# Hands-on Lab: COUNT, DISTINCT, LIMIT

Estimated time needed: 35 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 dataset source: <u>Film Locations in San Francisco</u> under a <u>PDDL: Public</u> 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:

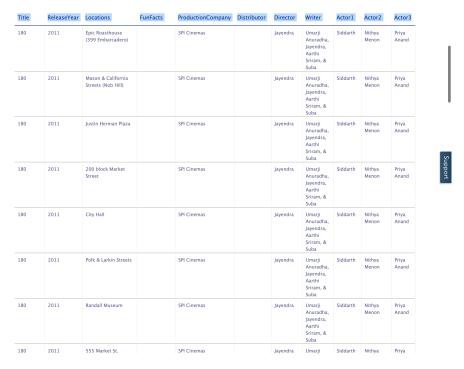
1. If the first statement listed below is not already in the Datasette textbox on the right, then copy the code below by clicking on the little copy button on the bottom right of the codeblock below and then paste it into the textbox of the Datasette tool using either Ctrl+V or right-click in the text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



2. Click Submit Query.

about:blank 1/15

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:

```
2. 2
 3. 3
 4.4
 5.5
 6.6
 7.
    7
 8.8
 9.9
10.10
11. 11
12. 12
13. 13
    FilmLocations(
 1.
        Title:
                             titles of the films,
 2.
 3.
        ReleaseYear:
                             time of public release of the films,
                             locations of San Francisco where the films were shot,
 4.
        Locations:
 5.
        FunFacts:
                             funny facts about the filming locations,
 6.
        ProductionCompany:
                             companies who produced the films,
 7.
        Distributor:
                             companies who distributed the films,
 8.
        Director:
                             people who directed the films,
 9.
                             people who wrote the films,
        Writer:
10.
        Actor1:
                             person 1 who acted in the films,
11.
        Actor2:
                             person 2 who acted in the films,
12.
        Actor3:
                             person 3 who acted in the films
13.
Copied!
```

## **Exercise 1: COUNT**

In this exercise, you will first go through some examples of using COUNT in queries and then solve some exercise problems by using it.

## Task A

about:blank 2/15

### **Example exercises on COUNT**

Let us go through some examples of COUNT related queries:

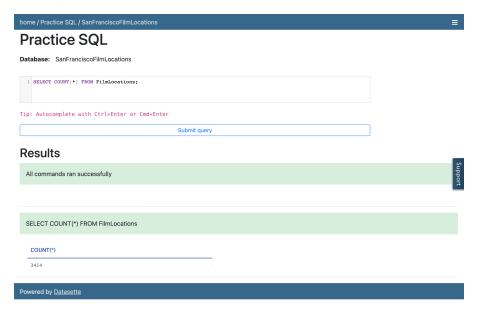
- 1. In this example, suppose we want to count the number of records or rows of the "FilmLocations" table.
  - 1. Problem:

Retrieve the number of rows from the "FilmLocations" table.

2. Solution:

```
1. 1
   1. SELECT COUNT(*) FROM FilmLocations;
Copied!
```

- 3. Copy the solution code above by clicking on the little copy button on the bottom right of the codeblock below and paste it to the textbox of the Datasette tool. Then click **Submit query**.
- 4. Your output resultset should look like the image below:



- 2. In this example, now we want to count the number of locations of the films. But we also want to restrict the output resultset in such a way that we only retrieve the number of locations of the films written by a certain writer.
  - 1. Problem:

Retrieve the number of locations of the films which are written by James Cameron.

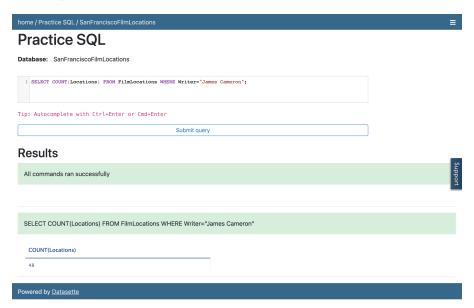
2. Solution:

```
1. 1
1. SELECT COUNT(Locations) FROM FilmLocations WHERE Writer="James Cameron";
Copied!
```

3. Copy the solution code above by clicking on the little copy button on the bottom right of the codeblock below and paste it to the textbox of the Datasette tool. Then click **Submit query**.

about:blank 3/15

4. Your output resultset should look like the image below:



# Task B

### **Practice exercises on COUNT**

Now, let us practice creating and running some COUNT related queries.

