



**IBM Developer
SKILLS NETWORK**

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: [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 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**.

```
SELECT * FROM FilmLocations;
```

home / Practice SQL / SanFranciscoFilmLocations

Practice SQL

Database: SanFranciscoFilmLocations

```
1 SELECT * FROM FilmLocations;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

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.

Title	ReleaseYear	Locations	FunFacts	ProductionCompany	Distributor	Director	Writer	Actor1	Actor2	Actor3
180	2011	Epic Roasthouse (399 Embarcadero)		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Mason & California Streets (Nob Hill)		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Justin Herman Plaza		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	200 block Market Street		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	City Hall		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Polk & Larkin Streets		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Randall Museum		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	555 Market St.		SPI Cinemas		Jayendra	Umarji	Siddarth	Nithya	Priya

Support

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
)
```

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: 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:

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

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:

The screenshot shows the Datasette SQL practice tool interface. At the top, there is a navigation bar with the text "home / Practice SQL / SanFranciscoFilmLocations" and a hamburger menu icon. Below this is the title "Practice SQL" and the database name "Database: SanFranciscoFilmLocations". A code editor contains the SQL query: `1 SELECT COUNT(*) FROM FilmLocations;`. Below the code editor is a tip: "Tip: Autocomplete with Ctrl+Enter or Cmd+Enter". A "Submit query" button is located below the tip. The "Results" section shows a green message: "All commands ran successfully". Below this, the query is repeated: `SELECT COUNT(*) FROM FilmLocations`. The result set shows a single column header "COUNT(*)" and a single row with the value "3414". A "Support" button is located on the right side of the results section. At the bottom, there is a footer that says "Powered by Datasette".

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:

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

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:

The screenshot shows the Datasette SQL practice tool interface. At the top, the breadcrumb is 'home / Practice SQL / SanFranciscoFilmLocations'. The main heading is 'Practice SQL'. Below it, the database is identified as 'SanFranciscoFilmLocations'. A code editor contains the SQL query: `1 SELECT COUNT(Locations) FROM FilmLocations WHERE Writer="James Cameron";`. A tip below the editor says: 'Tip: Autocomplete with Ctrl+Enter or Cmd+Enter'. A 'Submit query' button is present. The 'Results' section shows a green status bar: 'All commands ran successfully'. Below this, the query is repeated: `SELECT COUNT(Locations) FROM FilmLocations WHERE Writer="James Cameron"`. The result set has one column, 'COUNT(Locations)', and one row with the value '48'. A 'Support' button is on the right. The footer says 'Powered by Datasette'.

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

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

▼ Output

home / Practice SQL / SanFranciscoFilmLocations

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

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

COUNT(Locations)
62

Powered by [Datasette](#)

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

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

▼ Output

home / Practice SQL / SanFranciscoFilmLocations

Practice SQL

Database: SanFranciscoFilmLocations

```
1 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"

Count(Title)
1

Powered by Datasette

3. Problem:

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

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

▼ Output

[home](#) / [Practice SQL](#) / [SanFranciscoFilmLocations](#)

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

[Submit query](#)

Results

All commands ran successfully

[Support](#)

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

Count(*)
62

Powered by [Datasette](#)

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:

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

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:

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

