Steven Ha

Kimberly Sandoval

Noelle Nordgren

Robin Slaiwa

MIS380 Thurs 4:00-6:30pm

Team 3 Project

1 STEP 1: Topic/Domain Selection

Selection of the domain you will be working on. Proper study and knowledge of the domain is necessary.

Some of the outputs from this step is listed below (but not limited to this):

• Selected Topic/Application: students need to come up with a topic or application or service. The best topic is the one that you have extensive knowledge about (i.e., related to your job, can gain knowledge through your family or friends). Also, if you have a business idea (e.g., startup business, mobile application, etc.) and have passion about that, it is a great idea to start working on that through this project and have a prototype at the end.

• Introduction: Students need to provide detailed explanation about the topic that they selected and

The Application chosen is Online Media Rentals. We are now living in a digital world where we can purchase almost anything online. One of the online purchases that can be made is movie rentals. Online Media Rentals is a business within the market of online movie rentals. It allows consumers to buy or rent movies online. Consumers can sit at the convenience of their homes and will be able to choose a variety of categories. Within the business, consumers can start by creating a membership online by providing their name and address. Their membership account will keep track of their fees and current balances. Upon creating an account, the consumer will have access to a variety of movie rentals within different genres, titles, and release dates. Upon choosing a rental, their account will advise on the fee, due date, return date, and any late fees.

• Advantages and contributions: Include all the advantages of having this database management system. At least mention 3 advantages.

1. Some advantages as a consumer is its convenience. Consumers can purchase a movie or TV show at any location. It eliminates the need to travel to the store to purchase a rental. Instead, consumers can purchase a movie at the comfort of their own homes.

2. It saves money as the consumer. A consumer can watch endless movies throughout the membership and lower costs per each movie.

3. It saves money as a business. It avoids the costs of maintaining building expenses such as utilities. It also saves on the costs of labor expenses.

• Uses cases: Include all industries/places/areas where this can be used or any business that have systems similar to your devised database.

There are numerous of businesses that use online media rentals which has made an impact in the business industry. Technology has expanded throughout the years and will continue to be a powerful source of revenue for businesses. Examples of such are: Amazon Prime Video, Netflix, Apple Tv, and FandangoNow. These media businesses are able to collect the user's information by collecting their email address, name, and payment information. Most of the movie rentals have different types of options for consumers to purchase from such as "no ads" or "monthly subscription." Consumers are able to successfully purchase or rent movies that fits their budget but also their preference in the type of movies they want to watch. Being in a subscription with online rentals has changed the way individuals watch movies due to them having more authority to choose their movie preferences. Fortunately, this database is essential and "home" to millions of people because it is at their convenience.

2 STEP 2: Conceptual Data Modeling and Database Design
All logical diagrams like ER diagrams, EER diagrams should be made in the Database Design. The output for
this step are listed below:
• List of Business Rules: Group members need to start by preparing a complete list of business rules relating to the selected topics.

- A membership can hold zero or many rentals
- A rental will belong to only one membership at a time

- Rentals will have one or many details
- The DetailRentals will have only one video
- The video will only show one movie
- The movie will have zero or one price.
- The cost of a movie will have none or many prices

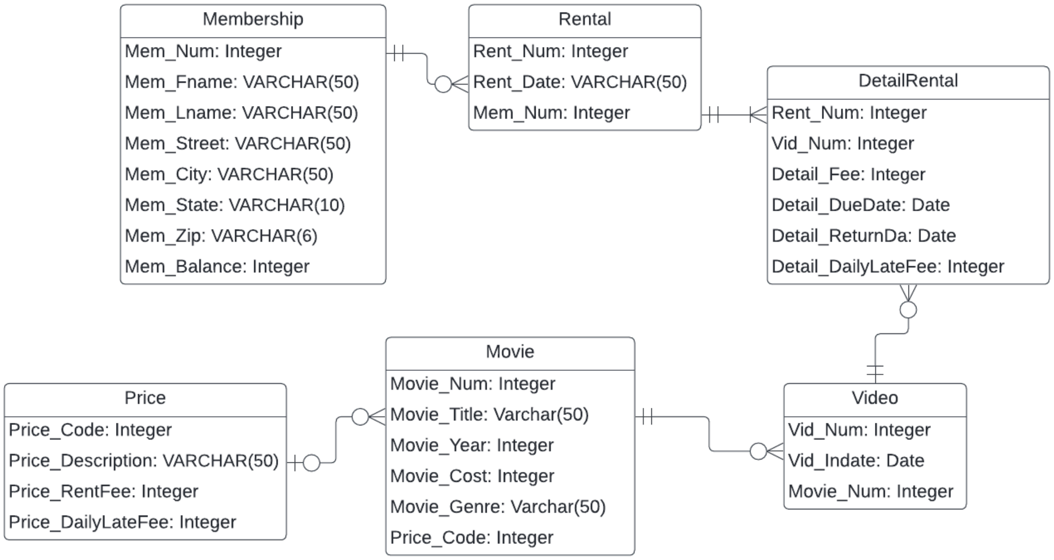
• Conceptual Data Modeling: ER/EER Diagram: Find the list of involved entities, the relationship among them and draw the entity-relationship (ER) diagram. Please don't forget to show the main concepts in the diagram such as supertype/subtype, cardinality, type of attributes, associative entity, etc.

• Transform ER/EER Model to Relational Model: Transform the developed ER/EER into a relational model using methods learned in the class. Please show all primary and foreign keys and references to other tables.

