

Description of the application domain:

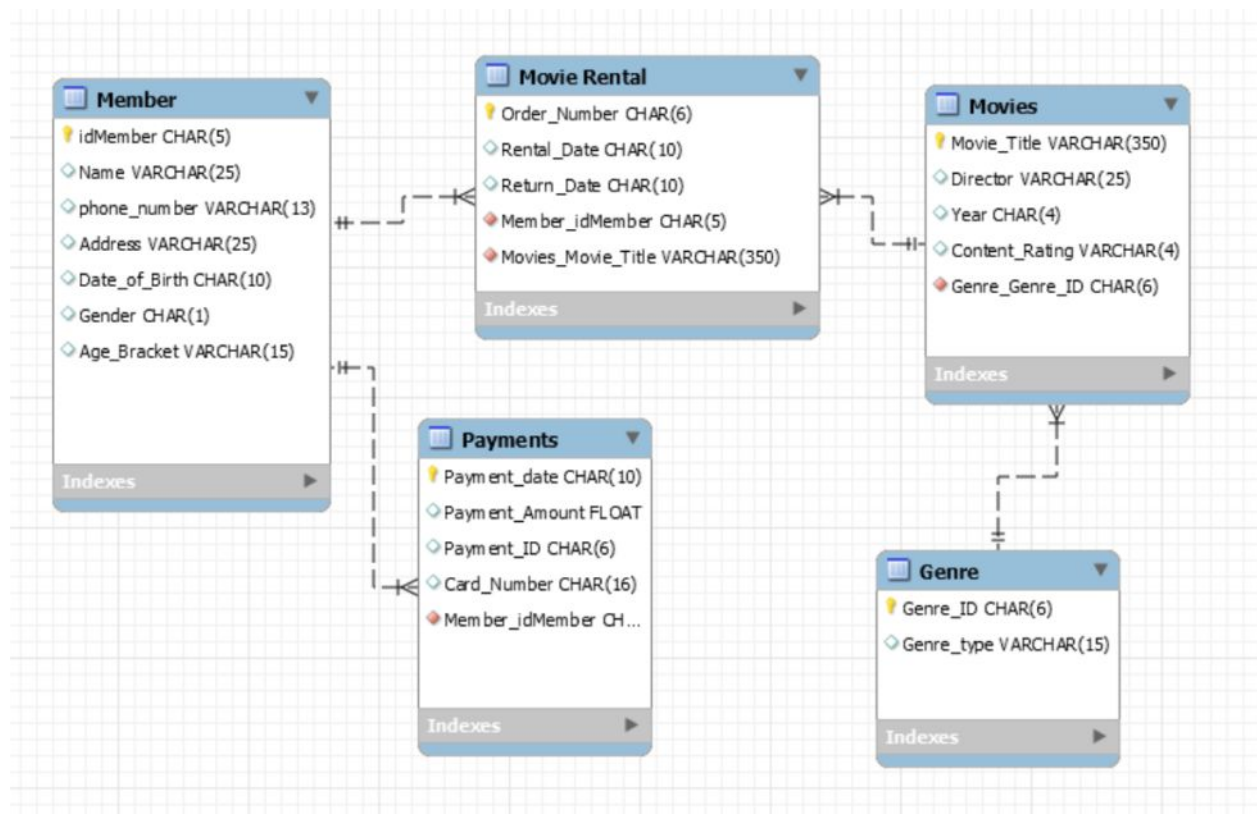
Retro Rentals (RR) is a small locally owned video rental store that focuses on bringing back the nostalgia of video rentals. RR rents out an unlimited range of films from old classics to new blockbuster hits, members are able to rent multiple films without having to return the previous rental. RR customers consist of individuals in their local community. As RR is operating in the modern era, it runs off a 30 dollar monthly subscription where customers are required to sign up for a membership. Each membership is given a unique ID that is a combination of 5 numbers (Ex. 12345 or 43421 and requires things such as a member's name, address, phone number, date of birth, and gender. Payments are made on a monthly basis and are connected to the members unique ID, if payment methods are needed to be changed an employee should be able to without a hassle, a date and amount should be visible along with a payment ID number (combination of 5 numbers with a P in the beginning ex P12345) when looking at a members payment. In order to rent a movie a member is given an order number (which is connected to a unique order number that is a combination of 5 random numbers with an R in front ex. R12345 or R94321) that shows the date of the rental along with what movie(s) they've rented, when members return a film employees should be able to see the return date when searching for the order number. Although rentals are made in person, members are able to view the catalog of films on the stores website and when searching for a movie members should be able to see basic information about the film they are renting such as the title, director, year, content rating and genre id (Genre ID are made up of 4 numbers with the first two letters of the genre in front ex. A comedy film may have a genre ID of CO1234) it belongs to. If members are not sure about what they want to watch they should be able to look under film genres to find a list of films that may belong to a genre they may prefer to view from. Note films will only have one specific genre but a genre may have multiple films.

