

Lesson 1.

SELECT queries 101

- 1.SELECT title FROM movies
- 2.SELECT director FROM movies
- 3.SELECT title,director FROM movies
- 4,SELECT title,year FROM movies
- 5.SELECT * FROM movies

Table: Movies

Id	Title	Director	Year	Length_minutes
1	Toy Story	John Lasseter	1995	81
2	A Bug's Life	John Lasseter	1998	95
3	Toy Story 2	John Lasseter	1999	93
4	Monsters, Inc.	Pete Docter	2001	92
5	Finding Nemo	Andrew Stanton	2003	107
6	The Incredibles	Brad Bird	2004	116
7	Cars	John Lasseter	2006	117
8	Ratatouille	Brad Bird	2007	115
9	WALL-E	Andrew Stanton	2008	104
10	Up	Pete Docter	2009	101

```
SELECT * FROM movies;
```

RESET

Exercise 1 – Tasks

1. Find the **title** of each film ✓
2. Find the **director** of each film ✓
3. Find the **title** and **director** of each film ✓
4. Find the **title** and **year** of each film ✓
5. Find **all** the information about each film ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 2.Queries with constraints (Pt. 1)

- 1.SELECT * FROM movies WHERE id = 6;
- 2.SELECT title FROM movies where year between 2000 AND 2010;
- 3.SELECT title FROM movies where year between 2000 AND 2010;
- 4.SELECT * FROM movies WHERE year LIMIT 5;

Table: Movies

Id	Title	Director	Year	Length_minutes
1	Toy Story	John Lasseter	1995	81
2	A Bug's Life	John Lasseter	1998	95
3	Toy Story 2	John Lasseter	1999	93
4	Monsters, Inc.	Pete Docter	2001	92
5	Finding Nemo	Andrew Stanton	2003	107

`SELECT * FROM movies WHERE year LIMIT 5;`

Exercise 2 — Tasks

- Find the movie with a row `id` of 6 ✓
- Find the movies released in the `year` s between 2000 and 2010 ✓
- Find the movies **not** released in the `year` s between 2000 and 2010 ✓
- Find the first 5 Pixar movies and their release `year` ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

RESET Continue >

Lesson 3.Queries with constraints (Pt. 2)

- 1.SELECT title FROM movies WHERE title LIKE "Toy Story%";
- 2.SELECT title FROM movies where director like 'John Lasseter';
- 3.SELECT title,director FROM movies where director not like 'John Lasseter';
- 4.SELECT title FROM movies where title like 'WALL%';

Table: Movies

Title
WALL-E
WALL-G

`SELECT title FROM movies where title like 'WALL%';`

Exercise 3 — Tasks

- Find all the Toy Story movies ✓
- Find all the movies directed by John Lasseter ✓
- Find all the movies (and director) not directed by John Lasseter ✓
- Find all the WALL-* movies ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

RESET Continue >

Lesson 4.Filtering and sorting Query results

- 1.SELECT DISTINCT director from Movies order by director
- 2.SELECT distinct title FROM movies order by year desc limit 4;
- 3.SELECT distinct title FROM movies order by title limit 5;
- 4.SELECT distinct title FROM movies order by title limit 5 offset 5;

Table: Movies

Title
Monsters University
Monsters, Inc.
Ratatouille
The Incredibles
Toy Story

Exercise 4 – Tasks

1. List all directors of Pixar movies (alphabetically), without duplicates ✓
2. List the last four Pixar movies released (ordered from most recent to least) ✓
3. List the first five Pixar movies sorted alphabetically ✓
4. List the next five Pixar movies sorted alphabetically ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue ›

RESET

Lesson 5.Simple SELECT Queries

- 1.SELECT city,population FROM north_american_cities where country='Canada'
- 2.SELECT city FROM north_american_cities where country ='United States' order by latitude desc
- 3.SELECT *FROM North_american_cities WHERE Longitude < -87.69 ORDER BY Longitude ASC;
- 4.SELECT city FROM north_american_cities where country='Mexico' order by population desc limit 2;
- 5.SELECT city FROM north_american_cities where country='United States' order by population desc limit 2 offset 2;

Table: North_american_cities

City
Chicago
Houston

Review 1 – Tasks

1. List all the Canadian cities and their populations ✓
2. Order all the cities in the United States by their latitude from north to south ✓
3. List all the cities west of Chicago, ordered from west to east ✓
4. List the two largest cities in Mexico (by population) ✓
5. List the third and fourth largest cities (by population) in the United States and their population ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue ›

RESET

Lesson 6. Multi-table queries with JOINS

1. `SELECT title,international_sales,domestic_sales FROM movies join boxoffice on id=Movie_id ;`
2. `SELECT title,International_sales, Domestic_sales FROM movies join boxoffice on id=Movie_id where international_sales > domestic_sales ;`
3. `SELECT title FROM movies join boxoffice on id=Movie_id order by Rating desc ;`

