Kareemah Ashiru
Exercise 3.6

1. **Check for and clean dirty data**
    a. **Duplicates:** No duplicates found for film and customer fact tables.

i.

Dashboard × Processes × Rockbuster/postgres@PostgreSQL 16* ×

Rockbuster/postgres@PostgreSQL 16

No limit

Query Query History

```
SELECT title,
       release_year,
       language_id,
       rental_duration,
       COUNT(*)
FROM film
GROUP BY title,
         release_year,
         language_id,
         rental_duration
HAVING COUNT(*) >1;
```

Data Output Messages Notifications

title character varying (255)	release_year integer	language_id smallint	rental_duration smallint	count bigint

Dashboard × Processes × Rockbuster/postgres@PostgreSQL 16* ×

Rockbuster/postgres@PostgreSQL 16

No limit

Query Query History

```
SELECT
       customer_id
       ,store_id
       ,first_name
       ,last_name
       ,email
       ,address_id
       ,activebool
       ,create_date
       ,last_update
       ,active
       ,COUNT(*)
FROM customer
GROUP BY
       customer_id
       ,store_id
       ,first_name
       ,last_name
       ,email
       ,address_id
       ,activebool
       ,create_date
       ,last_update
       ,active
HAVING COUNT(*) >1;
```

Data Output Messages Notifications

customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)	email character varying (50)	address_id smallint	activebool boolean	create_date date	last_update timestamp without time zone	active integer	count bigint

b. **Non Uniform data:** 1000 out of 1000 records were returned for the film table. This means that all records are unique. For the customer table; 599 out 599 records were returned. This means that all the records are unique.

Dashboard ✕ Processes ✕ Rockbuster/postgres@PostgreSQL 16* ✕

Rockbuster/postgres@PostgreSQL 16

No limit

Query Query History

```
SELECT DISTINCT
film_id,
title,
description,
release_year,
language_id,
rental_duration,
rental_rate,
length,
replacement_cost,
rating,
last_update,
special_features,
fulltext
FROM film;
```