Reporting Requirements:

1. List of all Members (name, address, phone number, Member ID, DOB, Gender, and age bracket.
2. What are the most popular films based on a member's gender and how many rentals were there of this film? (Most rented film based of gender)
3. What is the most popular genre being rented and how many rentals are there?

Conceptual model:

ER DIAGRAM



Data Dictionary:

Member Information:

A member is a paying consumer at Retro Rentals. Members are required to give specific information about themselves that are kept within the stores system.

Member Information has the following attributes.

Member_ID: a fixed-length character string that uniquely identifies a member at Retro Rentals. This character string has a length of 5 characters and consists of decimal digits. Valid values for Member_ID range from '00000' through '99999' and these values are assigned by Retro Rentals when members sign up for a subscription.

Name: a variable-length character string that represents the first and last names of the faculty. This string can be a maximum of 25 characters. Every member must have a name.

phone_number: - a variable-length character string of up to 12 characters that contains the phone number of the member in the form of 'XXX-XXX-XXXX'. Every member must have a phone number.

Address - a variable-length character string of up to 25 characters that contains the home address of the member. Every member must have an address.

Date_of_Birth: a fixed-length character string representing the date of birth of a member. This string has a length of 10 characters of the form 'yyyy-mm-dd'. Every member must have a DOB.

Gender: a fixed-length character string that uniquely identifies the gender of a member. This character string has a length of 1 character. Valid characters for gender are 'F' and 'M'. Every member must have a gender.

Age_bracket: a variable-length character string of up to 15 characters that contains the age bracket of the member. Every member must have an age bracket.

Payments

Payments are what keep Retro Rentals alive. Payments are made on a monthly basis based upon the members sign up date and are attached to a member's ID. Payments are fixed to the stores monthly cost and are paid with a required Credit/Debit Card.

Payment_date: fixed-length character string representing the date of payment. This string has a length of 10 characters of the form 'yyyy-mm-dd'. All payments have a payment date.

Payment_Amount: a float that indicated the amount being billed to members. All payments have a payment amount.

Payment_ID: a fixed-length character string that uniquely identifies a payment at Retro Rentals. This character string has a length of 6 characters and consists of 1 string ('P') and 5 decimal digits. Valid decimal values for Payment_ID range from '00000' through '99999'. All payments have a payment ID

Card_Number: a fixed-length character string that uniquely identifies a card at Retro Rentals. This character string has a length of 16 characters and consists of 16 decimal digits. Valid

decimal values for Card_Numberrange from '0000000000000000' through '9999999999999999'.
All payments have a card number

Member_ID: a fixed-length character string that uniquely identifies a member at Retro Rentals.
This character string has a length of 5 characters and consists of decimal digits. Valid values for Member_ID range from '00000' through '99999' and these values are assigned by Retro Rentals when members sign up for a subscription. All payments have a member id

Movie Rental:

A movie rental is used to keep track of what members are renting and returning. Movie rentals are assigned to a members id and have an unlimited limit on how many movies can be rented.

