Syed Khurshid
010081191
D191 – Advance Data Management

For this Project I am assuming the role as an owner of the DVD rental shop. Since this is a business the profitability and sales of the business provides an upper-level view of the business. However due to pending financial data, I am limiting to reviewing business performance based on the following 2:

- ***Sales made by location/Store***: This is to evaluate which location is profitable and can lead to business decisions such as deciding if the location is worth keeping or moving to another location
- ***Sales made by film genre***: Certain films attract certain level of customers and having a stock of films fitting that criteria will keep customers from going to our competitor

Sales made by location/Store

As a business owner, one of the most crucial decisions that an owner can make is to decide if the location provides a necessary stream of revenue. Another use case scenario is to see which area and which date the rental is being carried out. Dates provide an idea on which days are rental more frequent as well as the Genre usually being picked on.
Two views are being used here:

- ***sales_by_location_detailed***: which provides the sales data in detailed by Genre and location where the individual sales by location which uses two fields "city.city" and "country.country" columns concatenated with spaces. The columns are taken from the City table and the Country table.
  The table is made from 11 tables using a JOIN statement with the main table being the "Payment" Table, the other tables that are joined are from the "Rental", "Inventory", "Film", "Film_Category", "Category", "Staff", "Store", "Address", "City" and "Country" Tables which have been joined using the respective Primary and Foreign keys (Please see code).
  The main fields that are being utilized are the following:
    - City (City Table) Concatenated with ", " and Country (Country table) columns
    - Payment_ID (Payment Table)
    - Rental_ID (Payment Table)
    - Payment_Date (Payment Table) that using CAST() has been converted to a DATE Format making it easier to read what date the payment was made instead of a TIMESTAMP format
    - Inventory_ID (Rental Table)
    - Name (Category Table) which was renamed as "Genre" for it to be easier to identify respective film category
    - Title (Film Table) which is the name of the Films
    - Amount (Payment Table) which was using the CAST() function converted to MONEY as currency format is a financial view that is easy to understand when presenting to view at a high level view

