MySQL Data analysis Project

Welcome, @Prince Kunal

DESCRIPTION : The Company and the partner were recently approached by another local business owner who is interested in purchasing Maven Movies. He primarily owns restaurants and bars, so he has lots of questions about business in general. His offer seems very generous, so I am going to entertains his questions.

1. My partner and I want to come by each of the stores in person and meet the managers.
Please send over the managers' names at each store, with the full address of each property (street address, district, city, and country please).

```
SELECT
    staff.first_name AS manager_first_name,
    staff.last_name AS manager_last_name,
    address.address,
    address.district,
    city.city,
    country.country

FROM store
    LEFT JOIN staff ON store.manager_staff_id = staff.staff_id
    LEFT JOIN address ON store.address_id = address.address_id
    LEFT JOIN city ON address.city_id = city.city_id
    LEFT JOIN country ON city.country_id = country.country_id;
```

2. I would like to get a better understanding of all of the inventory that would come along with the business. Please pull together a list of each inventory item you have

stocked, including the store_id number, the inventory_id, the name of the film, the film's rating, its rental rate and replacement cost.

```
SELECT
   inventory.store_id,
   inventory.inventory_id,
   film.title,
   film.rating,
   film.rental_rate,
   film.replacement_cost
FROM inventory
      LEFT JOIN film
            ON inventory.film_id = film.film_id;
```

3. From the same list of films you just pulled, please roll that data up and provide a summary level overview
of your inventory. We would like to know how many inventory items you have with each rating at each store.

```
SELECT
      inventory.store_id,
   film.rating,
   COUNT(inventory_id) AS inventory_items
FROM inventory
      LEFT JOIN film
            ON inventory.film_id = film.film_id
GROUP BY
      inventory.store_id,
   film.rating;
```

4. Similarly, we want to understand how diversified the inventory is in terms of replacement cost. We want to see how big of a hit it would be if a certain category of film became unpopular at a certain store.We would like to see the number of films, as well as the average replacement cost, and total replacement cost, sliced by store and film category.

```
SELECT
    store_id,
  category.name AS category,
  COUNT(inventory.inventory_id) AS films,
  AVG(film.replacement_cost) AS avg_replacement_cost,
  SUM(film.replacement_cost) AS total_replacement_cost

FROM inventory
    LEFT JOIN film
        ON inventory.film_id = film.film_id
    LEFT JOIN film_category
        ON film.film_id = film_category.film_id
    LEFT JOIN category
        ON category.category_id = film_category.category_id

GROUP BY
    store_id,
  category.name

ORDER BY
    SUM(film.replacement_cost) DESC;
```

5. We want to make sure you folks have a good handle on who your customers are. Please provide a list of all customer names, which store they go to, whether or not they are currently active, and their full addresses – street address, city, and country.

```
SELECT
  customer.first_name,
  customer.last_name,
  customer.store_id,
  customer.active,
  address.address,
  city.city,
  country.country

FROM customer
  LEFT JOIN address ON customer.address_id = address.address_id
  LEFT JOIN city ON address.city_id = city.city_id
```

```
    LEFT JOIN country ON city.country_id = country.country_id;
```

6. We would like to understand how much your customers are spending with you, and also to know who your most valuable customers are. Please pull together a list of customer names, their total lifetime rentals, and the sum of all payments you have collected from them. It would be great to see this ordered on total lifetime value, with the most valuable customers at the top of the list.

```
SELECT
        customer.first_name,
    customer.last_name,
    COUNT(rental.rental_id) AS total_rentals,
    SUM(payment.amount) AS total_payment_amount

FROM customer
        LEFT JOIN rental ON customer.customer_id = rental.customer_id
    LEFT JOIN payment ON rental.rental_id = payment.rental_id

GROUP BY
        customer.first_name,
    customer.last_name

ORDER BY
        SUM(payment.amount) DESC
    ;
```