Scratch Pad ✕

Data Output Messages Notifications

	film_id [PK] integer	title character varying (255)	description text	release_year integer	language_id smallint	rental_duration smallint	rental_rate numeric (4,2)	length smallint	replacement_cost numeric (5,2)	rating mpaa_rating	last_update timestamp without time zone	special_features text[]
1	1	Academy Dinosaur	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies	2006	1	6	0.99	86	20.99	PG	2013-05-26 14:50:58.951	{"Deleted Scenes
2	2	Ace Goldfinger	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China	2006	1	3	4.99	48	12.99	G	2013-05-26 14:50:58.951	{Trailers,"Deleted
3	3	Adaptation Holes	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory	2006	1	7	2.99	50	18.99	NC-17	2013-05-26 14:50:58.951	{Trailers,"Deleted
4	4	Affair Prejudice	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank	2006	1	5	2.99	117	26.99	G	2013-05-26 14:50:58.951	{Commentaries,"
5	5	African Egg	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico	2006	1	6	2.99	130	22.99	G	2013-05-26 14:50:58.951	{"Deleted Scenes
6	6	Agent Truman	A Intrepid Panorama of a Robot And a Boy who must Escape a Sumo Wrestler in Ancient China	2006	1	3	2.99	169	17.99	PG	2013-05-26 14:50:58.951	{"Deleted Scenes
7	7	Airplane Sierra	A Touching Saga of a Hunter And a Butler who must Discover a Butler in A Jet Boat	2006	1	6	4.99	62	28.99	PG-13	2013-05-26 14:50:58.951	{Trailers,"Deleted
8	8	Airport Pollock	A Epic Tale of a Moose And a Girl who must Confront a Monkey in Ancient India	2006	1	6	4.99	54	15.99	R	2013-05-26 14:50:58.951	{Trailers}
9	9	Alabama Devil	A Thoughtful Panorama of a Database Administrator And a Mad Scientist who must Outgun a Mad Scientist in A Jet Boat	2006	1	3	2.99	114	21.99	PG-13	2013-05-26 14:50:58.951	{Trailers,"Deleted
10	10	Aladdin Calendar	A Action-Packed Tale of a Man And a Lumberjack who must Reach a Feminist in Ancient China	2006	1	6	4.99	63	24.99	NC-17	2013-05-26 14:50:58.951	{Trailers,"Deleted
11	11	Alamo Videotape	A Boring Epistle of a Butler And a Cat who must Fight a Pastry Chef in A MySQL Convention	2006	1	6	0.99	126	16.99	G	2013-05-26 14:50:58.951	{Commentaries,"
12	12	Alaska Phantom	A Fanciful Saga of a Hunter And a Pastry Chef who must Vanquish a Boy in Australia	2006	1	6	0.99	136	22.99	PG	2013-05-26 14:50:58.951	{Commentaries,"
13	13	Ali Forever	A Action-Packed Drama of a Dentist And a Crocodile who must Battle a Feminist in The Canadian Rockies	2006	1	4	4.99	150	21.99	PG	2013-05-26 14:50:58.951	{"Deleted Scenes
14	14	Alice Fantasia	A Emotional Drama of a A Shark And a Database Administrator who must Vanquish a Pioneer in Soviet Georgia	2006	1	6	0.99	94	23.99	NC-17	2013-05-26 14:50:58.951	{Trailers,"Deleted
15	15	Alien Center	A Brilliant Drama of a Cat And a Mad Scientist who must Battle a Feminist in A MySQL Convention	2006	1	5	2.99	46	10.99	NC-17	2013-05-26 14:50:58.951	{Trailers,Comme
16	16	Alley Evolution	A Fast-Paced Drama of a Robot And a Composer who must Battle a Astronaut in New Orleans	2006	1	6	2.99	180	23.99	NC-17	2013-05-26 14:50:58.951	{Trailers,Comme
17	17	Alone Trip	A Fast-Paced Character Study of a Composer And a Dog who must Outgun a Boat in An Abandoned Fun House	2006	1	3	0.99	82	14.99	R	2013-05-26 14:50:58.951	{Trailers,"Behind
18	18	Alter Victory	A Thoughtful Drama of a Composer And a Feminist who must Meet a Secret Agent in The Canadian Rockies	2006	1	6	0.99	57	27.99	PG-13	2013-05-26 14:50:58.951	{Trailers,"Behind
19	19	Amadeus Holy	A Emotional Display of a Pioneer And a Technical Writer who must Battle a Man in A Baloon	2006	1	6	0.99	113	20.99	PG	2013-05-26 14:50:58.951	{Commentaries,"
20	20	Amelie Hellfighters	A Boring Drama of a Woman And a Squirrel who must Conquer a Student in A Baloon	2006	1	4	4.99	79	23.99	R	2013-05-26 14:50:58.951	{Commentaries,"
21	21	American Circus	A Insightful Drama of a Girl And a Astronaut who must Face a Database Administrator in A Shark Tank	2006	1	3	4.99	129	17.99	R	2013-05-26 14:50:58.951	{Commentaries,"

Dashboard ✕ Processes ✕ Rockbuster/postgres@PostgreSQL 16* ✕

Dashboard :buster/postgres@PostgreSQL 16

No limit

Query Query History

```sql
SELECT DISTINCT
customer_id,
store_id,
first_name,
last_name,
email,
address_id,
activebool,
create_date,
last_update,
active
FROM customer;
```

