QUERY :

--1. Identify the top 10 customers and their email so we can reward them

```
SELECT
	CONCAT(customer.first_name, ' ', customer.last_name) as FullName,
	customer.email,
	SUM(payment.amount) as TotalPurchase
FROM customer
	JOIN payment
	USING(customer_id)
GROUP BY FullName, customer.email
ORDER BY TotalPurchase DESC
LIMIT 10;
```

--2. Identify the bottom 10 customers and their emails

```
SELECT
	CONCAT(customer.first_name, ' ', customer.last_name) as FullName,
	customer.email,
	SUM(payment.amount) as TotalPurchase
FROM customer
	JOIN payment ON customer.customer_id = payment.customer_id
GROUP BY FullName, customer.email
ORDER BY TotalPurchase ASC
LIMIT 10;
```

--3. What are the most profitable movie genres (ratings)?

```
SELECT
	category.name AS GenreFIlm,
	--film.rating,
	SUM(payment.amount) AS total_revenue
FROM film
	JOIN film_category ON film.film_id = film_category.film_id
```

```
                JOIN category ON film_category.category_id = category.category_id
                        JOIN inventory ON film.film_id = inventory.film_id
                                JOIN rental ON inventory.inventory_id = rental.inventory_id
                                        JOIN payment ON rental.rental_id = payment.rental_id
GROUP BY category.name /*,film.rating*/
ORDER BY total_revenue DESC;
```

--4. How many rented movies were returned late, early, and on time?

```
WITH q1 AS (SELECT*, DATE_PART('day', return_date - rental_date)
                                AS date_difference
                FROM rental),
        q2 AS (SELECT rental_duration, date_difference,
                                CASE
                                        WHEN rental_duration > date_difference THEN 'EARLY'
                                        WHEN rental_duration < date_difference THEN 'LATE'
                                        ELSE 'ON TIME'
                                END AS Return_status
                        FROM film f
                        JOIN inventory i ON f.film_id = i.film_id
                        JOIN q1 ON i.inventory_id = q1.inventory_id)
SELECT return_status, count(*) AS total_film
FROM q2
GROUP BY return_status
ORDER BY total_film DESC;
```

--5. What is the customer base in the countries where we have a presence ?

```
SELECT country,
                count(DISTINCT customer_id) AS customer_base
FROM country
        JOIN city
        USING(country_id)
```

```
        JOIN address
        USING(city_id)
            JOIN customer
            USING(address_id)
GROUP BY country
ORDER BY customer_base DESC;
```

--6. Which country is the most profitable for the business ?