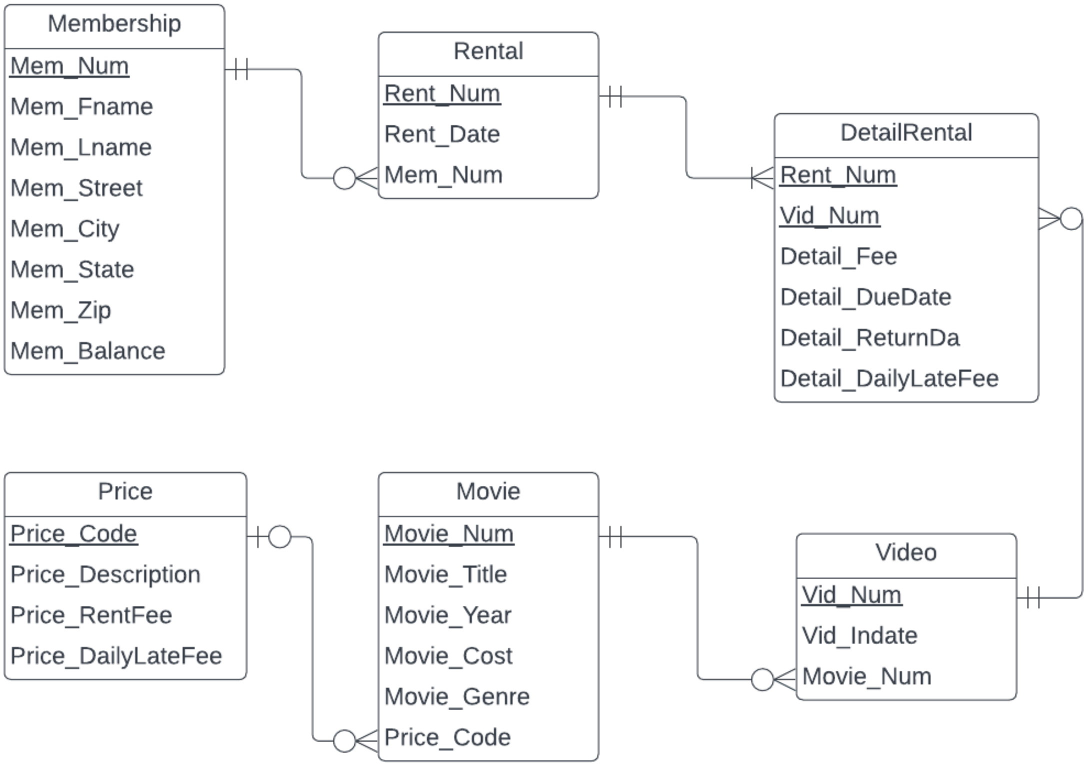
• Normalization: Check the tables (If necessary): Check the final relations (tables) for dependencies and perform normalization if required to have well-defined tables.

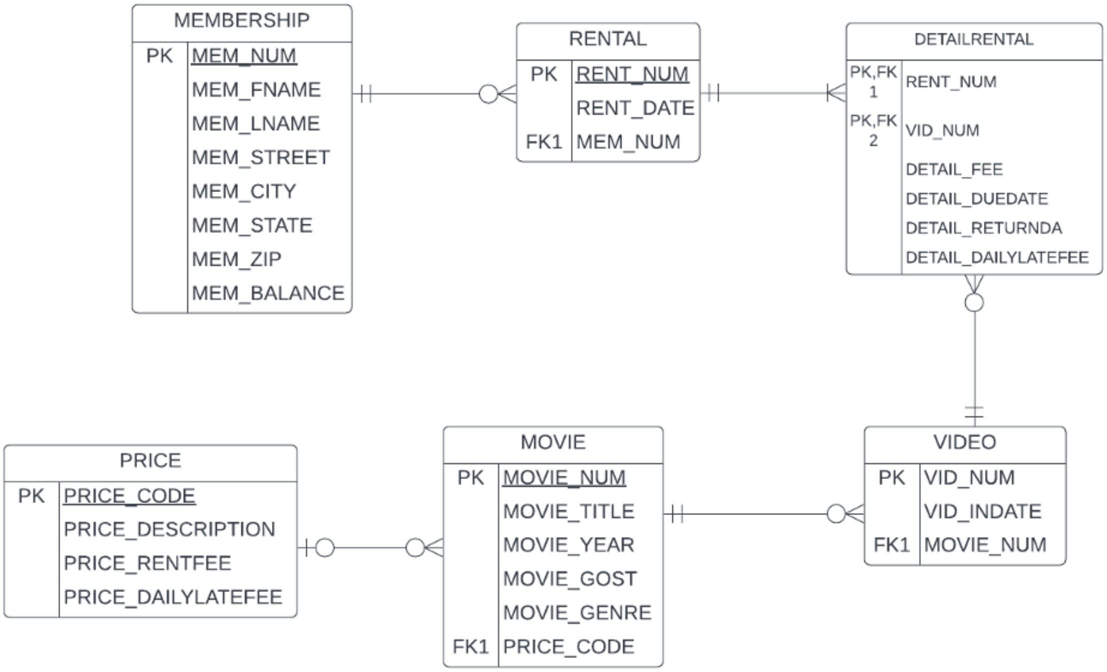
Answer/

ER Diagram



Relational Diagram





3 STEP 3: Database Implementation

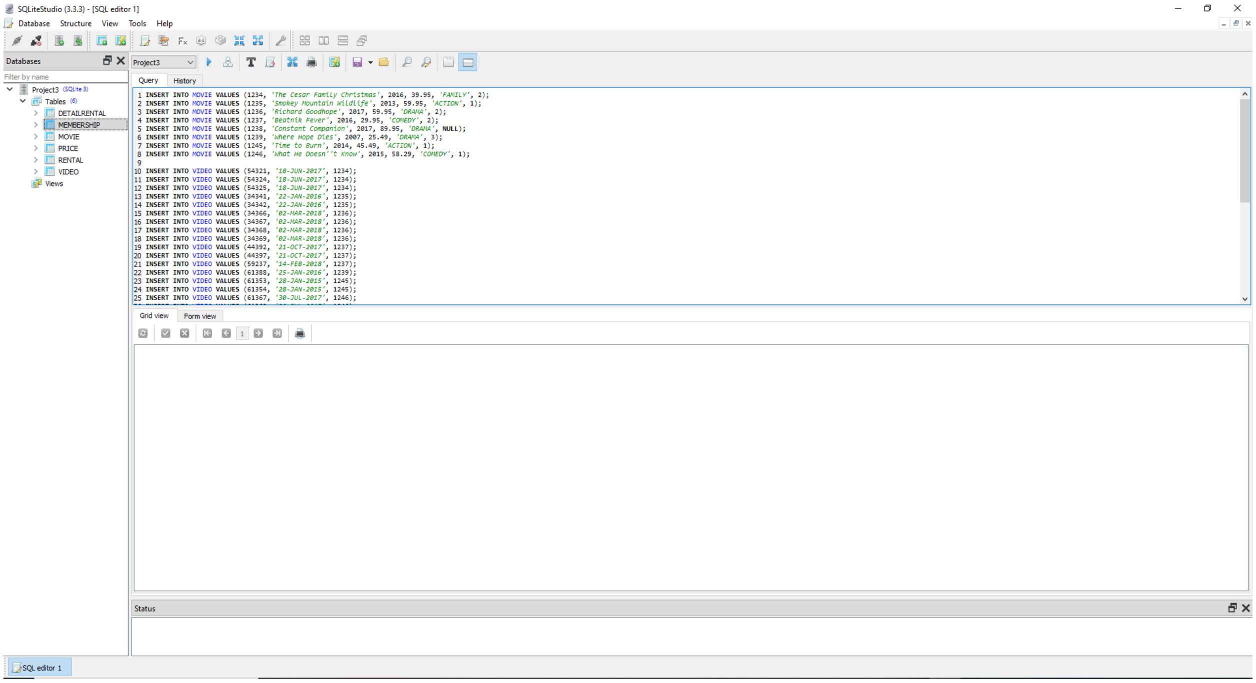
The output for this step are listed below:

- DDL SQL statements to create Tables, views, indexes, etc.
- Insert dummy (fake) data into tables
- Prepare analytic questions (single and multi table) and develop related SQL statements.

