Hands-on Lab: Simple SELECT Statements

Skills Network

Estimated time needed: 20 minutes

In this lab, you will learn one of the most commonly used statements of SQL (Structured Query Language), the SELECT statement. The SELECT statement is used to select data from a database.

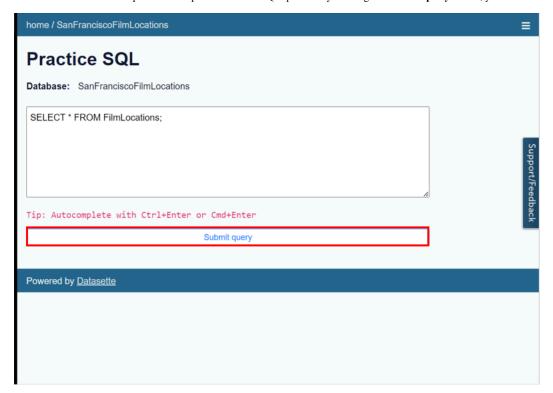## Objectives

After completing this lab, you will be able to:

- Query a database to obtain a response as a result set
- Retrieve all or selected columns of a dataset
- Apply criteria commands to filter the result set

Software Used in this Lab

In this lab, you will use Datasette, an open source-tool for exploring and publishing data. You can visit the [Datasette GitHub repository here](#).

Working with Datasette

The **Datasette** tool offers a platform to input and execute SQL queries. By clicking the **Submit query** button, you can execute the SQL query.

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](#).

Exploring the Database

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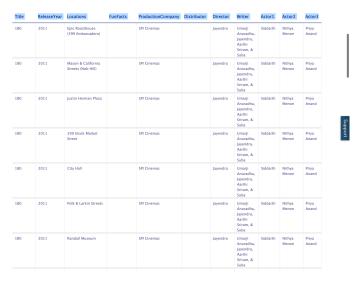
1. If the first statement listed below is not already in the Datasette textbox on the right, then copy the code below by clicking on the little copy button on the bottom right of the code block below and then paste it into the textbox of the Datasette tool using either **Ctrl+V** or right-click in the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

   1. 1

   1. SELECT * FROM FilmLocations;

Copied!

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 contents.

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

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

1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. 8
9. 9
10. 10
11. 11
12. 12
13. 13

```
1.  FilmLocations(
2.  Title:              titles of the films,
3.  ReleaseYear:        time of public release of the films,
4.  Locations:          locations of San Francisco where the films were shot,
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.  Writer:             people who wrote the films,
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!

Using SELECT statement
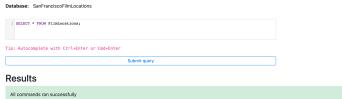
Now, let's go through some examples of SELECT queries.

1. Suppose we want to retrieve details of all the films from the **FilmLocations** table. The details of each film record should contain all the columns. The query statement for this is:

1. 1

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

Copied!

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

Practice SQL

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Your output resultset should match like the image below:

Database: SanFranciscoFilmLocations

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

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

SELECT * FROM FilmLocations

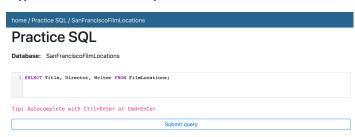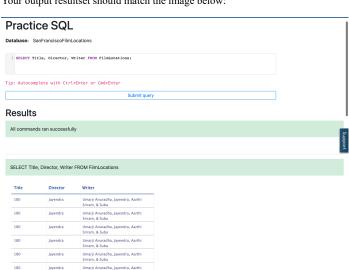| Title	ReleaseYear	Locations	FunFacts	ProductionCompany	Distributor	Director	Writer	Actor1	Actor2	Actor3
180	2011	Epic Roasthouse (399 Embarcadero)		SPI Cinemas		Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba	Siddarth	Nithya Menon	Priya Anand
180	2011	Mason & California Streets (Nob Hill)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

2. We want to retrieve the film names and director and writer names. The query now would be:

1. 1

1. SELECT Title, Director, Writer FROM FilmLocations;

Copied!

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

Practice SQL

Database: SanFranciscoFilmLocations

```
1  SELECT Title, Director, Writer FROM FilmLocations;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Your output resultset should match the image below:

Practice SQL

Database: SanFranciscoFilmLocations

```
1  SELECT Title, Director, Writer FROM FilmLocations;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

SELECT Title, Director, Writer FROM FilmLocations

Title	Director	Writer
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi Sriram, & Suba
180	Jayendra	Umarji Anuradha, Jayendra, Aarthi

3. We want to retrieve film names along with filming locations and release years. But we also want to restrict the output resultset to retrieve only the film records released in 2001 and onwards (release years after 2001, including 2001).
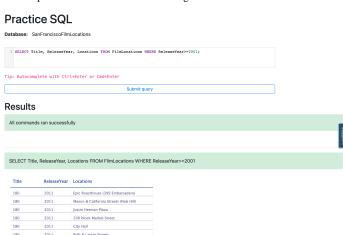
1. 1

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

Copied!

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

Your output resultset should match the image below:

Practice exercises on the SELECT statement

1. Retrieve the fun facts and filming locations of all films.

▼ Click here for a hint

Follow example 2 of SELECT, where records containing details of some particular columns have been retrieved.

▼ Click here for the solution

1. 1

1. `SELECT Locations, FunFacts FROM FilmLocations;`

Copied!

▼ Click here for the output

Practice SQL

Database: SanFranciscoFilmLocations