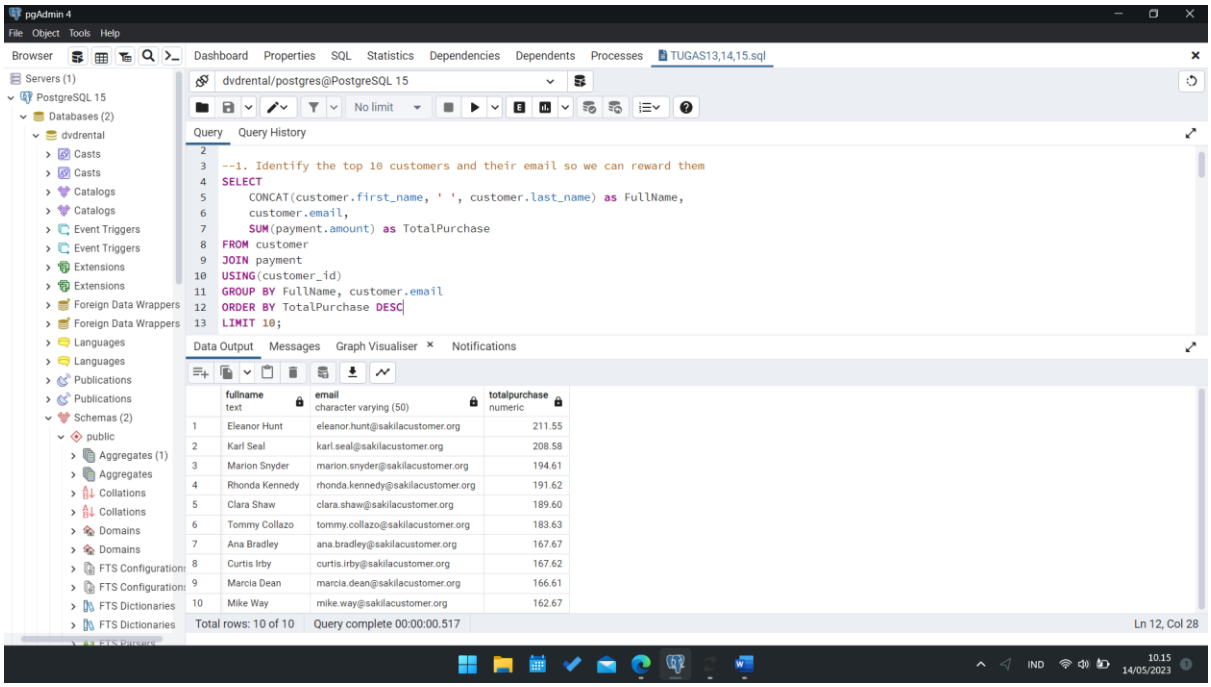
```
SELECT country,
        SUM(amount) AS total_sales
FROM country
    JOIN city
    USING(country_id)
        JOIN address
        USING(city_id)
            JOIN customer
            USING(address_id)
                JOIN payment
                USING(customer_id)
GROUP BY country
ORDER BY total_sales DESC;
```

--7. What is the average rental rate per movie genre (rating) ?

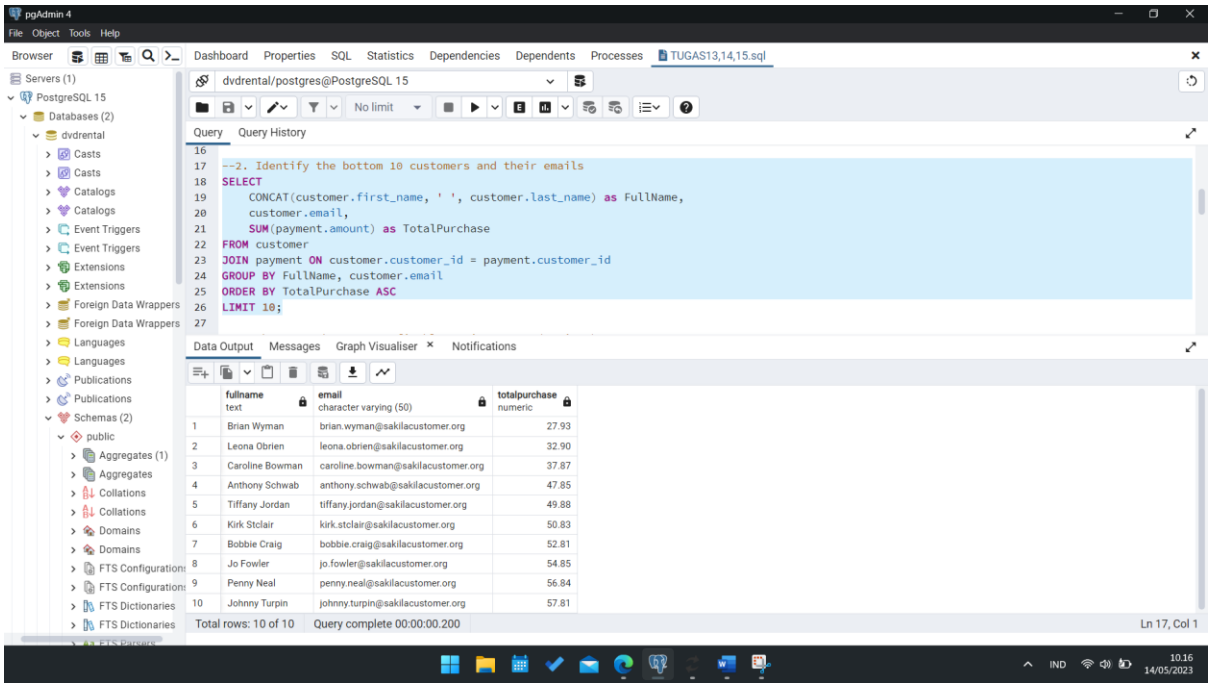
```
SELECT c.name AS genre, ROUND(AVG(f.rental_rate),2) AS AverageRentalRate
FROM category c
    JOIN film_category fc
    USING(category_id)
        JOIN film f
        USING(film_id)
GROUP BY 1
ORDER BY 2 DESC;
```