Data Output Messages Notifications

	customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)	email character varying (50)	address_id smallint	activebool boolean	create_date date	last_update timestamp without time zone	active integer
1	357	1	Keith	Rico	keith.rico@sakilacustomer.org	362	true	2006-02-14	2013-05-26 14:49:45.738	1
2	171	2	Dolores	Wagner	dolores.wagner@sakilacustomer.org	175	true	2006-02-14	2013-05-26 14:49:45.738	1
3	139	1	Amber	Dixon	amber.dixon@sakilacustomer.org	143	true	2006-02-14	2013-05-26 14:49:45.738	1
4	471	1	Dean	Sauer	dean.sauer@sakilacustomer.org	476	true	2006-02-14	2013-05-26 14:49:45.738	1
5	594	1	Eduardo	Hiatt	eduardo.hiatt@sakilacustomer.org	600	true	2006-02-14	2013-05-26 14:49:45.738	1
6	401	2	Tony	Carranza	tony.carranza@sakilacustomer.org	406	true	2006-02-14	2013-05-26 14:49:45.738	1
7	157	2	Darlene	Rose	darlene.rose@sakilacustomer.org	161	true	2006-02-14	2013-05-26 14:49:45.738	1
8	154	2	Michele	Grant	michele.grant@sakilacustomer.org	158	true	2006-02-14	2013-05-26 14:49:45.738	1
9	530	2	Darryl	Ashcraft	darryl.ashcraft@sakilacustomer.org	536	true	2006-02-14	2013-05-26 14:49:45.738	1
10	493	1	Brent	Harkins	brent.harkins@sakilacustomer.org	498	true	2006-02-14	2013-05-26 14:49:45.738	1
11	542	2	Lonnie	Tirado	lonnie.tirado@sakilacustomer.org	548	true	2006-02-14	2013-05-26 14:49:45.738	1
12	566	1	Casey	Mena	casey.mena@sakilacustomer.org	572	true	2006-02-14	2013-05-26 14:49:45.738	1
13	186	2	Holly	Fox	holly.fox@sakilacustomer.org	190	true	2006-02-14	2013-05-26 14:49:45.738	1
14	128	1	Marjorie	Tucker	marjorie.tucker@sakilacustomer.org	132	true	2006-02-14	2013-05-26 14:49:45.738	1
15	466	1	Leo	Ebert	leo.ebert@sakilacustomer.org	471	true	2006-02-14	2013-05-26 14:49:45.738	1
16	494	2	Ramon	Choate	ramon.choate@sakilacustomer.org	499	true	2006-02-14	2013-05-26 14:49:45.738	1
17	178	2	Marion	Snyder	marion.snyder@sakilacustomer.org	182	true	2006-02-14	2013-05-26 14:49:45.738	1
18	65	2	Rose	Howard	rose.howard@sakilacustomer.org	69	true	2006-02-14	2013-05-26 14:49:45.738	1
19	450	1	Jay	Robb	jay.robb@sakilacustomer.org	455	true	2006-02-14	2013-05-26 14:49:45.738	1
20	234	1	Claudia	Fuller	claudia.fuller@sakilacustomer.org	238	true	2006-02-14	2013-05-26 14:49:45.738	1
21	343	1	Douglas	Graf	douglas.graf@sakilacustomer.org	348	true	2006-02-14	2013-05-26 14:49:45.738	1
22	288	1	Bobbie	Craig	bobbie.craig@sakilacustomer.org	293	true	2006-02-14	2013-05-26 14:49:45.738	1

c. **Missing data:** There is no missing data in both the Film table and the Customer table

Dashboard × Processes × Rockbuster/postgres@PostgreSQL 16* ×

Rockbuster/postgres@PostgreSQL 16

No limit

Query Query History

```sql
SELECT *
FROM film
WHERE
film_id IS NULL OR
title IS NULL OR
description IS NULL OR
release_year IS NULL OR
language_id IS NULL OR
rental_duration IS NULL OR
rental_rate IS NULL OR
length IS NULL OR
replacement_cost IS NULL OR
rating IS NULL OR
last_update IS NULL OR
special_features IS NULL OR
fulltext is NULL;
```

Data Output Messages Notifications

film_id [PK] integer	title character varying (255)	description text	release_year integer	language_id smallint	rental_duration smallint	rental_rate numeric (4,2)	length smallint	replacement_cost numeric (5,2)	rating mpaa_rating	last_update timestamp without time zone	special_features text[]	fulltext tsvector

Dashboard × Processes × Rockbuster/postgres@PostgreSQL 16* ×

Rockbuster/postgres@PostgreSQL 16

No limit

Query Query History

```sql
SELECT *
FROM customer
WHERE
customer_id IS NULL OR
store_id IS NULL OR
first_name IS NULL OR
last_name IS NULL OR
Email IS NULL OR
address_id IS NULL OR
activebool IS NULL OR
create_date IS NULL OR
last_update IS NULL OR
active IS NULL;
```