```
1  SELECT Locations, FunFacts FROM FilmLocations;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

	Submit query

Results

All commands ran successfully

SELECT Locations, FunFacts FROM FilmLocations

Locations	FunFacts
Epic Roasthouse (399 Embarcadero)	
Mason & California Streets (Nob Hill)	
Justin Herman Plaza	
200 block Market	

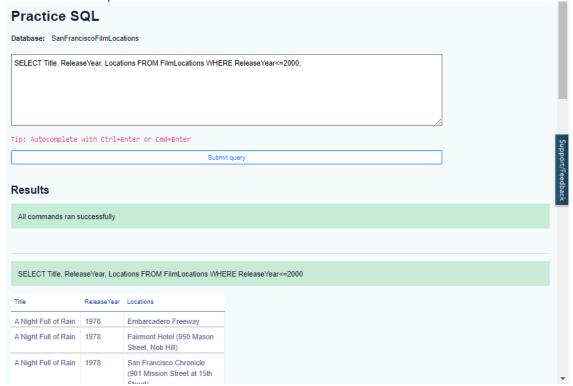|	
Street	
City Hall	
Polk & Larkin Streets	
Randall Museum	
555 Market St.	
Embarcadero	Embarcadero Freeway, which was featured in the film

2. Retrieve the names of all films released in the 20th century and before (release years before 2000 including 2000), along with filming locations and release years.

▼ Click here for a hint

Follow example 3 of SELECT, where we restricted the output resultset to retrieve only the film records with certain release years. Use WHERE clause comparison operator <=, which means **Less than or equal to**.

▼ Click here for the solution

1. 1

```
1. SELECT Title, ReleaseYear, Locations FROM FilmLocations WHERE ReleaseYear<=2000;
```

Copied!

▼ Click here for the output

Practice SQL

Database: SanFranciscoFilmLocations

```
SELECT Title, ReleaseYear, Locations FROM FilmLocations WHERE ReleaseYear<=2000;
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

SELECT Title, ReleaseYear, Locations FROM FilmLocations WHERE ReleaseYear<=2000

Title	ReleaseYear	Locations
A Night Full of Rain	1978	Embarcadero Freeway
A Night Full of Rain	1978	Fairmont Hotel (950 Mason Street, Nob Hill)
A Night Full of Rain	1978	San Francisco Chronicle (901 Mission Street at 15th Street)

3. Retrieve the names, production company names, filming locations, and release years of the films not written by James Cameron.

▼ Click here for a hint

Use WHERE clause comparison operator <>, which means **Not equal to**.

▼ Click here for the solution

1. 1

```
1. SELECT Title, ProductionCompany, Locations, ReleaseYear FROM FilmLocations WHERE Writer<>"James Cameron";
```

Copied!

▼ Click here for the output

Practice SQL

Database: SanFranciscoFilmLocations

```
1  SELECT Title, ProductionCompany, Locations, ReleaseYear FROM Fil
```

Tip: Autocomplete with Ctrl+Enter or Cmd+Enter

Submit query

Results

All commands ran successfully

SELECT Title, ProductionCompany, Locations, ReleaseYear FROM FilmLc

Title	ProductionCompany	Locations	ReleaseYear
180	SPI Cinemas	Epic Roasthouse (399 Embarcadero)	2011
180	SPI Cinemas	Mason & California Streets (Nob Hill)	2011

180	SPI Cinemas	Justin Herman Plaza	2011
180	SPI Cinemas	200 block Market Street	2011
180	SPI Cinemas	City Hall	2011
180	SPI Cinemas	Polk & Larkin Streets	2011
180	SPI Cinemas	Randall Museum	2011

Conclusion

Congratulations on completing this lab!

You are now able to:

- Query a database using SELECT statements
- Retrieve all or selected columns of data
- Filter the query response to meet a defined criteria

Author(s)

Sandip Saha Joy

Other Contributor(s)

Abhishek Gagneja

Changelog

Date	Version	Changed by	Change Description
2023-10-12	1.9	Mercedes Schneider	QA Pass w/Edits
2023-10-11	1.8	Misty Taylor	ID Check
2023-10-01	1.7	Abhishek Gagneja	Updated Lab instructions
2023-07-11	1.6	Lakshmi Holla	Updated labs
2023-06-02	1.5	Eric Hao	Fixed Page Styles
2023-05-10	1.4	Eric Hao & Vladislav Boyko	Updated Page Frames
2023-05-04	1.3	Benny Li	Republished
2022-07-27	1.2	Lakshmi Holla	Updated html tag
2020-11-23	1.1	Steve Ryan	ID Review
2020-11-20	1.0	Sandip Saha Joy	Initial version created