EXECUTE RESULT :

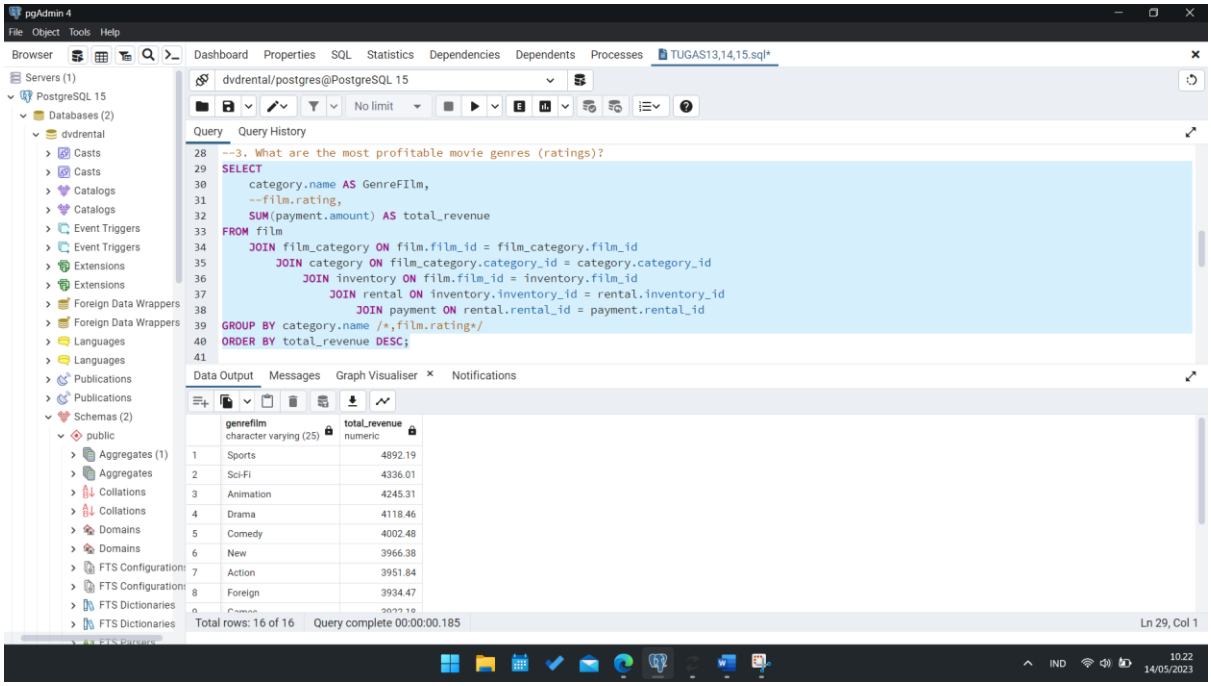
1.



