Hands-on Lab: COUNT, DISTINCT, LIMIT

Estimated time needed: 30 minutes

In this lab, you will learn a few useful expressions that are used with SELECT statements. First, you will learn COUNT, which is an aggregate function that retrieves the number of rows that matches the query criteria. Next, you will learn DISTINCT, which is used to remove duplicate values from a specified result set and only return the unique values. Lastly, you will learn LIMIT, which is used for restricting the number of rows retrieved from the table.

Software used in this lab

In this lab, you will use Datasette, an open-source multi-tool for exploring and publishing data.

Database used in this lab

The database used in this lab comes from the following data set source: Film Locations in San Francisco under a PDDL: Public Domain Dedication and License.

Objectives

After completing this lab, you will be able to:

- Retrieve the number of rows that match a query criteria
- Remove duplicate values from a result set and return the unique values
- Restrict the number of rows retrieved from a table

Exploring the Database

Let us first explore the **SanFranciscoFilmLocations** database using the **Datasette** tool:

1. If the first statement listed below is not already in the Datasette textbox on the right, then copy the code below by clicking the little copy button on the bottom right of the code block and then paste it into the textbox of the Datasette tool using either **Ctrl+V** or right-click in the text box and choose **Paste**.

   ```
   SELECT * FROM FilmLocations;
   ```

2. Click **Submit Query**.

3. Now, you can scroll down the table and explore all the columns and rows of the **FilmLocations** table to get an overall idea of the table.

4. These are the column attribute descriptions from the **FilmLocations** table:

```
FilmLocations(
Title:              titles of the films,
ReleaseYear:        time of public release of the films,
Locations:          locations of San Francisco where the films were shot,
FunFacts:           funny facts about the filming locations,
ProductionCompany:  companies who produced the films,
Distributor:        companies who distributed the films,
Director:           people who directed the films,
Writer:             people who wrote the films,
Actor1:             person 1 who acted in the films,
Actor2:             person 2 who acted in the films,
Actor3:             person 3 who acted in the films
)
```

Using COUNT statement

Let us go through some examples of COUNT-related queries.

1. Suppose we want to count the number of records or rows of the "FilmLocations" table. The query for this would be:

```
SELECT COUNT(*) FROM FilmLocations;
```

Copy the code above and paste it to the query box of the Datasette tool. Then click **Submit query**.
Your output result set should look like the image below:

2. We want to count the number of locations of the films. But we also want to restrict the output result set so that we only retrieve the number of locations of the films written by a certain writer. The query for this can be written as:

```
SELECT COUNT(Locations) FROM FilmLocations WHERE Writer="James Cameron";
```

Copy the code above and paste it to the textbox of the Datasette tool. Then click **Submit query**.

Using DISTINCT statement

In this exercise, you will go through some examples of using DISTINCT in queries.

1. Assume that we want to retrieve the titles of all films in the table so that duplicates will be discarded in the output result set.

   ```
   SELECT DISTINCT Title FROM FilmLocations;
   ```

Copy the code above and paste it to the textbox of the Datasette tool. Then click **Submit query**.

Your output resultset should look like the image below:

2. We want to retrieve the count of release years of the films produced by a specific company so that duplicate release years of those films will be discarded in the count.

```
SELECT COUNT(DISTINCT ReleaseYear) FROM FilmLocations WHERE ProductionCompany="Warner Bros. Pictures";
```

Copy the code above and paste it to the textbox of the Datasette tool. Then click **Submit query**.

Your output resultset should look like the image below:

Using LIMIT statement

In this exercise, you will first go through some examples of using LIMIT in queries.

1. Retrieve only the first 25 rows from the table so that rows other than those are not in the output result set.

```
SELECT * FROM FilmLocations LIMIT 25;
```

Copy the code above and paste it to the textbox of the Datasette tool. Then click **Submit query**.

Your output resultset should look like the image below:

2. Now, we want to retrieve 15 rows from the table starting from row 11.

```
SELECT * FROM FilmLocations LIMIT 15 OFFSET 10;
```

Copy the code above and paste it to the textbox of the Datasette tool. Then click **Submit query**.

Your output resultset should look like the image below:

Practice exercises

COUNT

1. Retrieve the number of locations of the films which are directed by Woody Allen.

▼ Hint

Follow example 2 of the COUNT exercise. Use the WHERE clause comparison operator `=`, which means **Equal to**.

▼ Query Solution

```
SELECT COUNT(Locations) FROM FilmLocations WHERE Director="Woody Allen";
```

▼ Output

2. Retrieve the number of films shot at Russian Hill.

▼ Hint

Follow example 2 of the COUNT exercise. Use the WHERE clause comparison operator `=`, which means **Equal to**.

▼ Query Solution

```
SELECT Count(Title) FROM FilmLocations WHERE Locations="Russian Hill";
```

▼ Output

3. Retrieve the number of rows having a release year older than 1950 from the "FilmLocations" table.

▼ Hint

Follow example 1 of the COUNT exercise. Use the WHERE clause comparison operator `<`, which means **Less than.**

▼ Query Solution

```
SELECT Count(*) FROM FilmLocations WHERE ReleaseYear<1950;
```

▼ Output

Practice exercises

DISTINCT

1. Retrieve the names of all unique films released in the 21st century and onwards, along with their release years.

▼ Hint

Follow example 1 of DISTINCT. Use WHERE clause comparison operator `>=`, which means **Greater than or equal to.**

▼ Query Solution

```
SELECT DISTINCT Title, ReleaseYear FROM FilmLocations WHERE ReleaseYear>=2001;
```

▼ Output

2. Retrieve the directors' names and their distinct films shot at City Hall.

▼ Hint

Follow example 1 of DISTINCT. Use WHERE clause comparison operator `=`, which means **Equal to.**

▼ Query Solution

```
SELECT DISTINCT Title, Director FROM FilmLocations WHERE Locations="City Hall";
```

▼ Output

3. Retrieve the number of distributors who distributed films with the 1st actor, Clint Eastwood.

▼ Hint

Follow example 2 of DISTINCT. Use the WHERE clause comparison operator `=`, which means **Equal to.**

▼ Query Solution
``` SELECT COUNT(DISTINCT Distributor) FROM FilmLocations WHERE Actor1="Clint Eastwood"; ```
▼ Output

Practice exercises

LIMIT

1. Retrieve the names of the first 50 films.

- ▶ Hint
- ▶ Query Solution
- ▶ Output

2. Retrieve the first 10 film names released in 2015.

- ▶ Hint
- ▶ Query Solution
- ▶ Output

3. Retrieve the next 3 film names that follow after the first 5 films released in 2015.

- ▶ Hint
- ▶ Query Solution
- ▶ Output

Conclusion

Congratulations! You have completed this lab.

You are now able to:

- Use COUNT statements to determine the number of entries in a database based on filtering conditions.
- Use DISTINCT statements to determine the unique entries in a database based on filtering conditions.
- Use LIMIT statements to restrict the response to a desired set of rows based on filtering conditions.
- Use a combination of these statements to execute more complex queries on the database.

Author(s)

Sandip Saha Joy

Other Contributor(s)

Abhishek Gagneja