count from customer_card_details a
join library_card_master b on a.card_id = b.card_id WHERE b.description like '%gold card%';
```

customer_count
2

PROBLEM 5 : Write a query to display the customer id, customer name, year of registration, library card id, card issue date of all the customers who hold library card. Display the records sorted by customer name in descending order. Use REGISTERED_YEAR as alias name for year of registration.

```
select a.customer_id, a.customer_name ,year(a.date_of_registration) as REGISTERED_YEAR,
 b.card_id, b.issue_date from customer_master a join customer_card_details b on
a.customer_id = b.customer_id
 order by a.customer_name;
```

customer_id	customer_name	REGISTERED_YEAR	card_id	issue_date
C00002	AGNESH	2012	CRD002	2012-05-13
C00001	NITIN	2012	CRD001	2012-05-13
C00003	T RAMACHANDRAN	2012	CRD002	2013-05-13
C00004	RAJIB MITRA	2012	CRD003	2013-05-13
C00005	SHIV PRASAD	2012	CRD003	2012-05-13

PROBLEM 6 : Write a query to display issue id, customer id, customer name for the customers who have paid fine and whose name starts with 'R'. Fine is calculated based on return date and actual date of return. If the date of actual return is after date of return then fine need to be paid by the customer.

Display the records sorted in ascending order based on customer name.

select a.issue_id, b.customer_id, b.customer_name from customer_issue_details a join customer_master b
on a.customer_id = b.customer_id where b.customer_name like 'R%' and a.actual_date_return > a.return_date;

issue_id	customer_id	customer_name
I00007	C00004	RAJIB MITRA
I00008	C00010	RAGHAV SINGH

PROBLEM 7 : Write a query to display customer id, customer name, card id, card description and card amount in dollars of customers who have taken movie on the same day the library card is registered. For Example Assume John registered a library card on 12th Jan 2013 and he took a movie on 12th Jan 2013 then display his details. AMOUNT_DOLLAR = amount/85.8 and round it to zero decimal places and display as $Amount. Example Assume 500 is the amount then dollar value will be $10. Use AMOUNT_DOLLAR as alias name for amount in dollar. Display the records in ascending order based on customer name.

select a.customer_id,a.customer_name, b.card_id, c.description,
concat('$', round(c.amount/85.8,0)) AS AMOUNT_DOLLAR from customer_master a join customer_card_details b
on a.customer_id = b.customer_id join library_card_master c on
b.card_id = c.card_id join customer_issue_details d on a.customer_id = d.customer_id
where b.issue_date = d.issue_date order by a.customer_name asc;

	customer_id	customer_name	card_id	description	AMOUNT_DOLLAR
▶	C00001	NITIN	CRD001	SILVER CARD	$12
	C00001	NITIN	CRD001	SILVER CARD	$12

PROBLEM 8 : Write a query to display the customer name and number of movies issued to that customer sorted by customer name in ascending order. If a customer has not been issued with any movie then display 0. Use MOVIE_COUNT as alias name for number of movies issued.

```
select a.customer_name, count(b.issue_id) as MOVIE_COUNT
from customer_master a left join customer_issue_details b on a.customer_id = b.customer_id
group by a.customer_name
order by a.customer_name asc;
```

	customer_name	MOVIE_COUNT
▶	AGNESH	3
	NITIN	2
	T RAMACHANDRAN	8
	AJAY GHOSH	0
	GEETHA REDDY	0
	RAGHAV SINGH	1
	RAJ SEKHANRAN	1
	RAJAN PILLAI	0
	RAJIB MITRA	4
	RIA NATRAJAN	0
	SHIV PRASAD	0

PROBLEM 9 : Write a query to display the issue id, issue date, customer id, customer name and contact number for videos that are issued in the year 2013.Display the records in descending order based on issue date of the video.

select a.issue_id, a.issue_date, b.customer_id,b.customer_name, b.contact_no
from customer_issue_details a join customer_master b on a.customer_id = b.customer_id
where year(a.issue_date) = 2013 order by issue_date desc;

issue_id	issue_date	customer_id	customer_name	contact_no
I00017	2013-04-15	C00003	T RAMACHANDRAN	9831289761
I00009	2013-03-16	C00011	RAJ SEKHANRAN	8423178906
I00016	2013-03-05	C00003	T RAMACHANDRAN	9831289761
I00008	2013-03-02	C00010	RAGHAV SINGH	9675167890
I00015	2013-02-03	C00003	T RAMACHANDRAN	9831289761
I00014	2013-01-02	C00003	T RAMACHANDRAN	9831289761

PROBLEM 10 : Write a query to display the director's name, number of movies directed by the director who directed more than one movie. Display the director name in capital letters. Use DIRECTOR_NAME as alias name for director name column Display the records sorted in ascending order based on director_name.

select director as DIRECTOR_NAME, count(*) as NO_OF_MOVIES from movies_master
group by director having count(*) > 1 order by director_name asc;

DIRECTOR_NAME	NO_OF_MOVIES
CHRISTOPHER NOLAN	2