Answer/ Use SQLITE

```
CREATE TABLE MEMBERSHIP (
MEM_NUM INT NOT NULL,
MEM_FNAME VARCHAR(50),
MEM_LNAME VARCHAR(50),
MEM_STREET VARCHAR(50),
MEM_CITY VARCHAR(50),
MEM_STATE VARCHAR(10) DEFAULT 'GA',
MEM_ZIP VARCHAR(6),
MEM_BALANCE INT,
PRIMARY KEY(MEM_NUM)
)
```

```
CREATE TABLE MOVIE (
MOVIE_NUM INT NOT NULL,
MOVIE_TITLE VARCHAR(50),
MOVIE_YEAR INT,
MOVIE_COST INT,
MOVIE_GENRE VARCHAR(50),
PRICE_CODE INT,
PRIMARY KEY(MOVIE_NUM),
CONSTRAINT Check_Movie_Cost CHECK (MOVIE_COST > 0),
CONSTRAINT Movie_Title UNIQUE (MOVIE_TITLE),
CONSTRAINT Price_fk FOREIGN KEY (PRICE_CODE) REFERENCES PRICE
(PRICE_CODE)
)

CREATE TABLE RENTAL (
RENT_NUM INT NOT NULL,
RENT_DATE DATE,
MEM_NUM INT,
PRIMARY KEY(RENT_NUM),
FOREIGN KEY (MEM_NUM) REFERENCES MEMBERSHIP(MEM_NUM)
)

CREATE TABLE VIDEO (
VID_NUM INT NOT NULL,
VID_INDATE DATE,
MOVIE_NUM INT,
PRIMARY KEY(VID_NUM),
FOREIGN KEY (MOVIE_NUM) REFERENCES MOVIE(MOVIE_NUM)
)

CREATE TABLE DETAILRENTAL (
RENT_NUM INT NOT NULL,
VID_NUM INT NOT NULL,
DETAIL_FEE INT,
DETAIL_DUEDATE DATE NOT NULL,
DETAIL_RETURNDATE DATE,
DETAIL_DAILYLATEFEE INT,
PRIMARY KEY (RENT_NUM, VID_NUM),
FOREIGN KEY (RENT_NUM) REFERENCES RENTAL(RENT_NUM),
FOREIGN KEY (VID_NUM) REFERENCES VIDEO(VID_NUM)
)

CREATE TABLE PRICE (
PRICE_CODE INT NOT NULL,
PRICE_DESCRIPTION VARCHAR(50),
PRICE_RENTFEE INT NOT NULL,
PRICE_DAILYLATEFEE INT,
PRIMARY KEY(PRICE_CODE)
)

INSERT INTO MEMBERSHIP VALUES (102, 'TAMI', 'DAWSON', '2632 TAKLI CIRCLE',
'NORENE', 'TN', '37136', 11);
INSERT INTO MEMBERSHIP VALUES (103, 'CURT', 'KNIGHT', '4025 CORNELL COURT',
'FLATGAP', 'KY', '41219', 6);
INSERT INTO MEMBERSHIP VALUES (104, 'JAMAL', 'MELENDEZ', '788 EAST 145TH
AVENUE', 'QUEBECK', 'TN', '38579', 0);
INSERT INTO MEMBERSHIP VALUES (105, 'IVA', 'MCCLAIN', '6045 MUSKET BALL
CIRCLE', 'SUMMIT', 'KY', '42783', 15);
INSERT INTO MEMBERSHIP VALUES (106, 'MIRANDA', 'PARKS', '4469 MAXWELL
PLACE', 'GERMANTOWN', 'TN', '38183', 0);
INSERT INTO MEMBERSHIP VALUES (107, 'ROSARIO', 'ELLIOTT', '7578 DANNER
AVENUE', 'COLUMBIA', 'TN', '38402', 5);
INSERT INTO MEMBERSHIP VALUES (108, 'MATTIE', 'GUY', '4390 EVERGREEN
STREET', 'LILY', 'KY', '40740', 0);
INSERT INTO MEMBERSHIP VALUES (109, 'CLINT', 'OCHOA', '1711 ELM STREET',
'GREENEVILLE', 'TN', '37745', 10);
INSERT INTO MEMBERSHIP VALUES (110, 'LEWIS', 'ROSALES', '4524 SOUTHWIND
CIRCLE', 'COUNCE', 'TN', '38326', 0);
INSERT INTO MEMBERSHIP VALUES (111, 'STACY', 'MANN', '2789 EAST COOK
AVENUE', 'MURFREESBORO', 'TN', '37132', 8);
INSERT INTO MEMBERSHIP VALUES (112, 'LUIS', 'TRUJILLO', '7267 MELVIN
AVENUE', 'HEISKELL', 'TN', '37754', 3);
INSERT INTO MEMBERSHIP VALUES (113, 'MINNIE', 'GONZALES', '6430 VASILI
DRIVE', 'WILLISTON', 'TN', '38076', 0);



INSERT INTO RENTAL VALUES (1001, '01-MAR-2018', 103);
INSERT INTO RENTAL VALUES (1002, '01-MAR-2018', 105);
INSERT INTO RENTAL VALUES (1003, '02-MAR-2018', 102);
INSERT INTO RENTAL VALUES (1004, '02-MAR-2018', 110);
INSERT INTO RENTAL VALUES (1005, '02-MAR-2018', 111);
INSERT INTO RENTAL VALUES (1006, '02-MAR-2018', 107);
INSERT INTO RENTAL VALUES (1007, '02-MAR-2018', 104);
INSERT INTO RENTAL VALUES (1008, '03-MAR-2018', 105);
INSERT INTO RENTAL VALUES (1009, '03-MAR-2018', 111);

INSERT INTO PRICE VALUES (1, 'Standard', 2, 1);
INSERT INTO PRICE VALUES (2, 'New Release', 3.5, 3);
INSERT INTO PRICE VALUES (3, 'Discount', 1.5, 1);
INSERT INTO PRICE VALUES (4, 'Weekly Special', 1, .5);

INSERT INTO MOVIE VALUES (1234, 'The Cesar Family Christmas', 2016, 39.95, 'FAMILY', 2);
INSERT INTO MOVIE VALUES (1235, 'Smokey Mountain Wildlife', 2013, 59.95, 'ACTION', 1);
INSERT INTO MOVIE VALUES (1236, 'Richard Goodhope', 2017, 59.95, 'DRAMA', 2);
INSERT INTO MOVIE VALUES (1237, 'Beatnik Fever', 2016, 29.95, 'COMEDY', 2);
INSERT INTO MOVIE VALUES (1238, 'Constant Companion', 2017, 89.95, 'DRAMA', NULL);
INSERT INTO MOVIE VALUES (1239, 'Where Hope Dies', 2007, 25.49, 'DRAMA', 3);
INSERT INTO MOVIE VALUES (1245, 'Time to Burn', 2014, 45.49, 'ACTION', 1);
INSERT INTO MOVIE VALUES (1246, 'What He Doesn''t Know', 2015, 58.29, 'COMEDY', 1);

INSERT INTO VIDEO VALUES (54321, '18-JUN-2017', 1234);
INSERT INTO VIDEO VALUES (54324, '18-JUN-2017', 1234);
INSERT INTO VIDEO VALUES (54325, '18-JUN-2017', 1234);
INSERT INTO VIDEO VALUES (34341, '22-JAN-2016', 1235);
INSERT INTO VIDEO VALUES (34342, '22-JAN-2016', 1235);
INSERT INTO VIDEO VALUES (34366, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (34367, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (34368, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (34369, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (44392, '21-OCT-2017', 1237);
INSERT INTO VIDEO VALUES (44397, '21-OCT-2017', 1237);
INSERT INTO VIDEO VALUES (59237, '14-FEB-2018', 1237);
INSERT INTO VIDEO VALUES (61388, '25-JAN-2016', 1239);
INSERT INTO VIDEO VALUES (61353, '28-JAN-2015', 1245);
INSERT INTO VIDEO VALUES (61354, '28-JAN-2015', 1245);
INSERT INTO VIDEO VALUES (61367, '30-JUL-2017', 1246);
INSERT INTO VIDEO VALUES (61369, '30-JUL-2017', 1246);

INSERT INTO DETAILRENTAL VALUES (1001, 34342, 2, '04-MAR-2018', '02-MAR-2018', NULL);
INSERT INTO DETAILRENTAL VALUES (1001, 34366, 3.5, '04-MAR-2018', '02-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1001, 61353, 2, '04-MAR-2018', '03-MAR-2018', 1);
INSERT INTO DETAILRENTAL VALUES (1002, 59237, 3.5, '04-MAR-2018', '04-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1003, 54325, 3.5, '04-MAR-2018', '09-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1003, 61369, 2, '06-MAR-2018', '09-MAR-2018', 1);
INSERT INTO DETAILRENTAL VALUES (1003, 61388, 0, '06-MAR-2018', '09-MAR-2018', 1);
INSERT INTO DETAILRENTAL VALUES (1004, 34341, 2, '07-MAR-2018', '07-MAR-2018', 1);
INSERT INTO DETAILRENTAL VALUES (1004, 34367, 3.5, '05-MAR-2018', '07-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1004, 44392, 3.5, '05-MAR-2018', '07-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1005, 34342, 2, '07-MAR-2018', '05-MAR-2018', 1);
INSERT INTO DETAILRENTAL VALUES (1005, 44397, 3.5, '05-MAR-2018', '05-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1006, 34366, 3.5, '05-MAR-2018', '04-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1006, 61367, 2, '07-MAR-2018', NULL, 1);
INSERT INTO DETAILRENTAL VALUES (1007, 34368, 3.5, '05-MAR-2018', NULL, 3);
INSERT INTO DETAILRENTAL VALUES (1008, 34369, 3.5, '05-MAR-2018', '05-MAR-2018', 3);
INSERT INTO DETAILRENTAL VALUES (1009, 54324, 3.5, '05-MAR-2018', NULL, 3);
```

