

---

## Answers 3.3\_Innocent Bayai

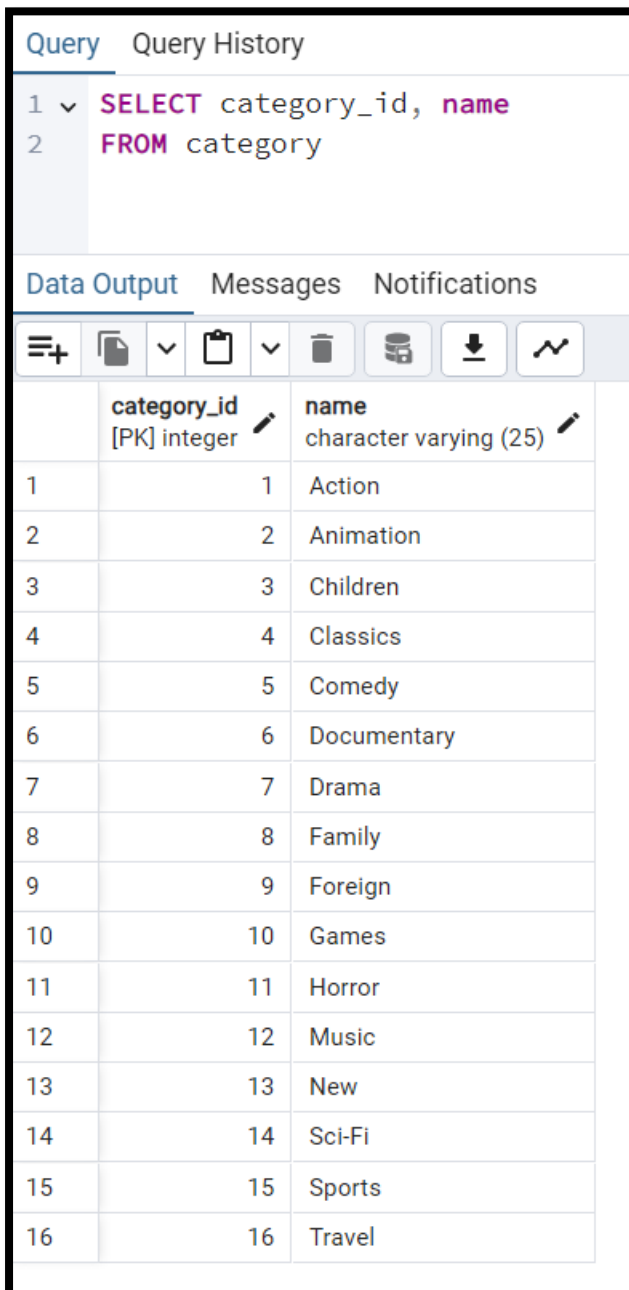
### SQL for Data Analysts

---

#### Step 1.s

Write a SELECT command to find out what film genres exist in the category table.

```
SELECT category_id, name  
FROM category
```



The screenshot shows a database query interface. At the top, there are tabs for 'Query' and 'Query History'. Below the tabs, the SQL query is displayed:   
1 SELECT category\_id, name  
2 FROM category  
Below the query, there are tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with 16 rows. The table has two columns: 'category\_id' (integer, primary key) and 'name' (character varying (25)). The rows list film genres from 1 to 16.

	category_id [PK] integer	name character varying (25)
1	1	Action
2	2	Animation
3	3	Children
4	4	Classics
5	5	Comedy
6	6	Documentary
7	7	Drama
8	8	Family
9	9	Foreign
10	10	Games
11	11	Horror
12	12	Music
13	13	New
14	14	Sci-Fi
15	15	Sports
16	16	Travel

---

#### Step 2.

- You're ready to add some new genres! Write an INSERT statement to add the following genres to the category table: Thriller, Crime, Mystery, Romance, and War

INSERT INTO category (name)

VALUES ('crime'),('thriller'),('mystery'),('romance'), ('war')

SELECT\*

FROM category

Query Query History			
1	▼	INSERT INTO category (name)	
2		VALUES ('crime'),('thriller'),('mystery'),('romance'), ('war')	
3			
4		SELECT*	
5		FROM category	
Data Output Messages Notifications			
<div><div>≡+</div><div>📄</div><div>▼</div><div>📋</div><div>▼</div><div>🗑️</div><div>🗄️</div><div>⬇️</div><div>📈</div></div>			
		category_id [PK] integer	name character varying (25)
			last_update timestamp without time zone
6		6	Documentary
7		7	Drama
8		8	Family
9		9	Foreign
10		10	Games
11		11	Horror
12		12	Music
13		13	New
14		14	Sci-Fi
15		15	Sports
16		16	Travel
17		31	crime
18		32	thriller
19		33	mystery
20		34	romance
21		35	war
Total rows: 21 of 21		Query complete 00:00:00.105	

2. b. Write a short paragraph explaining the various constraints that have been applied to the columns. What do these constraints do exactly? Why are they important?