Order_Number: a fixed-length character string that uniquely identifies an order at Retro Rentals.
This character string has a length of 6 characters and consists of 1 string ('R') and 5 decimal digits. Valid decimal values for Order_Number range from '00000' through '99999'. All Rental orders have a Order number

Member_ID: a fixed-length character string that uniquely identifies a member at Retro Rentals.
This character string has a length of 5 characters and consists of decimal digits. Valid values for Member_ID range from '00000' through '99999' and these values are assigned by Retro Rentals when members sign up for a subscription. All Rental orders have a member id.

Rental_Date: fixed-length character string representing the date of payment. This string has a length of 10 characters of the form 'yyyy-mm-dd'. All orders have a rental date.

Return_Date: fixed-length character string representing the date of payment. This string has a length of 10 characters of the form 'yyyy-mm-dd'. All orders may not have a return date.

Movie_Title: Name: a variable-length character string that represents the title of the film. This string can be a maximum of 350 characters. Every order must have a movie title.

Movies:

Movie_Title: Name: a variable-length character string that represents the title of the film. This string can be a maximum of 350 characters. Every order must have a movie title.

Director: a variable-length character string that represents the first and last names of the Director. This string can be a maximum of 25 characters. Every Movie must have a director.

Year: fixed-length character string that identifies the film's year of release. This character string has a length of 4 characters and consists of decimal digits. Valid decimal values for Year range from '1900' through '2100'. Every Movie must have a year.

Content_Rating: a variable-length character string of up to 4 characters that contains the content rating of a film. Valid values for content rating are 'G', 'PG', 'PG13', 'R'. Every Movie must have a content rating.

Genre_ID: a fixed-length character string that uniquely identifies a Genre at Retro Rentals. This character string has a length of 6 characters and consists of 2 strings (A-Z) and 4 decimal digits. Valid decimal values for Genre_ID range from '0000' through '9999'. Every Movie must have a genre id.

Genre:

Genres are used to keep track of what genre a movie belongs to. Films may have only one genre but a genre may have multiple films

Genre_ID: a fixed-length character string that uniquely identifies a Genre at Retro Rentals. This character string has a length of 6 characters and consists of 2 strings (A-Z) and 4 decimal digits. Valid decimal values for Genre_ID range from '0000' through '9999'. Every genre must have a genre id.

Genre_type: Name: a variable-length character string that represents the title of the film. This string can be a maximum of 15 characters. Every genre must have a genre type.

Member Information:

A member is a paying consumer at Retro Rentals. Members are required to give specific information about themselves that are kept within the stores system.

Member Information has the following attributes.

Member_ID: a fixed-length character string that uniquely identifies a member at Retro Rentals. This character string has a length of 5 characters and consists of decimal digits. Valid values for Member_ID range from '00000' through '99999' and these values are assigned by Retro Rentals when members sign up for a subscription.

Name: a variable-length character string that represents the first and last names of the faculty. This string can be a maximum of 25 characters. Every member must have a member ID.

phone_number: - a variable-length character string of up to 12 characters that contains the phone number of the member in the form of 'XXX-XXX-XXXX'. Every member must have a phone number.

Address - a variable-length character string of up to 25 characters that contains the home address of the member. Every member must have an address.

Date_of_Birth: a fixed-length character string representing the date of birth of a member. This string has a length of 10 characters of the form 'yyyy-mm-dd'. Every member must have a DOB.

Gender: a fixed-length character string that uniquely identifies the gender of a member. This character string has a length of 1 character. Valid characters for gender are 'F' and 'M'. Every member must have a gender.

Age_bracket: a variable-length character string of up to 15 characters that contains the age bracket of the member. Every member must have an age bracket.

Payments

Payments are what keep Retro Rentals alive. Payments are made on a monthly basis based upon the members sign up date and are attached to a member's ID. Payments are fixed to the stores monthly cost and are paid with a required Credit/Debit Card.

Payment_date: fixed-length character string representing the date of payment. This string has a length of 10 characters of the form 'yyyy-mm-dd'. All payments have a payment date.

Payment_Amount: a float that indicated the amount being billed to members. All payments have a payment amount.