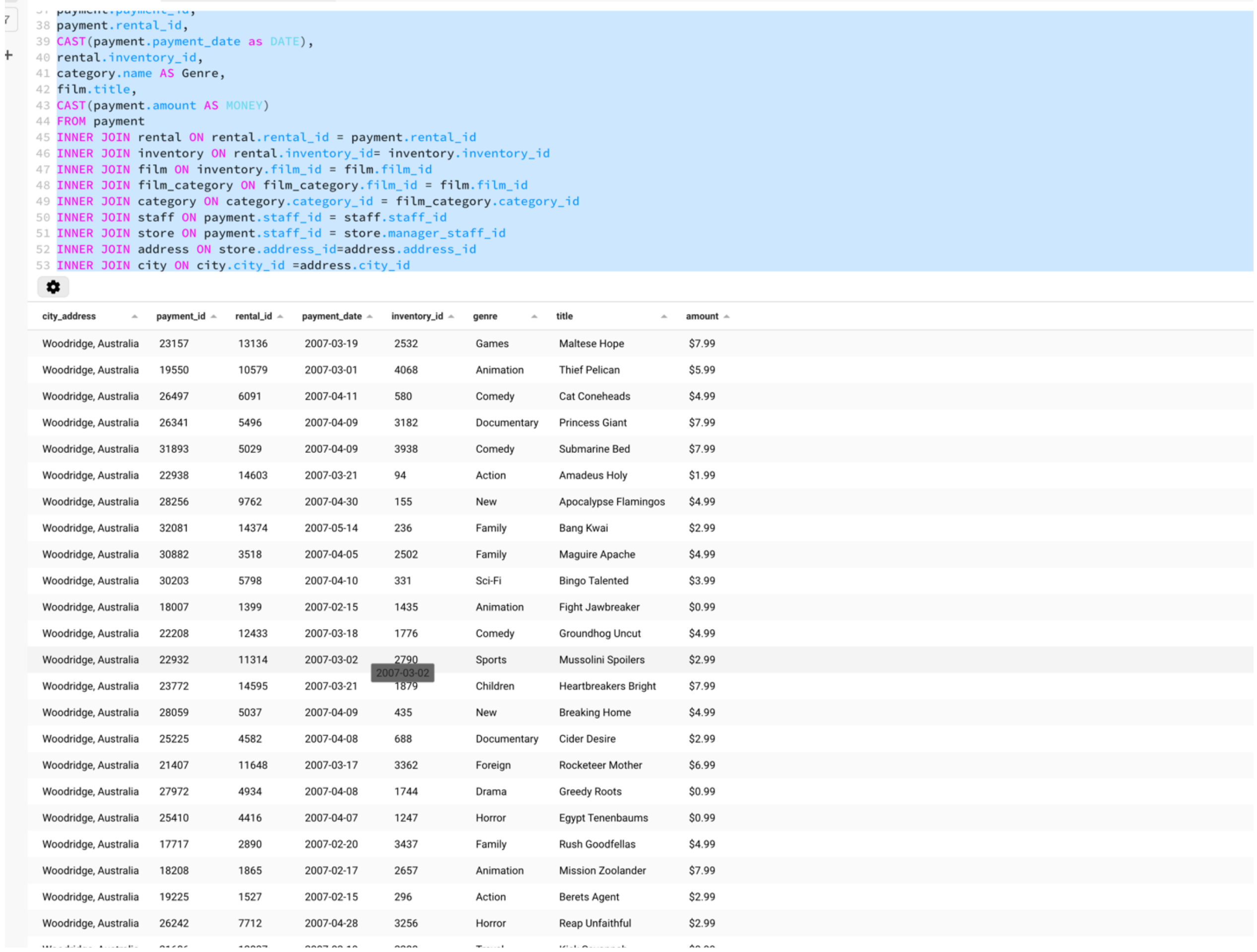
```
CREATE OR REPLACE VIEW sales_by_location_detailed AS
SELECT
city.city||', '||country.country AS city_address,
payment.payment_id,
```

Syed Khurshid
010081191
D191 – Advance Data Management

```
payment.rental_id,
CAST(payment.payment_date as DATE),
rental.inventory_id,
category.name AS Genre,
film.title,
CAST(payment.amount AS MONEY)
FROM payment
INNER JOIN rental ON rental.rental_id = payment.rental_id
INNER JOIN inventory ON rental.inventory_id= inventory.inventory_id
INNER JOIN film ON inventory.film_id = film.film_id
INNER JOIN film_category ON film_category.film_id = film.film_id
INNER JOIN category ON category.category_id = film_category.category_id
INNER JOIN staff ON payment.staff_id = staff.staff_id
INNER JOIN store ON payment.staff_id = store.manager_staff_id
INNER JOIN address ON store.address_id=address.address_id
INNER JOIN city ON city.city_id =address.city_id
INNER JOIN country ON city.country_id = country.country_id
GROUP BY category.name, payment.payment_id, rental.inventory_id, city.city, country.country, film.title
ORDER BY city_address DESC;

```

Screenshot also can be seen in the screenshot folder



```
payment.rental_id,
CAST(payment.payment_date as DATE),
rental.inventory_id,
category.name AS Genre,
film.title,
CAST(payment.amount AS MONEY)
FROM payment
INNER JOIN rental ON rental.rental_id = payment.rental_id
INNER JOIN inventory ON rental.inventory_id= inventory.inventory_id
INNER JOIN film ON inventory.film_id = film.film_id
INNER JOIN film_category ON film_category.film_id = film.film_id
INNER JOIN category ON category.category_id = film_category.category_id
INNER JOIN staff ON payment.staff_id = staff.staff_id
INNER JOIN store ON payment.staff_id = store.manager_staff_id
INNER JOIN address ON store.address_id=address.address_id
INNER JOIN city ON city.city_id =address.city_id
```