```
SELECT DISTINCT Title FROM FilmLocations
```

Title

180
24 Hours on Craigslist
A Night Full of Rain
About a Boy
Age of Adaline
After the Thin Man
Ant-Man
Americana
Another 48 Hours
Around the Fire
Attack of the Killer Tomatoes

Support

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:

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

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:

The screenshot shows the Datasette SQL practice tool interface. At the top, there's a breadcrumb trail: "home / Practice SQL / SanFranciscoFilmLocations". Below this is the title "Practice SQL" and the database name "Database: SanFranciscoFilmLocations". A code editor contains the SQL query: `1 SELECT COUNT(DISTINCT ReleaseYear) FROM FilmLocations WHERE ProductionCompany="Warner Bros. Pictures";`. A tip below the editor says: "Tip: Autocomplete with Ctrl+Enter or Cmd+Enter". A "Submit query" button is visible. The "Results" section shows a green message: "All commands ran successfully". Below this, the executed query is shown: `SELECT COUNT(DISTINCT ReleaseYear) FROM FilmLocations WHERE ProductionCompany="Warner Bros. Pictures"`. The result set is a single row with the column header `COUNT(DISTINCT ReleaseYear)` and the value `14`. A "Support" button is on the right. The footer says "Powered by Datasette".

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 comparison operator `>=` which means "**Greater than or equal to**".

▼ Solution

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

▼ Output

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

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

Title	ReleaseYear
180	2011
24 Hours on Craigslist	2005
About a Boy	2014
Age of Adaline	2015
Ant-Man	2015
Americana	2015
Blue Jasmine	2013
Bee Season	2005
Big Eyes	2014
Big Sur	2013
Summertime	2015

Support

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 comparison operator `=` which means "**Equal to**".

▼ Solution

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

▼ Output

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

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

Title	Director
180	Jayendra
Bedazzled	Harold Ramis
Bicentennial Man	Chris Columbus
Boys and Girls	Robert Iscove
Class Action	Michael Apted
Dawn of the Planet of the Apes	Matt Reeves
Final Analysis	Phil Joanou
The Rock	Michael Bay
The Wedding Planner	Adam Shankman
When We Rise	Gus Van Sant
The Right Stuff	Philip Kaufman

Support

3. Problem:

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

▼ Hint

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

▼ Solution

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

▼ Output

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

Support

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

```
COUNT(DISTINCT Distributor)
```

```
3
```

Powered by [Datasette](#)

Exercise 3: LIMIT

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:

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

- 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**.
- Your output resultset should look like the image below:

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

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

Title	ReleaseYear	Locations	FunFacts	ProductionCompany	Distributor	Director	Writer	Actor1	Actor2	Actor3
180	2011	Epic Roasthouse (399 Embarcadero)		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Mason & California Streets (Nob Hill)		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Justin Herman Plaza		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand

- 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 this 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.

- Problem:

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

- Solution:

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

- 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 Datasets tool. Then click **Submit query**.

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

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

SELECT * FROM FilmLocations LIMIT 15 OFFSET 10

Title	ReleaseYear	Locations	FunFacts	ProductionCompany	Distributor	Director	Writer	Actor1	Actor2	Actor3
A Night Full of Rain	1978	Fairmont Hotel (950 Mason Street, Nob Hill)	In 1945 the Fairmont hosted the United Nations Conference on International Organization as delegates arrived to draft a charter for the organization. The U.S. Secretary of	Liberty Film	Warner Bros. Pictures	Lina Wertmuller	Lina Wertmuller	Candice Bergen	Giancarlo Gianni	

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

```
SELECT DISTINCT Title FROM FilmLocations LIMIT 50;
```

▼ Output

Practice SQL

Database: SanFranciscoFilmLocations

```
1 SELECT DISTINCT Title FROM FilmLocations LIMIT 50;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

```
SELECT DISTINCT Title FROM FilmLocations LIMIT 50
```

Title

180
24 Hours on Craigslist
A Night Full of Rain
About a Boy
Age of Adaline
After the Thin Man
Ant-Man
Americana
Another 48 Hours
Around the Fire
Attack of the Killer Tomatoes

Support

2. Problem:

Retrieve first 10 film names distinctly released in 2015.

▼ Hint

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

▼ Solution

```
SELECT DISTINCT Title FROM FilmLocations WHERE ReleaseYear=2015 LIMIT 10;
```

▼ Output

Practice SQL

Database: SanFranciscoFilmLocations

```
1 SELECT DISTINCT Title FROM FilmLocations WHERE ReleaseYear=2015 LIMIT 10;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

```
SELECT DISTINCT Title FROM FilmLocations WHERE ReleaseYear=2015 LIMIT 10
```

Title

Age of Adaline
Ant-Man
Americana
Summertime
Cardinal X
I Am Michael
Steve Jobs
Quitters
San Andreas
Sense8

Support

3. Problem:

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

```
SELECT DISTINCT Title FROM FilmLocations WHERE ReleaseYear=2015 LIMIT 3 OFFSET
```

► Output

Congratulations! You have completed this Lab.

Author(s)

- [Sandip Saha Joy](#)

Other Contributor(s)

-

Changelog

Date	Version	Changed by	Change Description
2020-12-23	1.1	Steve Ryan	ID Review
2020-11-24	1.0	Sandip Saha Joy	Initial version created

© IBM Corporation 2020. All rights reserved.