Payment_ID: a fixed-length character string that uniquely identifies a payment at Retro Rentals. This character string has a length of 6 characters and consists of 1 string ('P') and 5 decimal digits. Valid decimal values for Payment_ID range from '00000' through '99999'. All payments have a payment ID

Card_Number: a fixed-length character string that uniquely identifies a card at Retro Rentals. This character string has a length of 16 characters and consists of 16 decimal digits. Valid decimal values for Card_Numberrange from '0000000000000000' through '9999999999999999'. All payments have a card number

Member_ID: a fixed-length character string that uniquely identifies a member at Retro Rentals. This character string has a length of 5 characters and consists of decimal digits. Valid values for Member_ID range from '00000' through '99999' and these values are assigned by Retro Rentals when members sign up for a subscription. All payments have a member id

Movie Rental:

A movie rental is used to keep track of what members are renting and returning. Movie rentals are assigned to a members id and have an unlimited limit on how many movies can be rented.

Order_Number: a fixed-length character string that uniquely identifies an order at Retro Rentals. This character string has a length of 6 characters and consists of 1 string ('R') and 5 decimal digits. Valid decimal values for Order_Number range from '00000' through '99999'. All Rental orders have a Order number

Member_ID: a fixed-length character string that uniquely identifies a member at Retro Rentals. This character string has a length of 5 characters and consists of decimal digits. Valid values for Member_ID range from '00000' through '99999' and these values are assigned by Retro Rentals when members sign up for a subscription. All Rental orders have a member id.

Rental_Date: fixed-length character string representing the date of payment. This string has a length of 10 characters of the form 'yyyy-mm-dd'. All orders have a rental date.

Return_Date: fixed-length character string representing the date of payment. This string has a length of 10 characters of the form 'yyyy-mm-dd'. All orders may not have a return date.

Movie_Title: Name: a variable-length character string that represents the title of the film. This string can be a maximum of 350 characters. Every order must have a movie title.

Movies:

Movie_Title: Name: a variable-length character string that represents the title of the film. This string can be a maximum of 350 characters. Every order must have a movie title.

Director: a variable-length character string that represents the first and last names of the Director. This string can be a maximum of 25 characters. Every Movie must have a director.

Year: fixed-length character string that identifies the film's year of release. This character string has a length of 4 characters and consists of decimal digits. Valid decimal values for Year range from '1900' through '2100'. Every Movie must have a year.

Content_Rating: a variable-length character string of up to 4 characters that contains the content rating of a film. Valid values for content rating are 'G', 'PG', 'PG13', 'R'. Every Movie must have a content rating.

Genre_ID: a fixed-length character string that uniquely identifies a Genre at Retro Rentals. This character string has a length of 6 characters and consists of 2 strings (A-Z) and 4 decimal digits. Valid decimal values for Genre_ID range from '0000' through '9999'. Every Movie must have a genre id.

Genre:

Genres are used to keep track of what genre a movie belongs to. Films may have only one genre but a genre may have multiple films

Genre_ID: a fixed-length character string that uniquely identifies a Genre at Retro Rentals. This character string has a length of 6 characters and consists of 2 strings (A-Z) and 4 decimal digits. Valid decimal values for Genre_ID range from '0000' through '9999'. Every genre must have a genre id.

Genre_type: Name: a variable-length character string that represents the title of the film. This string can be a maximum of 15 characters. Every genre must have a genre type.

Logical model:

[Feedback](#) [Help](#) [nguyenst@my.easternct.edu](#)