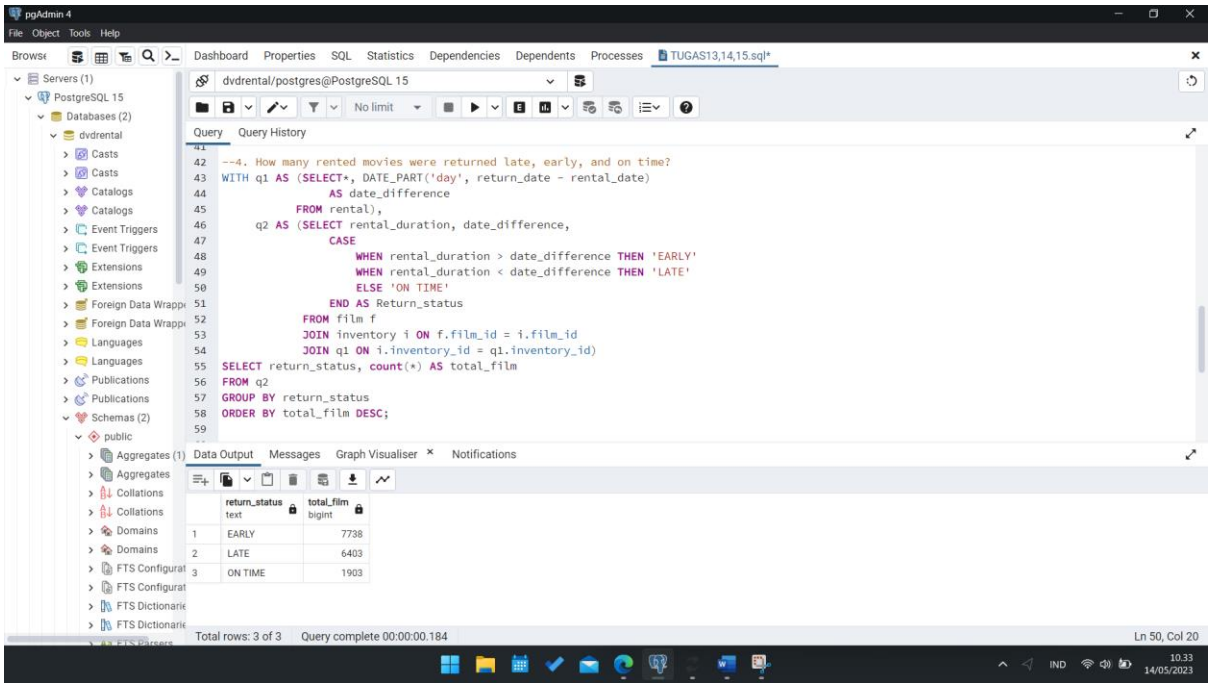
2.



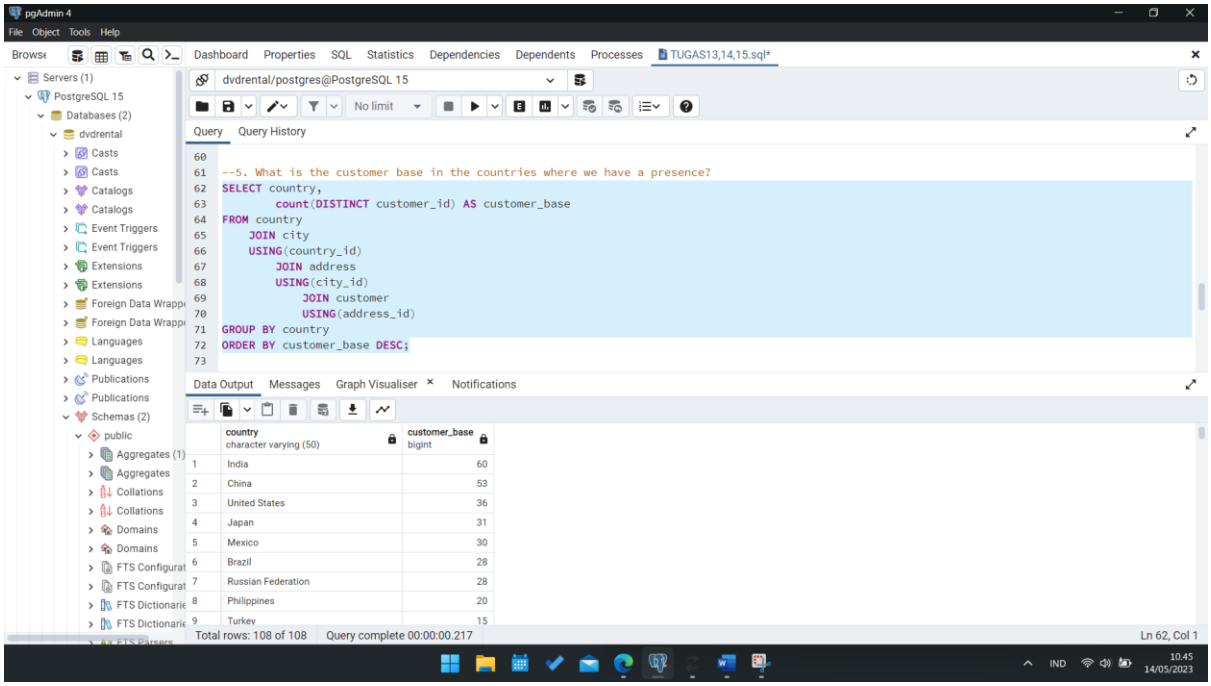
3.