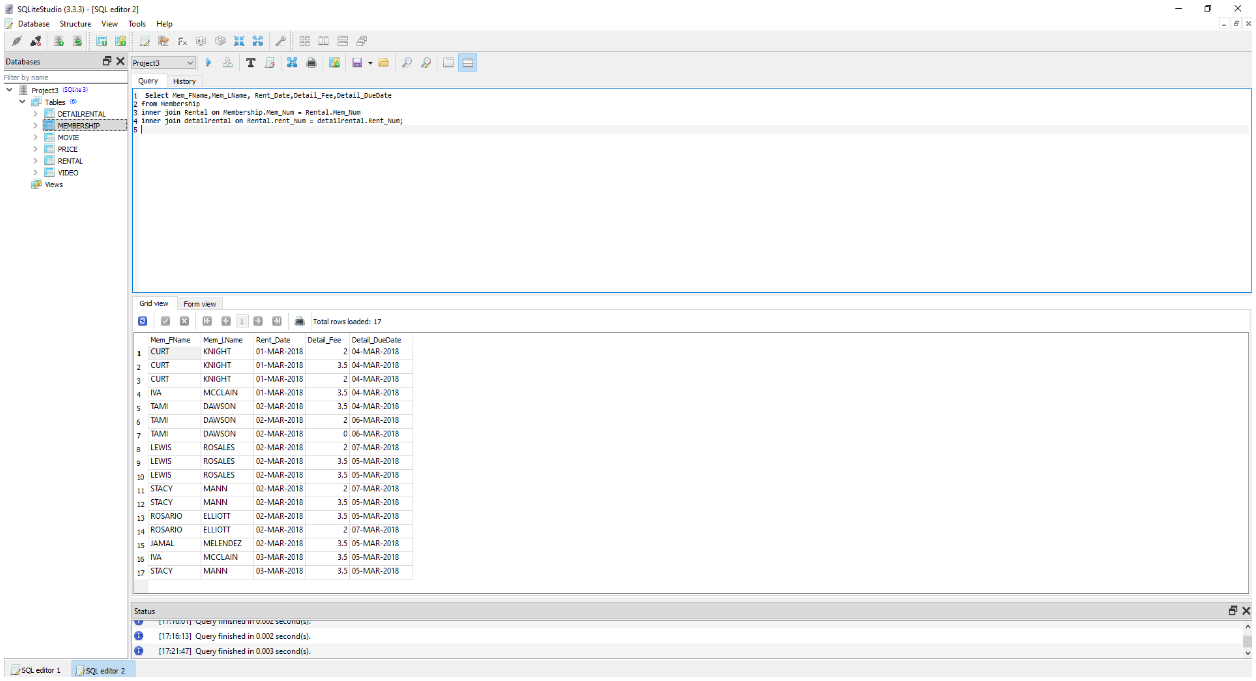


```sql
INSERT INTO MOVIE VALUES (1234, 'The Cesar Family Christmas', 2016, 39.95, 'FAMILY', 2);
INSERT INTO MOVIE VALUES (1235, 'Smokey Mountain Wildlife', 2013, 59.95, 'ACTION', 1);
INSERT INTO MOVIE VALUES (1236, 'Richard Goodhope', 2017, 59.95, 'DRAMA', 2);
INSERT INTO MOVIE VALUES (1237, 'Beatnik Fever', 2016, 29.95, 'COMEDY', 2);
INSERT INTO MOVIE VALUES (1238, 'Constant Companion', 2017, 89.95, 'DRAMA', NULL);
INSERT INTO MOVIE VALUES (1239, 'Where Hope Dies', 2007, 25.49, 'DRAMA', 3);
INSERT INTO MOVIE VALUES (1245, 'Time to Burn', 2014, 45.49, 'ACTION', 1);
INSERT INTO MOVIE VALUES (1246, 'What He Doesn''t Know', 2015, 58.29, 'COMEDY', 1);

INSERT INTO VIDEO VALUES (54321, '18-JUN-2017', 1234);
INSERT INTO VIDEO VALUES (54324, '18-JUN-2017', 1234);
INSERT INTO VIDEO VALUES (54325, '18-JUN-2017', 1234);
INSERT INTO VIDEO VALUES (34341, '22-JAN-2016', 1235);
INSERT INTO VIDEO VALUES (34342, '22-JAN-2016', 1235);
INSERT INTO VIDEO VALUES (34366, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (34367, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (34368, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (34369, '02-MAR-2018', 1236);
INSERT INTO VIDEO VALUES (44392, '21-OCT-2017', 1237);
INSERT INTO VIDEO VALUES (44397, '21-OCT-2017', 1237);
INSERT INTO VIDEO VALUES (59237, '14-FEB-2018', 1237);
INSERT INTO VIDEO VALUES (61388, '25-JAN-2016', 1239);
INSERT INTO VIDEO VALUES (61353, '28-JAN-2015', 1245);
INSERT INTO VIDEO VALUES (61354, '28-JAN-2015', 1245);
INSERT INTO VIDEO VALUES (61367, '30-JUL-2017', 1246);
```