Upload Script

Actions

+ Create Database Object

GENRE

Table

Status: Valid

Created 32 seconds ago

MEMBER

Table

Status: Valid

Created 32 seconds ago

MOVIES

Table

Status: Valid

Created 31 seconds ago

MOVIE_RENTAL

Table

Status: Valid

Created 29 seconds ago

PAYMENTS

Table

Status: Valid

Created 28 seconds ago

Data Definition Statements:

```
CREATE TABLE MEMBER (  
  Member_ID CHAR(5) NOT NULL PRIMARY KEY,  
  Name VARCHAR2(25) NOT NULL,  
  phone_number VARCHAR2(13) NOT NULL,  
  Address VARCHAR2(25) NOT NULL,  
  Date_of_Birth CHAR(10) NOT NULL,  
  Gender CHAR(1) NOT NULL,  
  Age_Bracket VARCHAR2(15) NOT NULL)
```

```
;  
;
```

```
CREATE TABLE GENRE (  
  Genre_ID CHAR(6) NOT NULL PRIMARY KEY,  
  Genre_type VARCHAR2(15) NOT NULL)
```

```
;  
;
```

```
CREATE TABLE MOVIES (  
  Movie_Title VARCHAR2(350) NOT NULL PRIMARY KEY,  
  Director VARCHAR2(25) NOT NULL,  
  Year CHAR(4) NOT NULL,  
  Content_Rating VARCHAR2(4) NOT NULL,  
  Genre_ID CHAR(6) NOT NULL,  
  FOREIGN KEY (Genre_ID) REFERENCES GENRE (Genre_ID) ON DELETE CASCADE)
```

```
;  
;
```

```
CREATE TABLE MOVIE_RENTAL (  
  Order_Number CHAR(6) NOT NULL PRIMARY KEY,  
  Rental_Date CHAR(10) NOT NULL,  
  Return_Date CHAR(10) NULL,  
  Member_ID CHAR(5) NOT NULL,  
  Movie_Title VARCHAR2(350) NOT NULL,  
  FOREIGN KEY (Member_ID) REFERENCES MEMBER (Member_ID) ON DELETE  
CASCADE,  
  FOREIGN KEY (Movie_Title) REFERENCES MOVIES (Movie_Title) ON DELETE CASCADE)
```

```
;  
;
```

```
CREATE TABLE PAYMENTS (  
  Payment_date CHAR(10) NOT NULL,  
  Payment_Amount FLOAT CHECK (Payment_Amount > 0) NOT NULL,  
  Payment_ID CHAR(6) NOT NULL PRIMARY KEY,  
  Card_Number CHAR(16) NOT NULL,  
  Member_ID CHAR(5) NOT NULL,
```

```
FOREIGN KEY (Member_ID) REFERENCES MEMBER (Member_ID) ON DELETE  
CASCADE)  
;  
;
```