7. My partner and I would like to get to know your board of advisors and any current investors.
Could you please provide a list of advisor and investor names in one table?
Could you please note whether they are an investor or an advisor, and for the investors,
it would be good to include which company they work with.

```
SELECT
        'investor' AS type,
    first_name,
```

```
    last_name,
    company_name
FROM investor

UNION

SELECT
        'advisor' AS type,
    first_name,
    last_name,
    NULL
FROM advisor;
```

8. We're interested in how well you have covered the most-awarded actors.
Of all the actors with three types of awards, for what % of them do we carry a film?
And how about for actors with two types of awards? Same questions.
Finally, how about actors with just one award?

```
SELECT
        CASE
            WHEN actor_award.awards = 'Emmy, Oscar, Tony ' THEN '3 awards'
        WHEN actor_award.awards IN ('Emmy, Oscar','Emmy, Tony', 'Oscar, Tony') THEN '2
awards'
            ELSE '1 award'
        END AS number_of_awards,
    AVG(CASE WHEN actor_award.actor_id IS NULL THEN 0 ELSE 1 END) AS
pct_w_one_film

FROM actor_award


GROUP BY
        CASE
            WHEN actor_award.awards = 'Emmy, Oscar, Tony ' THEN '3 awards'
        WHEN actor_award.awards IN ('Emmy, Oscar','Emmy, Tony', 'Oscar, Tony') THEN '2
awards'
            ELSE '1 award'
        END
```

SECOND CASE:

Description:

The company has Maven Movies rental business in city. The company's insurance policy is up for renewal and the insurance company's underwriters need some updated information from us before they will issue a new policy.

As a analyst I have to use my SQL skills to extract and analyze data from various tables in Maven Movies database to answer the underwriter's queries.

1. **We will need a list of all staff members ,including their first and last name ,email address and the store identification number where they work.**

```
SELECT

first_name,
last_name,
email,
store_id

FROM staff;
```

2. **We will need separate counts of inventory items held at each of your two stores.**

```
SELECT

store_id,

COUNT (inventory_id) AS inventory_items

FROM inventory
```

```
GROUP BY
    store_id;
```

3. **We will need a count of active customers for each of your stores. Separately.**

```
SELECT

    store_id,

    COUNT (customer_id) AS active_customers

FROM customer

WHERE active=1
GROUP BY
store_id;
```

4. **In order to assess the liability of a data breach, we will need you to provide a count of all customer email addresses stored in the database.**

```
SELECT

COUNT (email)

FROM customers
```

5. **We are interested in how diverse your film offering is as a means of understanding how likely you are to keep customers engaged in the future. Please provide a count of unique film titles you have in inventory at each store and then provide a count of the unique categories of films you provide.**

```
SELECT
    store_id,
    COUNT(DISTINCT film_id) AS unique_films

FROM inventory
GROUP BY
```

```
    store_id;

SELECT
    COUNT(DISTINCT name) AS unique_categories
FROM category;
```

6. **We would like to understand the replacement cost of your films. Please provide the replacement cost for the film that is least expensive to replace, the most expensive to replace, and the average of all films you carry.**

```
SELECT

  MIN(replacement_cost) AS least_expensive,

  MAX(replacement_cost) AS most_expensive,

  AVG(replacement_cost) AS average_replacement_cost

FROM film;
```

7. **We are interested in having you put payment monitoring systems and maximum payment processing restrictions in place in order to minimize the future risk of fraud by your staff. Please provide the average payment you process, as well as the maximum payment you have processed.**

```
SELECT
    AVG(amount) AS average_payment,
   MAX(amount) AS max_payment
FROM payment;
```

8. **We would like to better understand what your customer base looks like. Please provide a list of all customer identification values, with a count of rentals they have made all-time, with your highest volume customers at the top of the list.**

```
SELECT
    customer_id,
  COUNT(rental_id) AS number_of_rentals

FROM rental
```

```
GROUP BY
    customer_id
ORDER BY
    COUNT(rental_id) DESC;



/
```