CREATE TABLE category

(

```

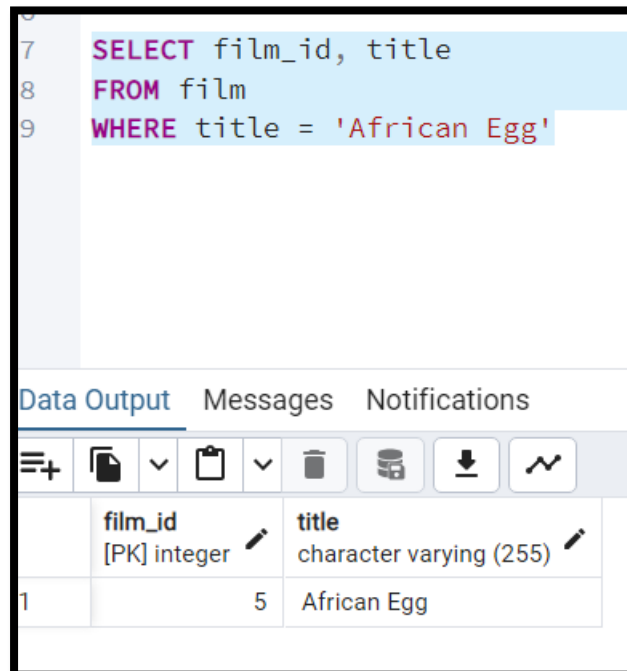
category_id integer NOT NULL DEFAULT nextval('category_category_id_seq'::regclass),
name text COLLATE pg_catalog."default" NOT NULL,
last_update timestamp with time zone NOT NULL DEFAULT now(),
CONSTRAINT category_pkey PRIMARY KEY (category_id)
);

```

- i. In creating or updating a record in the table, the **category id** cannot be null as the **NOT NULL DEFAULT** constraint will prompt you to enter the category id. This ensures that there are no missing IDs for each record entered.
- ii. The **PRIMARY KEY** here entered as category\_id performs the role of making all values in the column a primary key. Primary keys uniquely identify each record in a table row and cannot contain NULL values.
- iii. The last\_update timestamp cannot contain null values. This means that each time a new entry is made or an update is made, the very time of effecting those changes must be captured. This normally is automatically updated in SQL.
- iv. The name cannot be null thus there are no gaps in the name column as the constraint does not allow empty entries.

### Step 3.

- a. Write the SELECT statement to find the film\_id for the movie African Egg.



The screenshot shows a SQL query editor with the following query:

```

SELECT film_id, title
FROM film
WHERE title = 'African Egg'

```

Below the query editor, there is a 'Data Output' tab showing the results of the query. The results are displayed in a table with two columns: 'film\_id' and 'title'.

	film_id [PK] integer	title character varying (255)
1	5	African Egg

- b. Once you have the film\_ID and category\_ID, write an UPDATE command to change the category in the film\_category table (not the category table). Copy-paste this command into your answers document.
  - I need to know the film category of the film African Egg, now that I know the film\_id, hence the following commands and results.

```

11 SELECT film_id, category_id
12 FROM film_category
13 WHERE film_id = 5
14

```

Data Output Messages Notifications

	film_id [PK] smallint	category_id [PK] smallint
1	5	8

- Now that I have the film\_id and the category\_id, I can update the category\_id from 8 to thriller (category\_id 32). Results are shown hereunder.

```

15 UPDATE film_category
16 SET category_id = 32
17 WHERE film_id = 5
18
19 /*
20 Output for film African Egg, film_id 5 now under category_id 32 (thriller)
21 */
22 SELECT category_id, film_id
23 FROM film_category
24 WHERE category_id = 32
25

```

Data Output Messages Notifications

	category_id [PK] smallint	film_id [PK] smallint
1	32	5

#### Step 4.

Since there aren't many movies in the mystery category, you and your manager decide to remove it from the category table. Write a DELETE command to do so and copy-paste it into your answers document.

See results on next page. NB: there are now 20 categories instead of the initial 21 and the category mystery doesn't appear on the list of categories anymore.

```

26 v /* deleting Mystery from categories
27 */
28 DELETE FROM category
29 WHERE category_id = 33
30 v /*checking if mystery (category_id 33) is deleted
31 */
32 SELECT*
33 FROM category

```

Data Output Messages Notifications

	category_id [PK] integer	name character varying (25)	last_update timestamp without time zone
12	12	Music	2006-02-15 09:46:27
13	13	New	2006-02-15 09:46:27
14	14	Sci-Fi	2006-02-15 09:46:27
15	15	Sports	2006-02-15 09:46:27
16	16	Travel	2006-02-15 09:46:27
17	31	crime	2024-05-21 22:33:30.769329
18	32	thriller	2024-05-21 22:33:30.769329
19	34	romance	2024-05-21 22:33:30.769329
20	35	war	2024-05-21 22:33:30.769329

Total rows: 20 of 20 Query complete 00:00:00.243

### Step 5

Based on what you've learned so far, think about what it would be like to complete steps 1 to 4 with Excel instead of SQL. Are there any pros and cons to using SQL? Write a paragraph explaining your answer.

Using Excel to accomplish tasks 1-4 would have taken a lot more effort and manual edits to the database instead of using commands. Some of the Excel functions of course include the use of pivot tables as well as filters BUT, such processes become obsolete as data becomes big. SQL therefore makes it easy navigate large databases with ease. SQL also makes it easier to create, update, alter/update records whilst using commands.