Statements to populate tables:

```
INSERT INTO MEMBER VALUES ('54321', 'Miley Mchomes', '860-345-2134', '23 MacBurger  
Dr', '11-23-1978', 'F', '35-44 years old')
```

```
;  
;
```

```
INSERT INTO MEMBER VALUES ('34265', 'Tom Bordy', '860-168-9865', '23 Biden Dr',  
'11-23-2000', 'M', '18-24 years old')
```

```
;  
;
```

```
INSERT INTO MEMBER VALUES ('67843', 'Kim Jackson', '860-834-1863', '55 Chief Ave',  
'06-21-1965', 'F', '45+ years old')
```

```
;  
;
```

```
INSERT INTO MEMBER VALUES ('76954', 'Vanessa Prior', '860-287-3486', '654 Tumble St',  
'09-29-2001', 'F', '18-24 years old')
```

```
;  
;
```

```
INSERT INTO MEMBER VALUES ('23756', 'James Hyatt', '860-543-8665', '98 Deer St',  
'11-29-1990', 'M', '25-34 years old')
```

```
;  
;
```

```
INSERT INTO MEMBER VALUES ('64543', 'Stephen Potter', '860-645-2465', '543 Sea Dr',  
'11-25-1992', 'M', '25-34 years old')
```

```
;  
;
```

```
INSERT INTO MEMBER VALUES ('96643', 'David Pham', '860-899-8523', '55 King St',  
'01-23-2000', 'M', '18-24 years old')
```

```
;  
;
```

```
Select*
```

```
from member
```

```
;  
;
```

```
INSERT INTO PAYMENTS VALUES ('12-3-2020', '30.00', 'P33542', '9687234516548763',  
'13245')
```

```
;  
;
```

```
INSERT INTO PAYMENTS VALUES ('11-3-2020', '30.00', 'P33541', '9687234516548763',  
'13245')
```

```
;  
;
```

```
INSERT INTO PAYMENTS VALUES ('11-23-2020', '30.00', 'P65432', '1235675478962354',  
'54321')
```

```
;  
;
```

```
INSERT INTO PAYMENTS VALUES ('10-23-2020', '30.00', 'P65423', '1235675478962354',  
'54321')
```

```
;  
;
```

```

INSERT INTO PAYMENTS VALUES ('09-23-2020', '30.00', 'P27432', '1235675478962354',
'54321')
;
INSERT INTO PAYMENTS VALUES ('10-16-2020', '30.00', 'P84321', '4548243595741322',
'34265')
;
INSERT INTO PAYMENTS VALUES ('11-16-2020', '30.00', 'P83451', '4548243595741322',
'34265')
;
INSERT INTO PAYMENTS VALUES ('08-12-2020', '30.00', 'P94321', '9857345686791423',
'67843')
;
INSERT INTO PAYMENTS VALUES ('09-12-2020', '30.00', 'P33221', '9857345686791423',
'67843')
;
INSERT INTO PAYMENTS VALUES ('10-12-2020', '30.00', 'P96431', '9857345686791423',
'67843')
;
INSERT INTO PAYMENTS VALUES ('11-12-2020', '30.00', 'P93231', '9857345686791423',
'67843')
;
INSERT INTO PAYMENTS VALUES ('07-12-2020', '30.00', 'P09721', '2839495848593848',
'76954')
;
INSERT INTO PAYMENTS VALUES ('08-12-2020', '30.00', 'P53431', '2839495848593848',
'76954')
;
INSERT INTO PAYMENTS VALUES ('09-12-2020', '30.00', 'P65441', '2839495848593848',
'76954')
;
INSERT INTO PAYMENTS VALUES ('10-12-2020', '30.00', 'P67831', '2839495848593848',
'76954')
;
INSERT INTO PAYMENTS VALUES ('11-12-2020', '30.00', 'P13464', '2839495848593848',
'76954')
;
INSERT INTO PAYMENTS VALUES ('11-12-2020', '30.00', 'P76431', '9786094580975427',
'23756')
;
INSERT INTO PAYMENTS VALUES ('09-12-2020', '30.00', 'P10543', '5678432398674567',
'64543')
;
INSERT INTO PAYMENTS VALUES ('10-12-2020', '30.00', 'P34564', '5678432398674567',
'64543')

```

```

-
;
INSERT INTO PAYMENTS VALUES ('11-01-2020', '30.00', 'P78453', '5678432398674567',
'64543')
-
;
INSERT INTO PAYMENTS VALUES ('10-16-2020', '30.00', 'P48594', '8654568714609860',
'96643')
-
;
INSERT INTO PAYMENTS VALUES ('11-16-2020', '30.00', 'P34394', '8654568714609860',
'96643')
-
;
select*
from payments
-
;
INSERT INTO GENRE VALUES ('RO4322', 'Romance')
-
;
INSERT INTO MOVIES VALUES ('Forrest Gump', 'Robert Zemeckis', '1994', 'PG13', 'RO4322')
-
;
INSERT INTO GENRE VALUES ('AC6432', 'Action')
-
;
INSERT INTO MOVIES VALUES ('The Hunger Games', 'Gary Ross', '2012', 'PG13', 'AC6432')
-
;
INSERT INTO GENRE VALUES ('CO8432', 'Comedy')
-
;
INSERT INTO MOVIES VALUES ('Home Alone', 'Chris Columbus', '1990', 'PG', 'CO8432')
-
;
INSERT INTO GENRE VALUES ('DR8302', 'Drama')
-
;
INSERT INTO MOVIES VALUES ('The Godfather', 'Francis Coppola', '1972', 'R', 'DR8302')
-
;
INSERT INTO GENRE VALUES ('DR0583', 'Drama')
-
;
INSERT INTO MOVIES VALUES ('Dunkirk', 'Christopher Nolan', '2017', 'PG13', 'DR0583')
-
;
INSERT INTO GENRE VALUES ('AC9845', 'Action')
-
;
INSERT INTO MOVIES VALUES ('Logan', 'James Mangold', '2017', 'R', 'AC9845')
-
;
INSERT INTO GENRE VALUES ('AC3654', 'Action')
-
;
INSERT INTO MOVIES VALUES ('The Equailizer', 'Antoine Fuqua', '2014', 'R', 'AC3654')
-
;
INSERT INTO GENRE VALUES ('CO3475', 'Comedy')
-
;
INSERT INTO MOVIES VALUES ('Meet the Parents', 'Jay Roach', '2000', 'PG13', 'CO3475')