city_address	payment_id	rental_id	payment_date	inventory_id	genre	title	amount
Woodridge, Australia	23157	13136	2007-03-19	2532	Games	Maltese Hope	$7.99
Woodridge, Australia	19550	10579	2007-03-01	4068	Animation	Thief Pelican	$5.99
Woodridge, Australia	26497	6091	2007-04-11	580	Comedy	Cat Coneheads	$4.99
Woodridge, Australia	26341	5496	2007-04-09	3182	Documentary	Princess Giant	$7.99
Woodridge, Australia	31893	5029	2007-04-09	3938	Comedy	Submarine Bed	$7.99
Woodridge, Australia	22938	14603	2007-03-21	94	Action	Amadeus Holy	$1.99
Woodridge, Australia	28256	9762	2007-04-30	155	New	Apocalypse Flamingos	$4.99
Woodridge, Australia	32081	14374	2007-05-14	236	Family	Bang Kwai	$2.99
Woodridge, Australia	30882	3518	2007-04-05	2502	Family	Maguire Apache	$4.99
Woodridge, Australia	30203	5798	2007-04-10	331	Sci-Fi	Bingo Talented	$3.99
Woodridge, Australia	18007	1399	2007-02-15	1435	Animation	Fight Jawbreaker	$0.99
Woodridge, Australia	22208	12433	2007-03-18	1776	Comedy	Groundhog Uncut	$4.99
Woodridge, Australia	22932	11314	2007-03-02	2790	Sports	Mussolini Spoilers	$2.99
Woodridge, Australia	23772	14595	2007-03-21	1879	Children	Heartbreakers Bright	$7.99
Woodridge, Australia	28059	5037	2007-04-09	435	New	Breaking Home	$4.99
Woodridge, Australia	25225	4582	2007-04-08	688	Documentary	Cider Desire	$2.99
Woodridge, Australia	21407	11648	2007-03-17	3362	Foreign	Rocketeer Mother	$6.99
Woodridge, Australia	27972	4934	2007-04-08	1744	Drama	Greedy Roots	$0.99
Woodridge, Australia	25410	4416	2007-04-07	1247	Horror	Egypt Tenenbaums	$0.99
Woodridge, Australia	17717	2890	2007-02-20	3437	Family	Rush Goodfellas	$4.99
Woodridge, Australia	18208	1865	2007-02-17	2657	Animation	Mission Zoolander	$7.99
Woodridge, Australia	19225	1527	2007-02-15	296	Action	Berets Agent	$2.99
Woodridge, Australia	26242	7712	2007-04-28	3256	Horror	Reap Unfaithful	$2.99

- ***<u>sales_by_location_summary :</u>*** The Summary view provides the business owners a quick glance of how much revenue each location made. As a business owner I would use such a report first and compare location revenue to see under or over performing and based on that next decision to review further can be taken. This view only requires to Fields

Syed Khurshid
010081191
D191 – Advance Data Management

- City (City Table Column) Concatenated with “, “ and Country (Country table) columns
- Amount (Payment Table Column) using the SUM() function location was summed up and then using the CAST() was converted to MONEY Format as the “$” makes it easier financially to identify money related transactions