- Prepare analytic questions (single and multi table) and develop related SQL statements.

1- List the price description and rent fee for all of the different price types?

```
Select Mem_FName,Mem_LName, Rent_Date,Detail_Fee,Detail_DueDate
from Membership
inner join Rental on Membership.Mem_Num = Rental.Mem_Num
inner join detailrental on Rental.rent_Num = detailrental.Rent_Num;
```



```
 Select Mem_FName,Mem_LName, Rent_Date,Detail_Fee,Detail_DueDate
from Membership
inner join Rental on Membership.Mem_Num = Rental.Mem_Num
inner join detailrental on Rental.rent_Num = detailrental.Rent_Num;
```

Total rows loaded: 17

	Mem_FName	Mem_LName	Rent_Date	Detail_Fee	Detail_DueDate
1	CURT	KNIGHT	01-MAR-2018	2	04-MAR-2018
2	CURT	KNIGHT	01-MAR-2018	3.5	04-MAR-2018
3	CURT	KNIGHT	01-MAR-2018	2	04-MAR-2018
4	IVA	MCCLAIN	01-MAR-2018	3.5	04-MAR-2018
5	TAMI	DAWSON	02-MAR-2018	3.5	04-MAR-2018
6	TAMI	DAWSON	02-MAR-2018	2	06-MAR-2018
7	TAMI	DAWSON	02-MAR-2018	0	06-MAR-2018
8	LEWIS	ROSALES	02-MAR-2018	2	07-MAR-2018
9	LEWIS	ROSALES	02-MAR-2018	3.5	05-MAR-2018
10	LEWIS	ROSALES	02-MAR-2018	3.5	05-MAR-2018
11	STACY	MANN	02-MAR-2018	2	07-MAR-2018
12	STACY	MANN	02-MAR-2018	3.5	05-MAR-2018
13	ROSARIO	ELLIOTT	02-MAR-2018	3.5	05-MAR-2018
14	ROSARIO	ELLIOTT	02-MAR-2018	2	07-MAR-2018
15	JAMAL	MELENDEZ	02-MAR-2018	3.5	05-MAR-2018
16	IVA	MCCLAIN	03-MAR-2018	3.5	05-MAR-2018
17	STACY	MANN	03-MAR-2018	3.5	05-MAR-2018

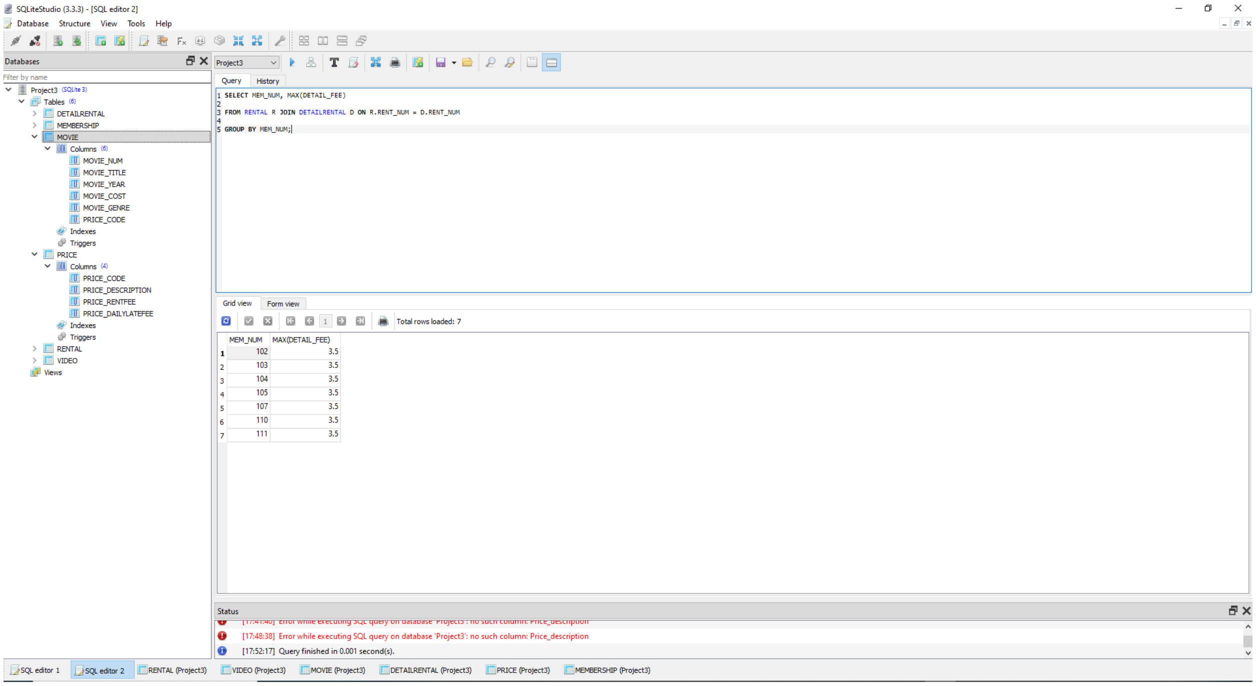
```
[17:16:01] Query finished in 0.002 second(s).
[17:16:13] Query finished in 0.002 second(s).
[17:21:47] Query finished in 0.003 second(s).
```