```

```
,
,
INSERT INTO GENRE VALUES ('RO9854', 'Romance')
,
,
```

```
INSERT INTO MOVIES VALUES ('The Notebook', 'Nick Cassavetes', '2004', 'PG13', 'RO9854')
,
,
```

```
INSERT INTO GENRE VALUES ('DR8964', 'Drama')
,
,
```

```
INSERT INTO MOVIES VALUES ('Mystic River', 'Clint Eastwood', '2003', 'R', 'DR8964')
,
,
```

```
INSERT INTO GENRE VALUES ('HO9534', 'Horror')
,
,
```

```
INSERT INTO MOVIES VALUES ('The Conjuring', 'James Wan', '2013', 'R', 'HO9534')
,
,
```

```
INSERT INTO GENRE VALUES ('HO4334', 'Horror')
,
,
```

```
INSERT INTO MOVIES VALUES ('Halloween', 'John Carpenter', '1978', 'R', 'HO4334')
,
,
```

```
INSERT INTO GENRE VALUES ('AC8547', 'Action')
,
,
```

Sample runs for each required report

1).

SQL Worksheet Clear Find Actions Save Run

```
1 select*
2 from Member
```

MEMBER_ID	NAME	PHONE_NUMBER	ADDRESS	DATE_OF_BIRTH	GENDER	AGE_BRACKET
13245	Bob Mac	860-132-4356	34 Hart Ln	01-23-1998	M	18-24 years old
54321	Miley Mchomes	860-345-2134	23 MacBurger Dr	11-23-1978	F	35-44 years old
34265	Tom Bordy	860-168-9865	23 Biden Dr	11-23-2000	M	18-24 years old
67843	Kim Jackson	860-834-1863	55 Chief Ave	06-21-1965	F	45+ years old
76954	Vanessa Prior	860-287-3486	654 Tumble St	09-29-2001	F	18-24 years old
23756	James Hyatt	860-543-8665	98 Deer St	11-29-1990	M	25-34 years old
64543	Stephen Potter	860-645-2465	543 Sea Dr	11-25-1992	M	25-34 years old
96643	David Pham	860-899-8523	55 King St	01-23-2000	M	18-24 years old

[Download CSV](#)
8 rows selected.

2).

Feedback ? Help nguyenst@my.easternct.edu

SQL Worksheet Clear Find Actions Save Run

```

1 SELECT Movie_title, count(*)
2 FROM movie_Rental
3 WHERE member_id IN
4 (SELECT member_id
5 FROM member
6 WHERE Gender = 'F')
7 GROUP by Movie_title
8 ORDER BY count(*) desc
9 FETCH NEXT 1 ROWS ONLY

```

MOVIE_TITLE	COUNT(*)
Logan	3

Download CSV

Feedback ? Help nguyenst@my.easternct.edu

SQL Worksheet Clear Find Actions Save Run

```

1 SELECT Movie_title, count(*)
2 FROM movie_Rental
3 WHERE member_id IN
4 (SELECT member_id
5 FROM member
6 WHERE Gender = 'M')
7 GROUP by Movie_title
8 ORDER BY count(*) desc
9 FETCH NEXT 1 ROWS ONLY

```

MOVIE_TITLE	COUNT(*)
Halloween	3

3).

Feedback ? Help nguyenst@my.easternct.edu

SQL Worksheet Clear Find Actions Save Run

```

1 SELECT genre.genre_type, count(*)
2 FROM movie_Rental
3 INNER JOIN movies
4 ON movies.movie_title = movie_Rental.movie_title
5 INNER JOIN genre
6 on genre.genre_id = movies.genre_id
7 GROUP BY genre.genre_type
8 order by count(*) desc
9 fetch next 1 row only

```

GENRE_TYPE	COUNT(*)
Horror	6

Download CSV