pgAdmin 4
File Object Tools Help
Browser
Dashboard Properties SQL Statistics Dependencies Dependents Processes TUGAS13,14,15.sql*
Servers (1)
PostgreSQL 15
Databases (2)
dvdrental
dvdrental/postgres@PostgreSQL 15
No limit
Query Query History
--3. What are the most profitable movie genres (ratings)?
SELECT
category.name AS GenreFIlm,
--film.rating,
SUM(payment.amount) AS total_revenue
FROM film
JOIN film_category ON film.film_id = film_category.film_id
JOIN category ON film_category.category_id = category.category_id
JOIN inventory ON film.film_id = inventory.film_id
JOIN rental ON inventory.inventory_id = rental.inventory_id
JOIN payment ON rental.rental_id = payment.rental_id
GROUP BY category.name /*,film.rating*/
ORDER BY total_revenue DESC;
Data Output Messages Graph Visualiser Notifications
genrefilm character varying (25)
total_revenue numeric
1 Sports 4892.19
2 Sci-Fi 4336.01
3 Animation 4245.31
4 Drama 4118.46
5 Comedy 4002.48
6 New 3966.38
7 Action 3951.84
8 Foreign 3934.47
Total rows: 16 of 16
Query complete 00:00:00.185
Ln 29, Col 1
IND
10.22
14/05/2023

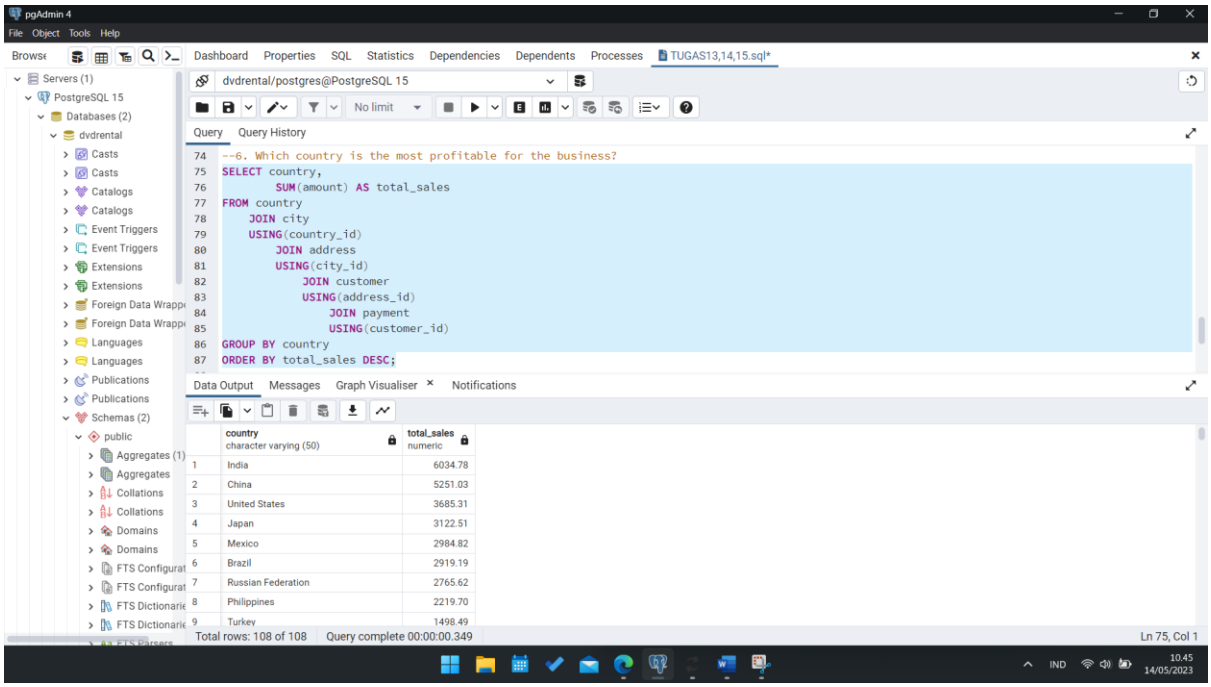
4.