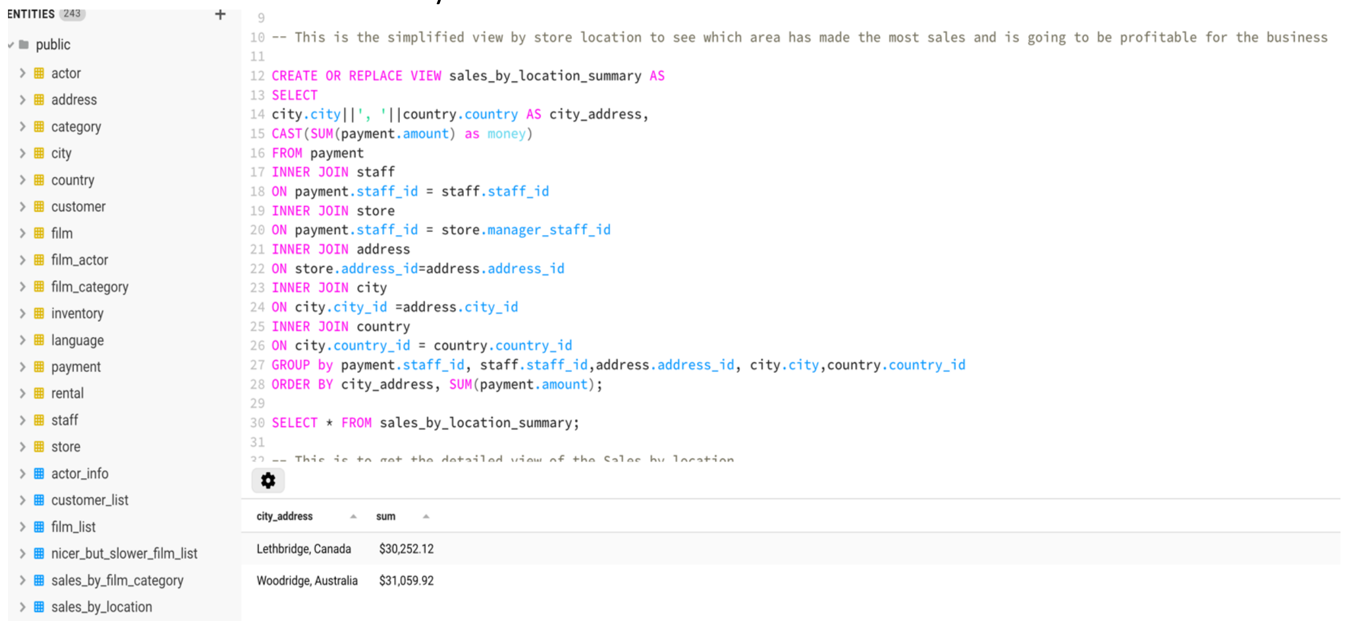
Two Tables are necessary for this view to work; The Payment Table and the Staff Table. The Payment provides the Revenue details whereas the Staff table provides the Staff id which can identify the location of the where the rental revenue was made.
The Table utilizes the same JOINS and GROUPING as from the summary table however, the only difference are only 2 columns being used mentioned above.

```sql
CREATE OR REPLACE VIEW sales_by_location_summary AS
SELECT
city.city||', '||country.country AS city_address,
CAST(SUM(payment.amount) as money)
FROM payment
INNER JOIN staff
ON payment.staff_id = staff.staff_id
INNER JOIN store
ON payment.staff_id = store.manager_staff_id
INNER JOIN address
ON store.address_id=address.address_id
INNER JOIN city
ON city.city_id =address.city_id
INNER JOIN country
ON city.country_id = country.country_id
GROUP by payment.staff_id, staff.staff_id,address.address_id, city.city,country.country_id
ORDER BY city_address, SUM(payment.amount);

SELECT * FROM sales_by_location_summary;

```

Screenshot of Summary also found in the Screenshot folder



```sql

-- This is the simplified view by store location to see which area has made the most sales and is going to be profitable for the business

CREATE OR REPLACE VIEW sales_by_location_summary AS
SELECT
city.city||', '||country.country AS city_address,
CAST(SUM(payment.amount) as money)
FROM payment
INNER JOIN staff
ON payment.staff_id = staff.staff_id
INNER JOIN store
ON payment.staff_id = store.manager_staff_id
INNER JOIN address
ON store.address_id=address.address_id
INNER JOIN city
ON city.city_id =address.city_id
INNER JOIN country
ON city.country_id = country.country_id
GROUP by payment.staff_id, staff.staff_id,address.address_id, city.city,country.country_id
ORDER BY city_address, SUM(payment.amount);

SELECT * FROM sales_by_location_summary;

-- This is to get the detailed view of the Sales by location
```

city_address	sum
Lethbridge, Canada	$30,252.12
Woodridge, Australia	$31,059.92