Data Output Messages Notifications

customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)	email character varying (50)	address_id smallint	activebool boolean	create_date date	last_update timestamp without time zone	active integer

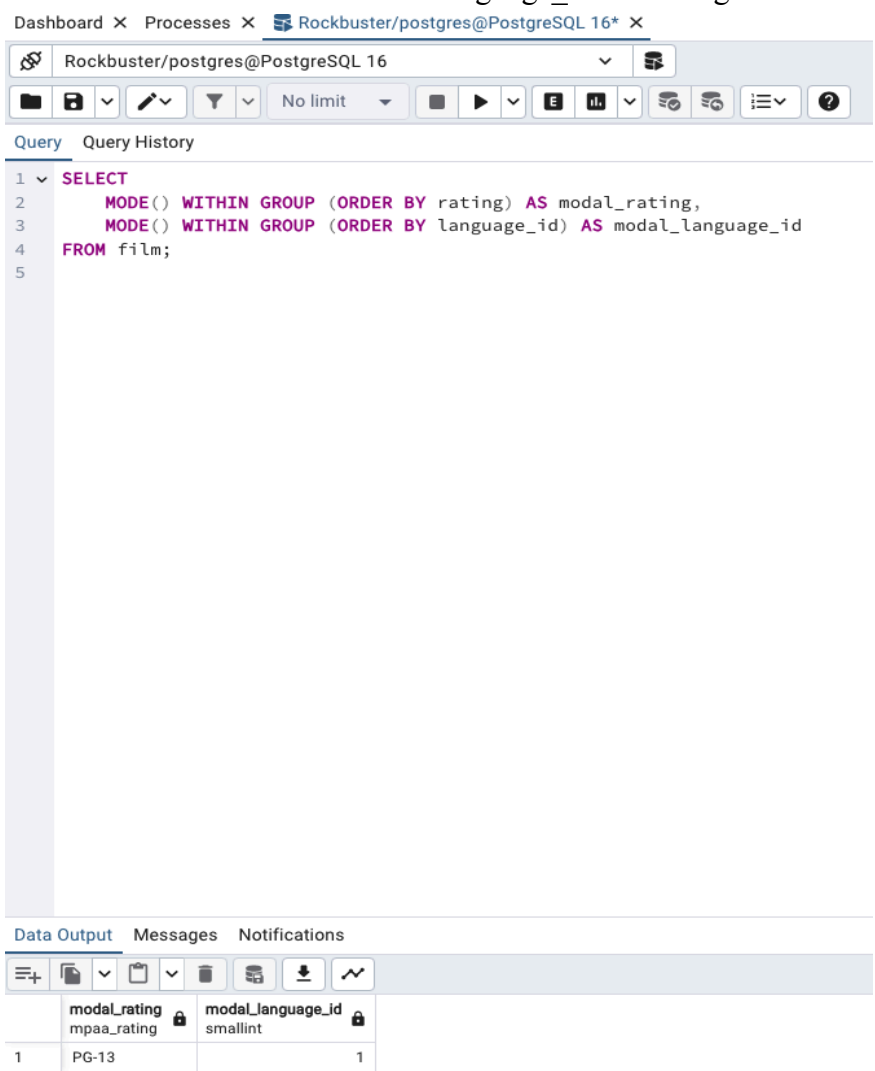
2. Summarize your data using statistics

a. **Film table numeric values:** release_year, rental_duration, rental_rate, and replacement_cost.

```
SELECT
    MIN(release_year) AS release_year,
    MAX(release_year) AS release_year,
    AVG(release_year) AS release_year,
    COUNT(release_year) AS release_year,
    MIN(rental_duration) AS rental_duration,
    MAX(rental_duration) AS rental_duration,
    AVG(rental_duration) AS rental_duration,
    COUNT(rental_duration) AS rental_duration,
    MIN(rental_rate) AS min_rent,
    MAX(rental_rate) AS max_rent,
    AVG(rental_rate) AS avg_rent,
    COUNT(rental_rate) AS rental_rate,
    MIN(replacement_cost) AS replacement_cost,
    MAX(replacement_cost) AS replacement_cost,
    AVG(replacement_cost) AS replacement_cost,
    COUNT(replacement_cost) AS replacement_cost
FROM film;
```