2- What was the highest fee paid by each member (report last name and fee). (use Inner Join, functions MIN Max and Group By)

SELECT MEM_NUM, MAX(DETAIL_FEE)

FROM RENTAL R JOIN DETAILRENTAL D ON R.RENT_NUM = D.RENT_NUM

GROUP BY MEM_NUM;



```
SELECT MEM_NUM, MAX(DETAIL_FEE)

FROM RENTAL R JOIN DETAILRENTAL D ON R.RENT_NUM = D.RENT_NUM

GROUP BY MEM_NUM;
```

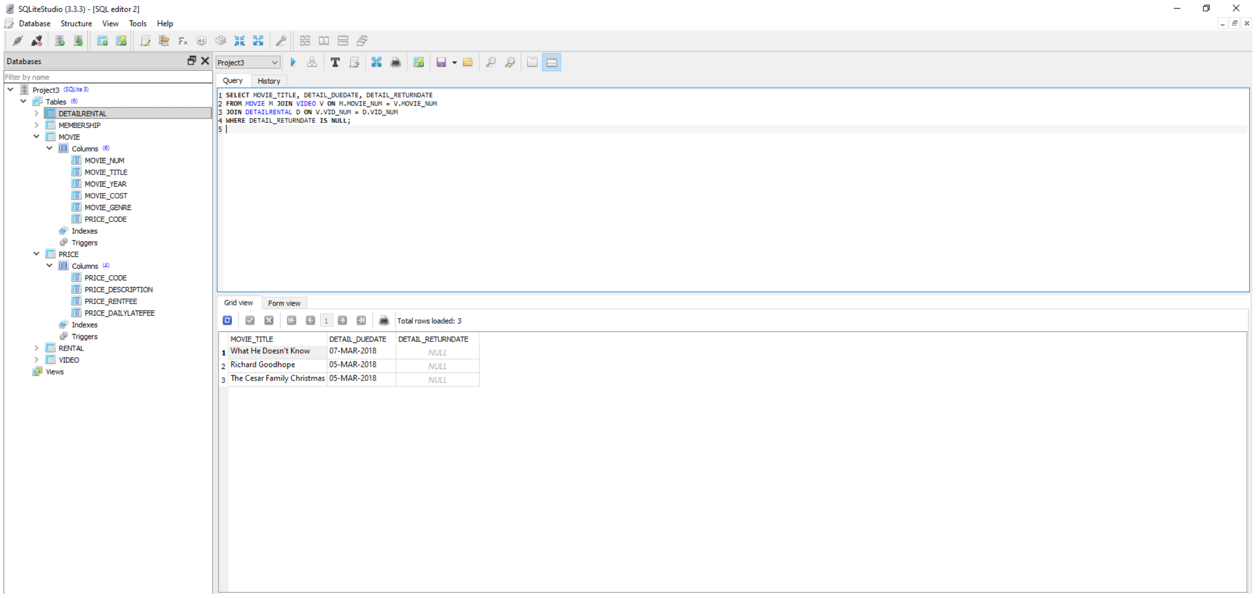
Total rows loaded: 7

	MEM_NUM	MAX(DETAIL_FEE)
1	102	3.5
2	103	3.5
3	104	3.5
4	105	3.5
5	107	3.5
6	110	3.5
7	111	3.5

```
[17:41:40] Error while executing SQL query on database 'Project3': no such column: Price_description
[17:48:38] Error while executing SQL query on database 'Project3': no such column: Price_description
[17:52:17] Query finished in 0.001 second(s).
```

3- What are the movie titles, due dates and return dates for movies that have not been returned? (Null Values)

```
SELECT MOVIE_TITLE, DETAIL_DUEDATE, DETAIL_RETURNDATE
FROM MOVIE M JOIN VIDEO V ON M.MOVIE_NUM = V.MOVIE_NUM
JOIN DETAILRENTAL D ON V.VID_NUM = D.VID_NUM
WHERE DETAIL_RETURNDATE IS NULL;
```



