3.3: SQL for Data Analysts

Step 1:

QUERY:

SELECT category_id, name

FROM category;

OUTPUT

category_id	name
1	Action
2	Animation
3	Children
4	Classics
5	Comedy
6	Documentary
7	Drama
8	Family
9	Foreign
10	Games
11	Horror
12	Music
13	New
14	Sci-Fi
15	Sports
16	Travel

Step 2:

ROCKBUSTER/postgres@PostgreSQL 18

Query   Query History   Scratch Pad ✕

```
1   INSERT INTO category(name)VALUES
2   ('Thriller'),
3   ('Crime'),
4   ('Mystery'),
5   ('Romance'),
6   ('War')
7
8
9
10
```

Data Output   Messages   Notifications

```
INSERT 0 5

Query returned successfully in 49 msec.
```

Total rows: Query complete 00:00:00.049

Q Search

Data Output   Messages   Notifications

Showing rows: 1 to 21

	name character varying (25)
1	Thriller
2	Sci-Fi
3	Comedy
4	War
5	Family
6	Games
7	Crime
8	Animation
9	Romance
10	Documentary
11	Classics
12	Sports
13	New
14	Mystery
15	Children
16	Music
17	Travel
18	Foreign
19	Horror
20	Drama
21	Action

KBUSTER
asts
atalogs
vent Triggers
xtensions
oreign Data Wrappers
anguages
ublications
chemas (1)
public
   Aggregates
   Collations
   Domains
   FTS Configurations
   FTS Dictionaries
Aa FTS Parsers
   FTS Templates
   Foreign Tables
   Functions
   Materialized Views
   Operators
   Procedures
1..3 Sequences
   Tables (15)
   > actor
   > address
   > category

Write a short paragraph explaining the various constraints that have been applied to the columns. What do these constraints do exactly?
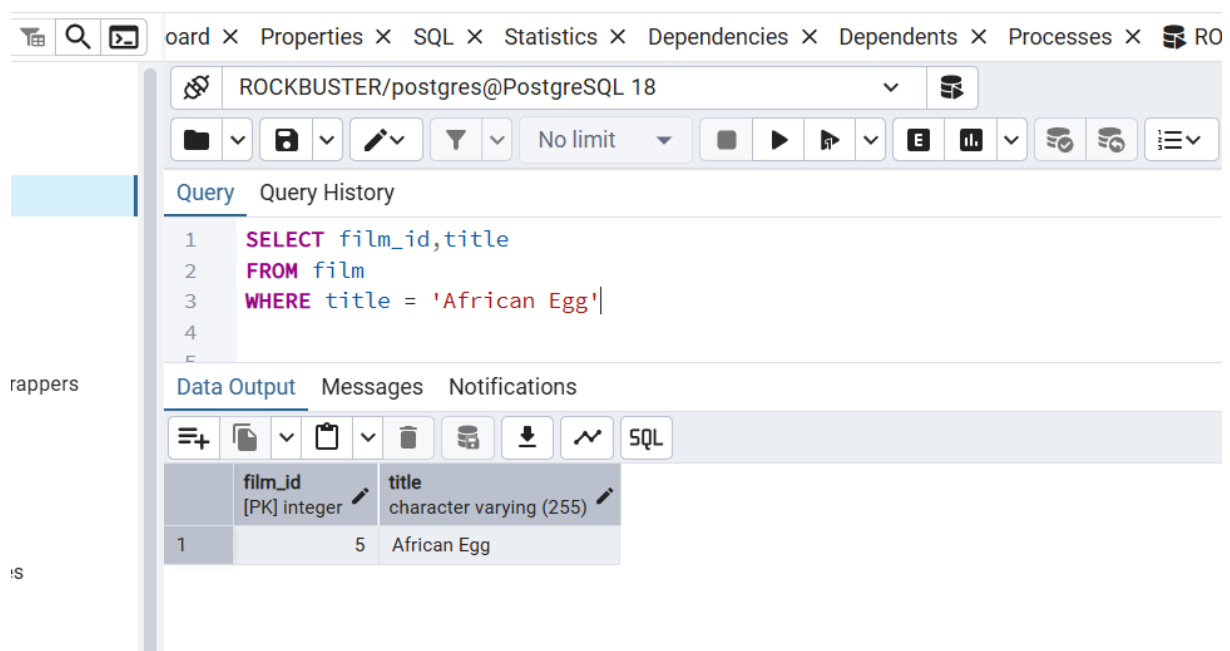
The primary key: the primary key column can't contain any null or duplicate values.

NOT NULL Constraint: This ensures that the column of catogry_id can't have any empty or missing values

Why are they important? Because it keeps the data organized and in making it easier and faster to query the database.

Step 3:

Write the SELECT statement to find the film_id for the movie *African Egg*.