1. Problem:

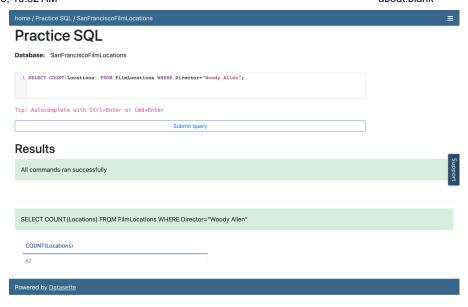
Retrieve the number of locations of the films which are directed by Woody Allen.

**▼** Hint

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

- **▼** Solution
- 1. 1
  1. SELECT COUNT(Locations) FROM FilmLocations WHERE Director="Woody Allen";
  Copied!
- **▼** Output

about:blank 4/15



#### 2. Problem:

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".

#### **▼** Solution

- 1. 1
  1. SELECT Count(Title) FROM FilmLocations WHERE Locations="Russian Hill";
  Copied!
- Nome / Practice SQL / SanFranciscoFilmLocations

  Practice SQL

  Database: SanFranciscoFilmLocations

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

  | Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

  | Submit query

  | Results

  | All commands ran successfully | SELECT Count(Title) FROM FilmLocations WHERE Locations="Russian Hill" | SELECT Count(Title) FROM FilmLocations WHERE Locations="Russian Hill" | Count(Title) | FROM FilmLocations WHERE Locations="Russian Hill" | Count(Title) | I

#### 3. Problem:

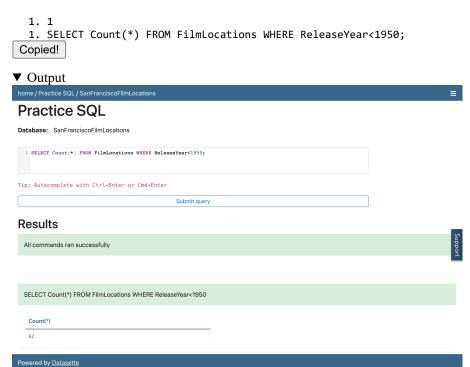
about:blank 5/15

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".

**▼** Solution



## **Exercise 2: DISTINCT**

In this exercise, you will first go through some examples of using DISTINCT in queries, and then solve some exercise problems by using it.

## Task A

# **Example exercises of DISTINCT**

Let us go through some examples of DISTINCT related queries:

- 1. In this example, we want to retrieve the title of all films in the table in such a way that duplicates will be discarded in the output resultset.
  - 1. Problem:

Retrieve the name of all films without any repeated titles.

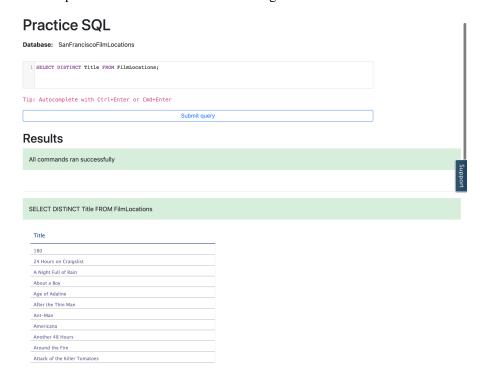
2. Solution:

- 1. 1
- SELECT DISTINCT Title FROM FilmLocations;

about:blank 6/15

Copied!

- 3. Copy the solution code above by clicking on the little copy button on the bottom right of the codeblock below and paste it to the textbox of the Datasette tool. Then click **Submit query**.
- 4. Your output resultset should look like the image below:



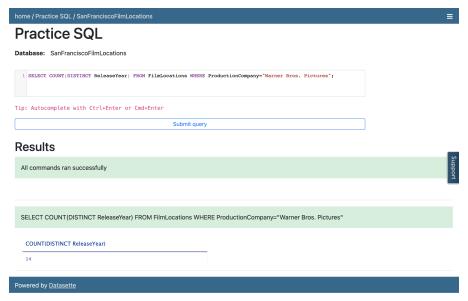
- 2. In this example, we want to retrieve the count of release years of the films produced by a specific company in such a way that duplicate release years of those films will be discarded in the count.
  - 1. Problem:

Retrieve the number of release years of the films distinctly, produced by Warner Bros. Pictures.

- 2. Solution:
  - 1. 1
    1. SELECT COUNT(DISTINCT ReleaseYear) FROM FilmLocations WHERE ProductionCompany="Warner Bros. Pictures";

    Copied!
- 3. Copy the solution code above by clicking on the little copy button on the bottom right of the codeblock below and paste it to the textbox of the Datasette tool. Then click **Submit query**.
- 4. Your output resultset should look like the image below:

about:blank 7/15



# Task B

### **Practice exercises on DISTINCT**

Now, let us practice creating and running some DISTINCT related queries.