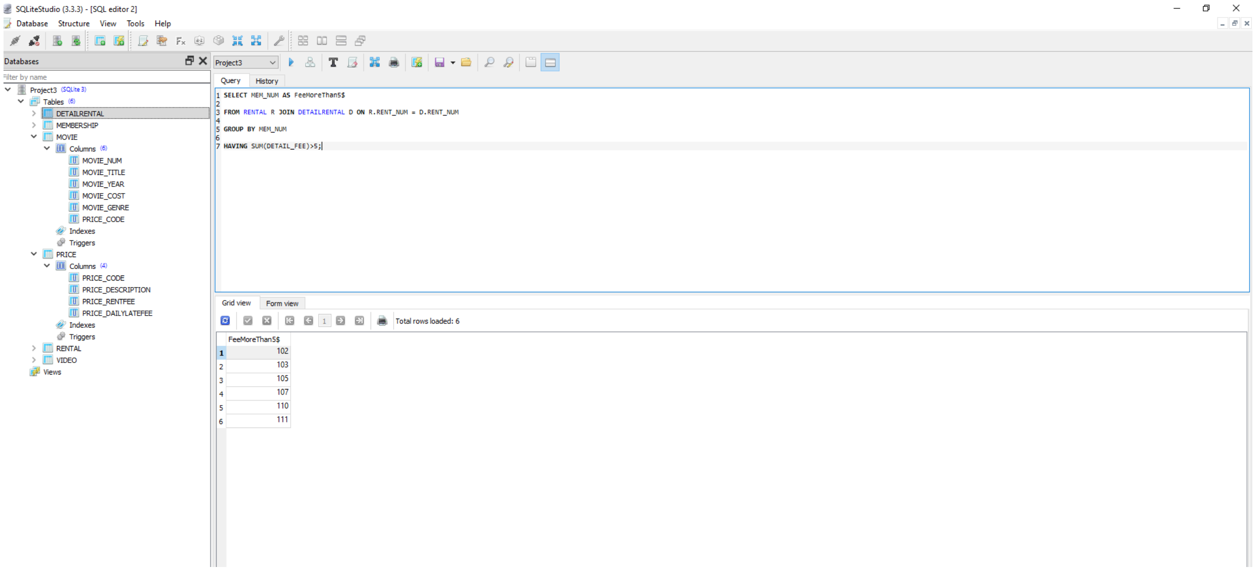
```
SELECT MOVIE_TITLE, DETAIL_DUEDATE, DETAIL_RETURNDATE
FROM MOVIE M JOIN VIDEO V ON M.MOVIE_NUM = V.MOVIE_NUM
JOIN DETAILRENTAL D ON V.VID_NUM = D.VID_NUM
WHERE DETAIL_RETURNDATE IS NULL;
```

Total rows loaded: 3

	MOVIE_TITLE	DETAIL_DUEDATE	DETAIL_RETURNDATE
1	What He Doesn't Know	07-MAR-2018	NULL
2	Richard Goodhope	05-MAR-2018	NULL
3	The Cesar Family Christmas	05-MAR-2018	NULL

4- What were the total rental fees for members (report Mem_num) who total fees were greater than $5 - change the output label to “FeesMoreThan$5’? (use Inner Join, and SQL Functions count, Avg, Sum & SQL Group By & SQL Having)

```
SELECT MEM_NUM AS FeeMoreThan5$
FROM RENTAL R JOIN DETAILRENTAL D ON R.RENT_NUM = D.RENT_NUM
GROUP BY MEM_NUM
HAVING SUM(DETAIL_FEE)>5;
```



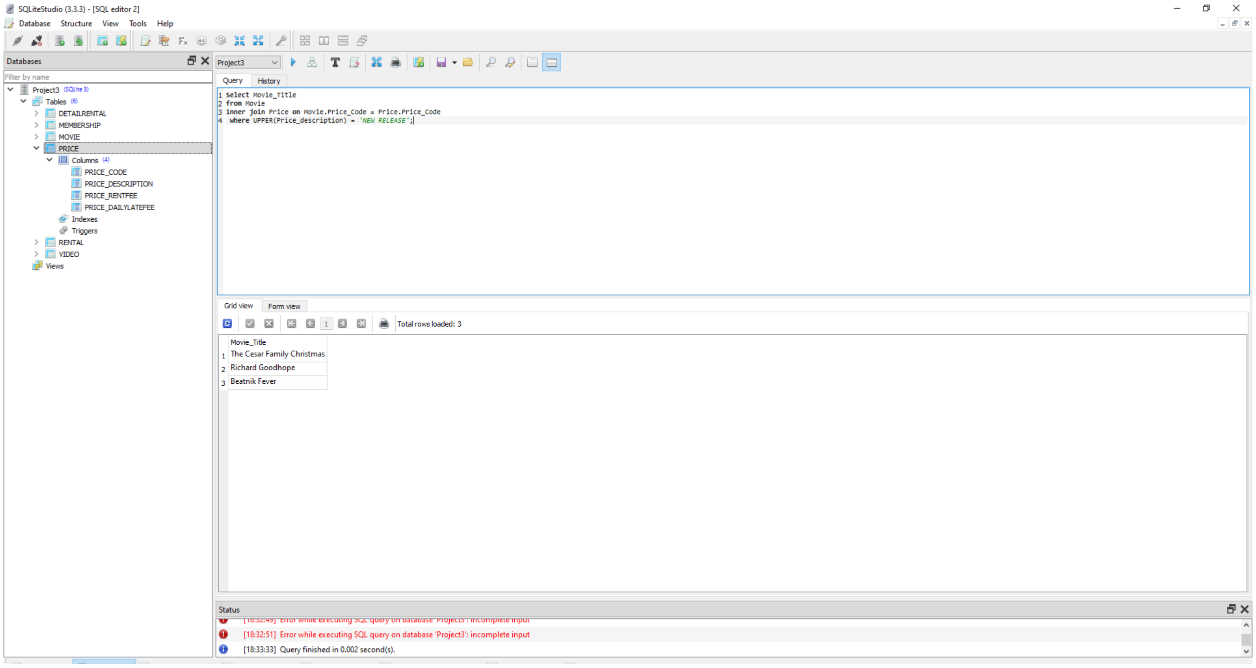
```
SELECT MEM_NUM AS FeeMoreThan5$

FROM RENTAL R JOIN DETAILRENTAL D ON R.RENT_NUM = D.RENT_NUM

GROUP BY MEM_NUM

HAVING SUM(DETAIL_FEE)>5;
```

Total rows loaded: 6

	FeeMoreThan5$
1	102
2	103
3	105
4	107
5	110
6	111

5- List the movie title for all new releases (use SQL Inner Join and SQL Where)?

Select Movie_Title
from Movie
inner join Price on Movie.Price_Code = Price.Price_Code
where UPPER(Price_description) = 'NEW RELEASE';



```
Select Movie_Title
from Movie
inner join Price on Movie.Price_Code = Price.Price_Code
 where UPPER(Price_description) = 'NEW RELEASE';
```