pgAdmin 4
File Object Tools Help
Dashboard Properties SQL Statistics Dependencies Dependents Processes TUGAS13,14,15.sql*
dvdrental/postgres@PostgreSQL 15
No limit
Query Query History
--4. How many rented movies were returned late, early, and on time?
WITH q1 AS (SELECT*, DATE_PART('day', return_date - rental_date)
AS date_difference
FROM rental),
q2 AS (SELECT rental_duration, date_difference,
CASE
WHEN rental_duration > date_difference THEN 'EARLY'
WHEN rental_duration < date_difference THEN 'LATE'
ELSE 'ON TIME'
END AS Return_status
FROM film f
JOIN inventory i ON f.film_id = i.film_id
JOIN q1 ON i.inventory_id = q1.inventory_id)
SELECT return_status, count(*) AS total_film
FROM q2
GROUP BY return_status
ORDER BY total_film DESC;
Data Output Messages Graph Visualiser Notifications
return_status text
total_film bigint
1 EARLY 7738
2 LATE 6403
3 ON TIME 1903
Total rows: 3 of 3
Query complete 00:00:00.184
Ln 50, Col 20
IND
10.33
14/05/2023

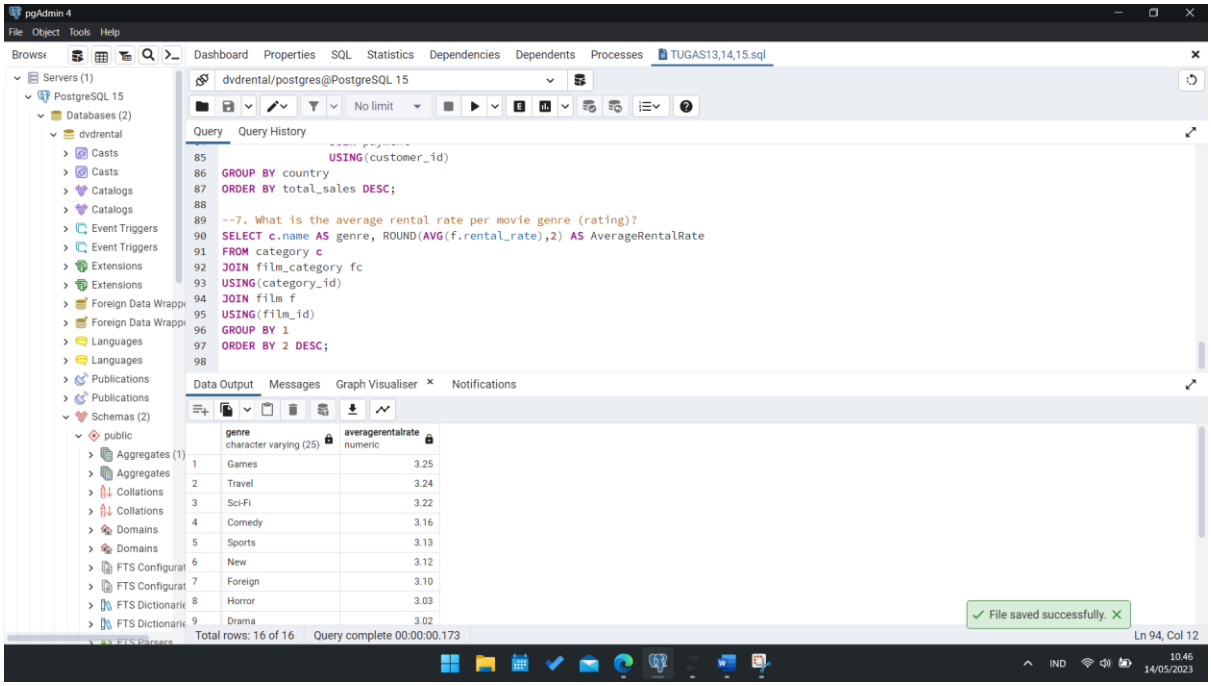
5.