#### 1. Problem:

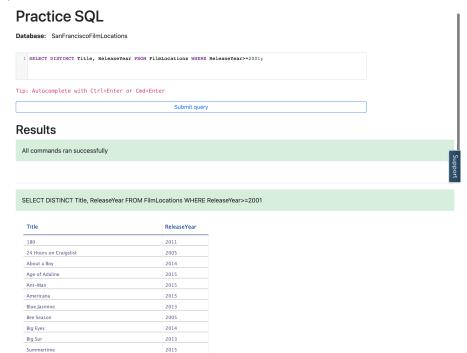
Retrieve the name of all unique films released in the 21st century and onwards, along with their release years.

**▼** Hint

Follow example 1 of DISTINCT. Use WHERE clause comparsion operator >= which means "Greater than or equal to".

- **▼** Solution
- 1. 1
  1. SELECT DISTINCT Title, ReleaseYear FROM FilmLocations WHERE ReleaseYear>=2001;
  Copied!
- **▼** Output

about:blank 8/15



#### 2. Problem:

Retrieve the names of all the directors and their distinct films shot at City Hall.

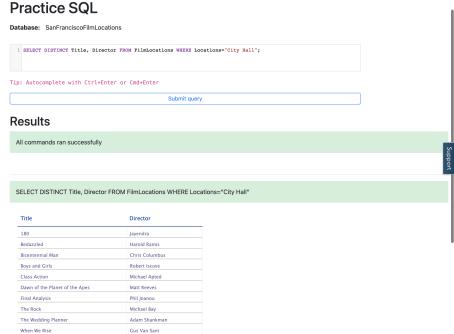
#### **▼** Hint

Follow example 1 of DISTINCT. Use WHERE clause comparsion operator = which means "Equal to".

#### **▼** Solution

- 1. 1
- 1. SELECT DISTINCT Title, Director FROM FilmLocations WHERE Locations="City Hall"; Copied!

### **▼** Output



#### 3. Problem:

The Right Stuff

Retrieve the number of distributors distinctly who distributed films acted by Clint Eastwood as 1st actor.

about:blank 9/15

**▼** Hint

Follow example 2 of DISTINCT. Use WHERE clause comparsion operator = which means "Equal to".

**▼** Solution

Solution	
<pre>1. 1    1. SELECT COUNT(DISTINCT Distributor) FROM FilmLocations WHERE Actor1="Clint Eastwo Copied!</pre>	od";
▼ Output	
home / Practice SQL / SanFranciscoFilmLocations	
Practice SQL	
Database: SanFranciscoFilmLocations	
1 SELECT COUNT(DISTINCT Distributor) FROM FilmLocations WHERE Actorl="Clint Eastwood";	
Tip: Autocomplete with Ctrl+Enter or Cmd+Enter	
Submit query	
Results	

SELECT COUNT(DISTINCT Distributor) FROM FilmLocations WHERE Actor1="Clint Eastwood"

COUNT(DISTINCT Distributor)

3

Powered by Datasette

## **Exercise 3: LIMIT**

All commands ran successfully

In this exercise, you will first go through some examples of using LIMIT in queries and then solve some exercise by using it.

### Task A: Example exercises of LIMIT

Let us go through some examples of LIMIT related queries:

- 1. In this example, let us retrieve a specific number of rows from the top of the table in such a way that rows other than those are not in the output resultset.
  - 1. Problem:

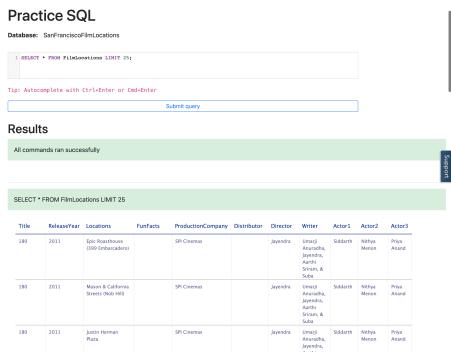
Retrieve the first 25 rows from the "FilmLocations" table.