Total rows loaded: 3

	Movie_Title
1	The Cesar Family Christmas
2	Richard Goodhope
3	Beatnik Fever

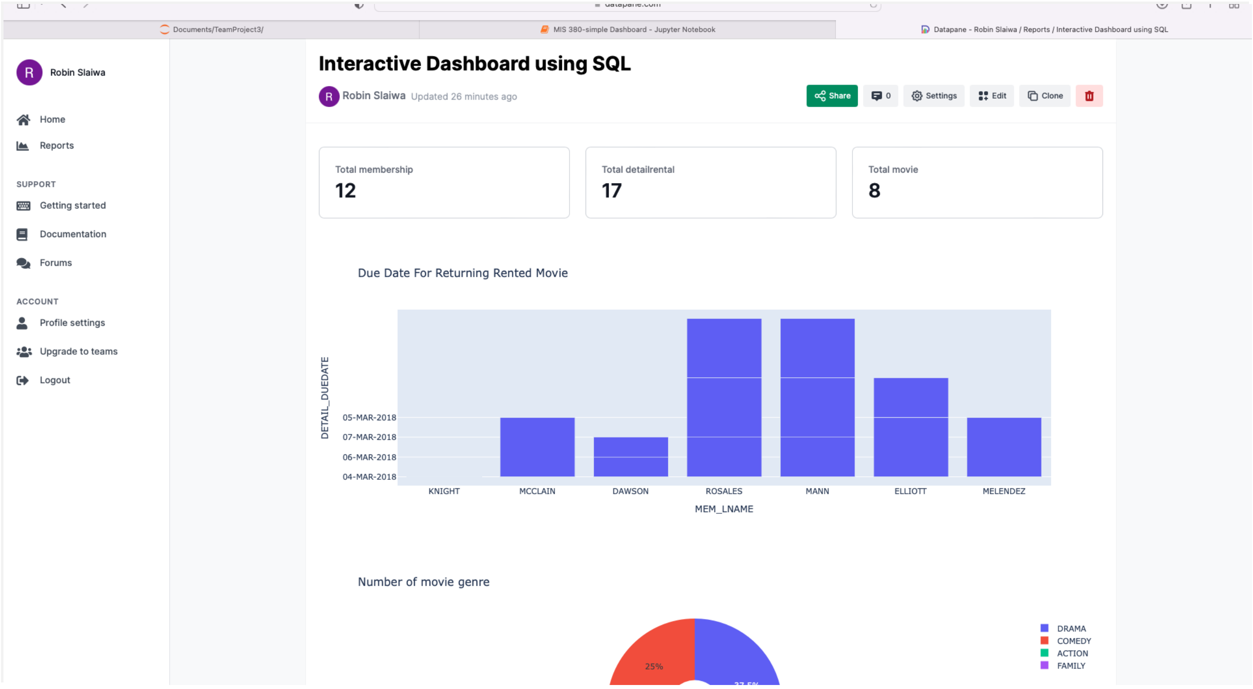
```
[18:32:49] Error while executing SQL query on database 'Project3': incomplete input
[18:32:51] Error while executing SQL query on database 'Project3': incomplete input
[18:33:33] Query finished in 0.002 second(s).
```

4 STEP 4: Enterprise (web) Database Dashboard

In this step, you will create a simple dashboard to illustrate the required analytics (e.g., from the previous

step). A sample dashboard for the music store database is available here (Sample Dashboard (click here)).
For this step, you can find and use software, online services, or programming codes based on your knowl-
edge. Otherwise, you can follow the instructions and use the template python/sql codes and modify them
accordingly to create the dashboard.

https://datapane.com/reports/43grvOA/interactive-dashboard-using-sql/



Interactive Dashboard using SQL

Robin Slaiwa Updated 26 minutes ago

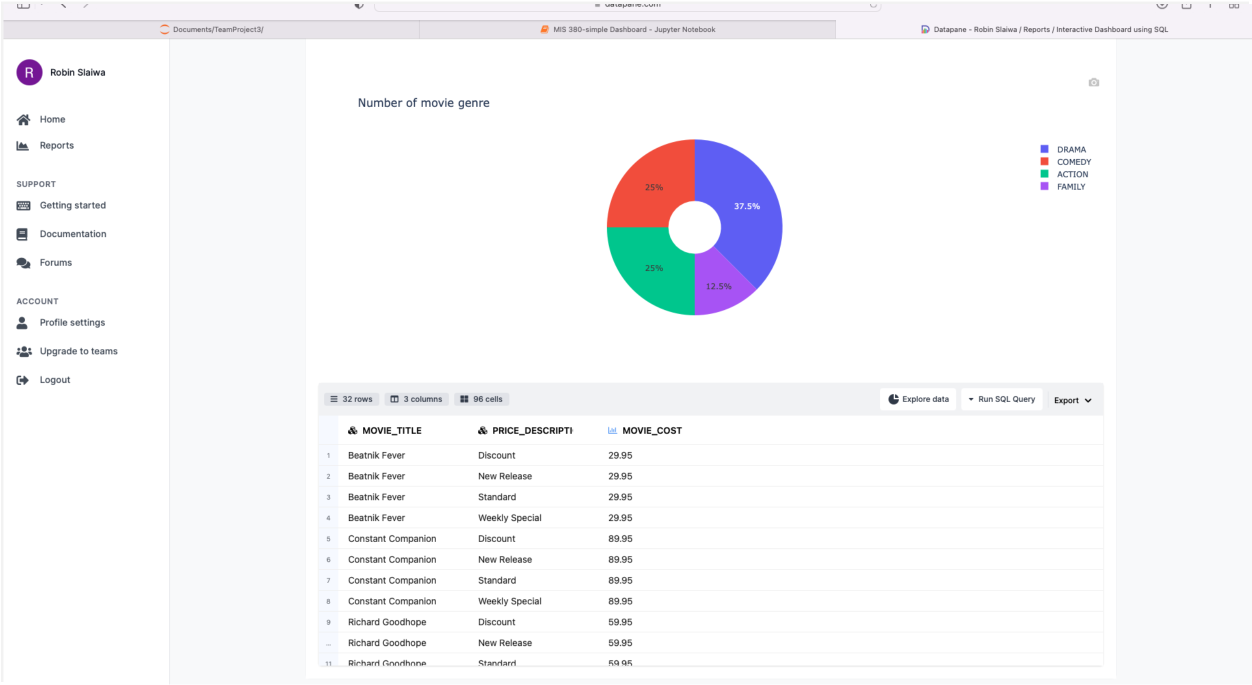
Total membership
12

Total detailrental
17

Total movie
8

Due Date For Returning Rented Movie

Number of movie genre



Number of movie genre

	MOVIE_TITLE	PRICE_DESCRIPTI	MOVIE_COST
1	Beatnik Fever	Discount	29.95
2	Beatnik Fever	New Release	29.95
3	Beatnik Fever	Standard	29.95
4	Beatnik Fever	Weekly Special	29.95
5	Constant Companion	Discount	89.95
6	Constant Companion	New Release	89.95
7	Constant Companion	Standard	89.95
8	Constant Companion	Weekly Special	89.95
9	Richard Goodhope	Discount	59.95
…	Richard Goodhope	New Release	59.95
11	Richard Goodhope	Standard	59.95