	release_year integer	release_year integer	release_year numeric	release_year bigint	rental_duration smallint	rental_duration smallint	rental_duration numeric	rental_duration bigint	min_rent numeric	max_rent numeric	avg_rent numeric	rental_rate bigint	replacement_cost numeric	replacement_cost numeric	replacement_cost numeric	replacement_cost bigint
1	2006	2006	2006.0000000000000000	1000	3	7	4.9850000000000000	1000	0.99	4.99	2.9800000000000000	1000	9.99	29.99	19.9840000000000000	1000

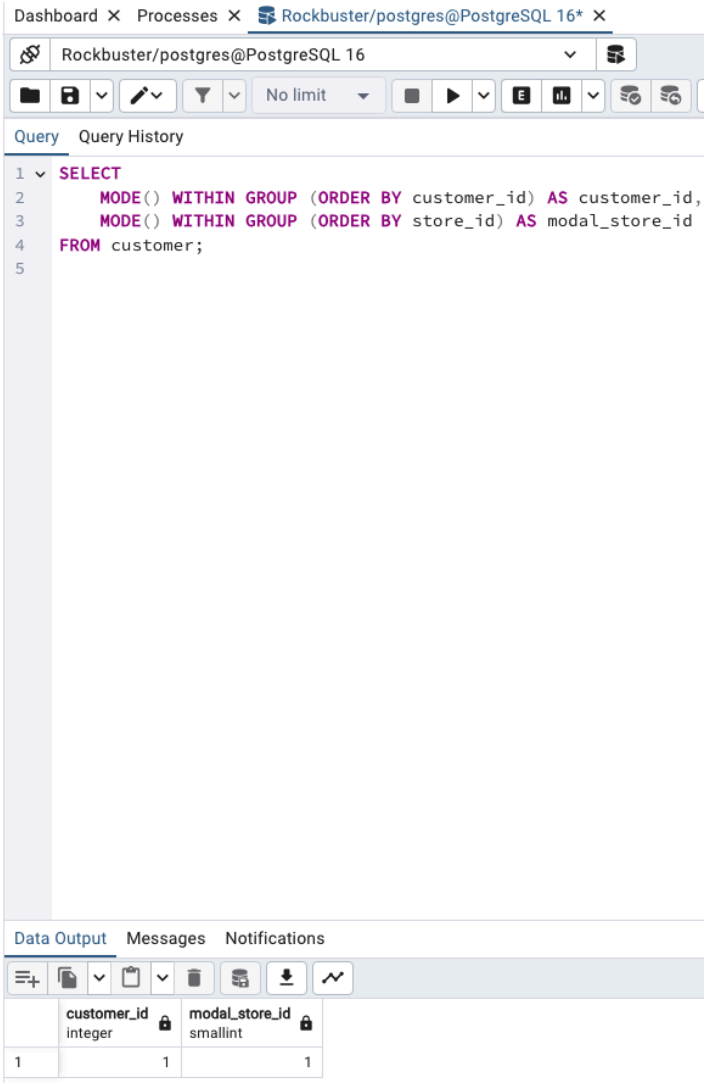
b. **Film table non-numeric values:** language_id and rating



```
SELECT
    MODE() WITHIN GROUP (ORDER BY rating) AS modal_rating,
    MODE() WITHIN GROUP (ORDER BY language_id) AS modal_language_id
FROM film;
```

	modal_rating mpaa_rating	modal_language_id smallint
1	PG-13	1

c. **Customer table non-numeric values**: customer_id and store_id



3. **Reflection:** I find SQL to be faster and more efficient overall when it comes to cleaning dirty data and statistical data summary. What I prefer with SQL over Excel is the ability to immediately errors when performing a cleaning task. In excel the output of my results may not necessarily be what is correct. The only instance that I find Excel a bit easier is identifying missing, incorrect, or non uniform data. By clicking the small arrows on the filtered columns, I am able to quickly identify what values stand out from the crowd. However, with SQL, I have to manually sift through the data values for anomalies.