2. Solution:

```
1. 1
1. SELECT * FROM FilmLocations LIMIT 25;
Copied!
```

- 3. Copy the solution code above by clicking on the little copy button on the bottom right of the codeblock below and paste it to the textbox of the Datasette tool. Then click **Submit query**.
- 4. Your output resultset should look like the image below:

about:blank 10/15



- 2. In this example, let us take the first example to a more advanced level. Now we want to retrieve a specific number of rows from the table, but thid time, not from the top of the table. This time we want to retrieve a specific number of rows starting from a specific row in the table.
  - 1. Problem:

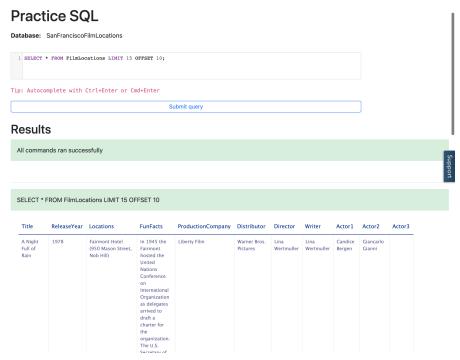
Retrieve the first 15 rows from the "FilmLocations" table starting from row 11.

2. Solution:

```
1. 1
   1. SELECT * FROM FilmLocations LIMIT 15 OFFSET 10;
Copied!
```

- 3. Copy the solution code above by clicking on the little copy button on the bottom right of the codeblock below and paste it to the textbox of the Datasette tool. Then click **Submit query**.
- 4. Your output resultset should look like the image below:

about:blank 11/15



### Task B: Practice exercises on LIMIT

Now, let us practice creating and running some LIMIT related queries.

#### 1. Problem:

Retrieve the name of first 50 films distinctly.

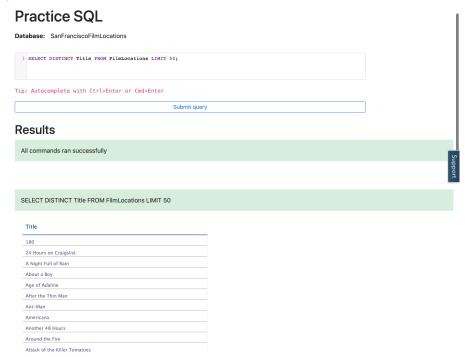
**▼** Hint

Follow example 1 of LIMIT. Use DISTINCT.

- **▼** Solution
- 1. 1
  1. SELECT DISTINCT Title FROM FilmLocations LIMIT 50;
  Copied!

**▼** Output

about:blank 12/15



#### 2. Problem:

Retrieve first 10 film names distinctly released in 2015.

#### **▼** Hint

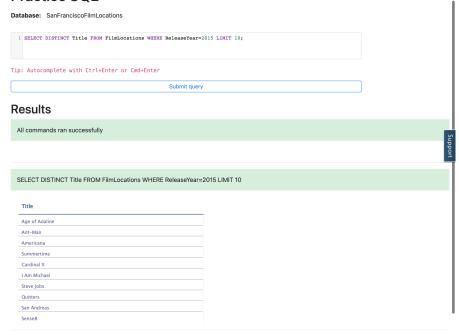
Follow example 1 of LIMIT. Use DISTINCT. Use WHERE clause comparsion operator = which means "Equal to".

#### **▼** Solution

- 1. 1
- 1. SELECT DISTINCT Title FROM FilmLocations WHERE ReleaseYear=2015 LIMIT 10; Copied!

#### **▼** Output

#### **Practice SQL**



#### 3. Problem:

about:blank 13/15

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

#### **▼** Hint

Follow example 2 of the LIMIT exercise to learn how to use OFFSET. Use DISTINCT and use the WHERE clause comparison operator = which means "Equal to".

#### **▼** Solution

Congratulations! You have completed this Lab.

## Author(s)

• Sandip Saha Joy

## Other Contributor(s)

## Changelog

Date Ve	ersion Chan	iged by	<b>Change Description</b>
2023-05-11 1.6	Eric Hao & V	<sup>7</sup> ladislav Boyko	Updated Page Frames
2023-05-10 1.5	Eric Hao & V	<sup>7</sup> ladislav Boyko	Updated Page Frames
2023-05-10 1.4	Eric Hao & V	ladislav Boyko	Updated Page Frames
2023-05-05 1.3	Benny Li		Reformatted and republished
2022-07-27 1.2	2 Lakshmi Holl	la	Updated html tag
2020-12-23 1.1	Steve Ryan		ID Review

about:blank 14/15

DateVersionChanged byChange Description2020-11-241.0Sandip Saha JoyInitial version created

© IBM Corporation 2023. All rights reserved.

about:blank 15/15