pgAdmin 4
File Object Tools Help
Dashboard Properties SQL Statistics Dependencies Dependents Processes TUGAS13,14,15.sql*
dvdrental/postgres@PostgreSQL 15
Query Query History
--5. What is the customer base in the countries where we have a presence?
SELECT country,
count(DISTINCT customer_id) AS customer_base
FROM country
JOIN city
USING(country_id)
JOIN address
USING(city_id)
JOIN customer
USING(address_id)
GROUP BY country
ORDER BY customer_base DESC;
Data Output Messages Graph Visualiser Notifications
country character varying (50)
customer_base bigint
1 India 60
2 China 53
3 United States 36
4 Japan 31
5 Mexico 30
6 Brazil 28
7 Russian Federation 28
8 Philippines 20
9 Turkey 15
Total rows: 108 of 108
Query complete 00:00:00.217
Ln 62, Col 1

6.

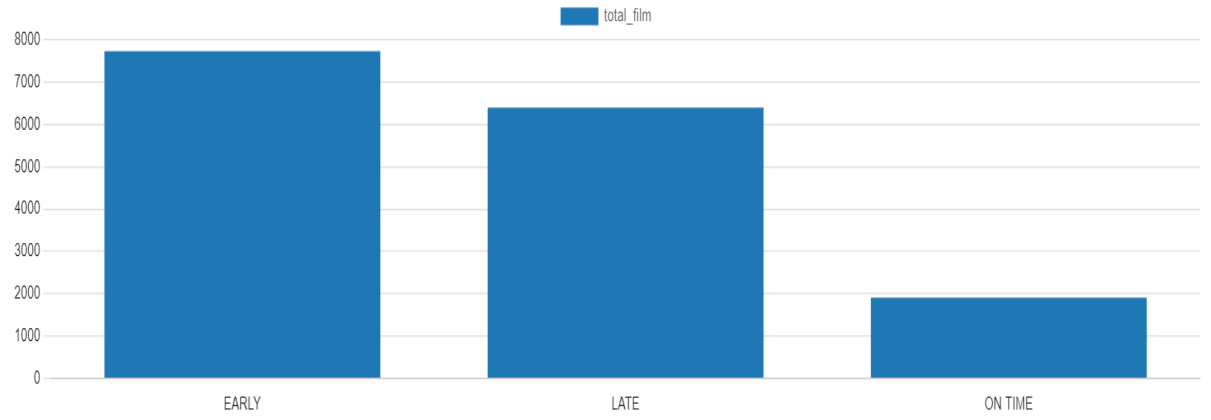

pgAdmin 4
File Object Tools Help
Dashboard Properties SQL Statistics Dependencies Dependents Processes TUGAS13,14,15.sql*
dvdrental/postgres@PostgreSQL 15
Query Query History
--6. Which country is the most profitable for the business?
SELECT country,
SUM(amount) AS total_sales
FROM country
JOIN city
USING(country_id)
JOIN address
USING(city_id)
JOIN customer
USING(address_id)
JOIN payment
USING(customer_id)
GROUP BY country
ORDER BY total_sales DESC;
Data Output Messages Graph Visualiser Notifications
country character varying (50)
total_sales numeric
1 India 6034.78
2 China 5251.03
3 United States 3685.31
4 Japan 3122.51
5 Mexico 2984.82
6 Brazil 2919.19
7 Russian Federation 2765.62