Once you have the film_ID and category_ID, write an UPDATE command to change the category in the film_category table

ncies ✕   Dependents ✕   Processes ✕   **⬛ ROCKBUSTER/pos… ✕**   **⬛ ROCKBUSTER/pos… ✕**   **⬛ ROCKBUSTER/postgr**

ROCKBUSTER/postgres@PostgreSQL 18

Query   Query History

```
1    UPDATE film_category
2    SET category_id = 17
3    WHERE film_id = 5;
4
```

Data Output   Messages   Notifications

```
UPDATE 1

Query returned successfully in 43 msec.
```

18
(3)
JSTER
s
logs
t Triggers
nsions
gn Data Wrappers
uages
ications
mas (1)
blic
Aggregates
Collations
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates

Step 4:

Query   Query History

```
1    SELECT category_id, name
2    FROM category
3    WHERE category_id = 19;
4
```

Data Output   Messages   Notifications

category_id [PK] integer	name character varying (25)
19	Mystery

```
1   DELETE FROM category
2   Where category_id=19 |
3
```

Data Output   Messages   Notifications

```
DELETE 1

Query returned successfully in 59 msec.
```
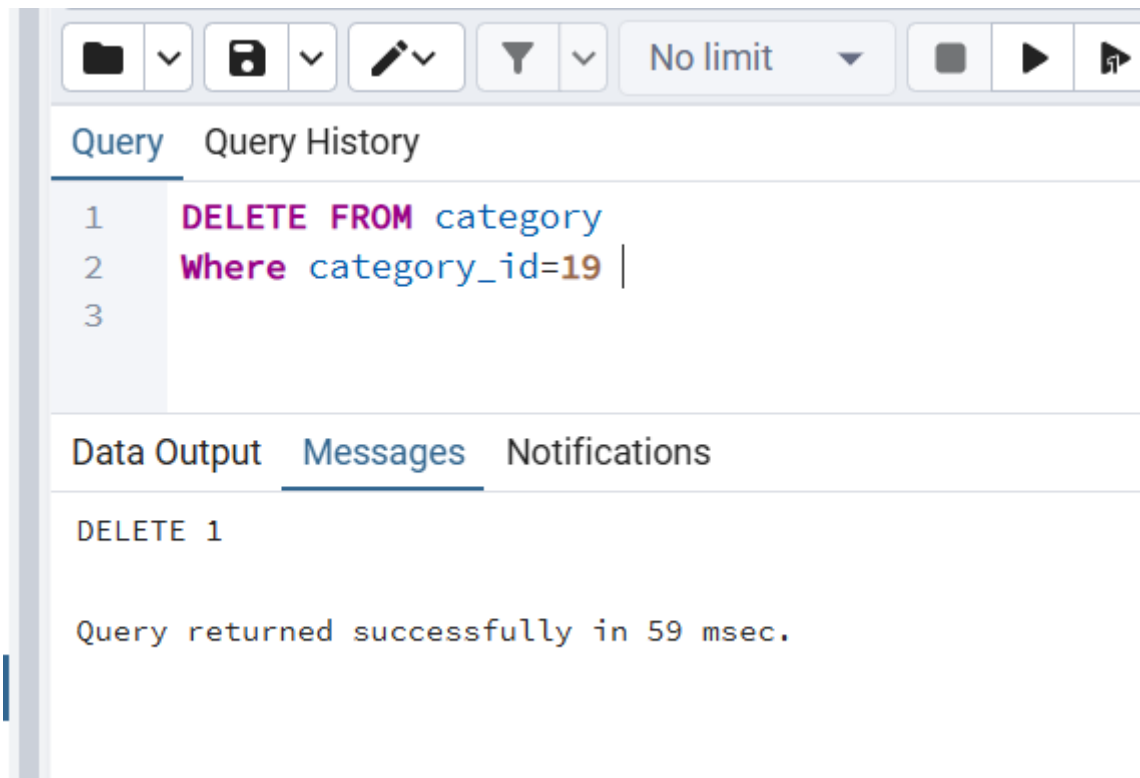
Step 5:

Pros of SQL: SQL efficiently handles millions of rows, allows for complex queries and joins, and makes multi-user collaboration possible.

Cons of SQL: Its syntax is more difficult to learn, it does not provide an intuitive graphical interface, and there is no automatic "Undo" function.

Pros of Excel: Excel offers an intuitive interface, quick visualization through charts, pivot tables, and filters, and flexibility since you can mix text, numbers, and formulas in the same worksheet.

Cons of Excel: It has a volume limit that makes it difficult to manage beyond a few hundred thousand rows, carries a risk of errors where a bad formula or a copy-paste mistake can corrupt the entire file, is not designed for relational databases, and becomes slow with complex calculations or large files.