8 Philippines 2219.70
9 Turkey 1498.49
Total rows: 108 of 108
Query complete 00:00:00.349
Ln 75, Col 1

7.

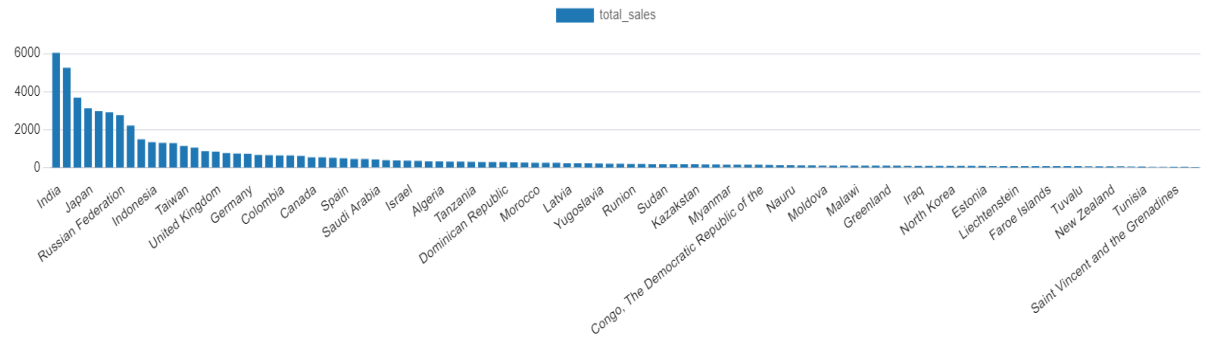

pgAdmin 4
File Object Tools Help
Dashboard Properties SQL Statistics Dependencies Dependents Processes TUGAS13,14,15.sql
dvdrental/postgres@PostgreSQL 15
No limit
Query Query History
USING(customer_id)
GROUP BY country
ORDER BY total_sales DESC;
--7. What is the average rental rate per movie genre (rating)?
SELECT c.name AS genre, ROUND(AVG(f.rental_rate),2) AS AverageRentalRate
FROM category c
JOIN film_category fc
USING(category_id)
JOIN film f
USING(film_id)
GROUP BY 1
ORDER BY 2 DESC;
Data Output Messages Graph Visualiser Notifications
genre character varying (25)
averagerentalrate numeric
1 Games 3.25
2 Travel 3.24
3 Sci-Fi 3.22
4 Comedy 3.16
5 Sports 3.13
6 New 3.12
7 Foreign 3.10
8 Horror 3.03
9 Drama 3.02
File saved successfully.
Total rows: 16 of 16
Query complete 00:00:00.173
Ln 94, Col 12

VISUAL :

3.


total_film
8000
7000
6000
5000
4000
3000
2000
1000
0
EARLY
LATE
ON TIME

6.


total_sales
6000
4000
2000
0
India
Japan
Russian Federation
Indonesia
Taiwan
United Kingdom
Germany
Colombia
Canada
Spain
Saudi Arabia
Israel
Algeria
Tanzania
Dominican Republic
Morocco
Latvia
Yugoslavia
Runion
Sudan
Kazakstan
Myanmar
Congo, The Democratic Republic of the
Nauru
Moldova
Malawi
Greenland
Iraq
North Korea
Estonia
Liechtenstein
Faroe Islands
Tuvalu
New Zealand
Tunisia
Saint Vincent and the Grenadines