Query Results

Up
Finding Nemo
Monsters, Inc.
Ratatouille
The Incredibles
Toy Story 2
Monsters University
Cars
A Bug's Life
Brave
Cars 2

```
SELECT title FROM movies join boxoffice on id=Movie_id order by
Rating desc ;|
```

RESET

Exercise 6 – Tasks

1. Find the domestic and international sales for each movie ✓
2. Show the sales numbers for each movie that did better internationally rather than domestically ✓
3. List all the movies by their ratings in descending order ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 7. OUTER JOINS

1. `SELECT distinct Building FROM employees where years_employed is not null;`
2. `SELECT building_name,capacity FROM buildings`
3. `SELECT DISTINCT Building_name, Role FROM buildings left join employees ON Building=Building_name`

Query Results

Building_name	Role
1e	Engineer
1e	Manager
1w	
2e	
2w	Artist
2w	Manager

```
SELECT DISTINCT Building_name, Role FROM buildings left join
employees ON Building=Building_name
```

RESET

Exercise 7 – Tasks

1. Find the list of all buildings that have employees ✓
2. Find the list of all buildings and their capacity ✓
3. List all buildings and the distinct employee roles in each building (including empty buildings) ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 8.A short note on NULLs

- 1.SELECT Name,Role FROM employees where building is null;
- 2.SELECT * FROM Buildings LEFT JOIN Employees ON Building_name=Building WHERE Building IS NULL;

Building_name	Capacity	Role	Name	Building	Years_employed
1w	32				
2e	16				

```
SELECT * FROM Buildings LEFT JOIN Employees ON Building_name=Building WHERE Building IS NULL;
```

Exercise 8 – Tasks

1. Find the name and role of all employees who have not been assigned to a building ✓
2. Find the names of the buildings that hold no employees ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 9.Queries with expressions ;

- 1.SELECT Title, (Domestic_sales + International_sales)/1000000 AS Total_Sales_Millions FROM Movies LEFT JOIN Boxoffice ON Id=Movie_Id;
- 2.SELECT Title, Rating*10 as Percent FROM Movies LEFT JOIN Boxoffice ON Id=Movie_Id;
- 3.SELECT Title, Year FROM Movies LEFT JOIN Boxoffice ON Id=Movie_Id WHERE Year % 2 = 0;

Title	Year
A Bug's Life	1998
The Incredibles	2004
Cars	2006
WALL-E	2008
Toy Story 3	2010
Brave	2012

```
SELECT Title, Year FROM Movies LEFT JOIN Boxoffice ON Id=Movie_Id WHERE Year % 2 = 0;
```

Exercise 9 – Tasks

1. List all movies and their combined sales in millions of dollars ✓
2. List all movies and their ratings in percent ✓
3. List all movies that were released on even number years ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 10.Queries with aggregates (Pt. 1)

- 1.SELECT MAX(Years_employed)FROM Employees
- 2.SELECT Role, AVG(Years_Employed) FROM Employees GROUP BY Role;
- 3.SELECT Building, SUM(Years_Employed) FROM Employees GROUP BY Building;

Building	SUM(Years_Employed)
1e	29
2w	36


```
SELECT Building, SUM(Years_Employed) FROM Employees GROUP BY Building;
```

RESET

Exercise 10 – Tasks

- Find the longest time that an employee has been at the studio ✓
- For each role, find the average number of years employed by employees in that role ✓
- Find the total number of employee years worked in each building ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

[Continue >](#)

Lesson 11: Queries with aggregates (Pt. 2)

1. SELECT Role, COUNT(*) AS Number_of_Artists FROM Employees WHERE Role = "Artist";
2. SELECT Role, COUNT(*) FROM Employees GROUP BY Role;
3. SELECT Role, SUM(Years_Employed) FROM Employees GROUP BY Role having Role = "Engineer";

Table: Employees

Role	SUM(Years_Employed)
Engineer	17


```
SELECT Role, SUM(Years_Employed) FROM Employees GROUP BY Role having Role = "Engineer";
```

RESET

Exercise 11 – Tasks

- Find the number of Artists in the studio (without a HAVING clause) ✓
- Find the number of Employees of each role in the studio ✓
- Find the total number of years employed by all Engineers ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

[Continue >](#)

Lesson 12: Order of execution of a Query

1. SELECT *, COUNT(Title) FROM Movies GROUP BY Director;
2. SELECT Director, sum(Domestic_sales + International_Sales) as Total_Sales FROM Movies LEFT JOIN Boxoffice ON Id = Movie_ID GROUP BY Director;

Query Results

Director	Total_Sales
Andrew Stanton	1458055121
Brad Bird	1255164910
Brenda Chapman	538983207
Dan Scanlon	743559607
John Lasseter	2232208025
Lee Unkrich	1063171911
Pete Docter	1294159000

```

SELECT Director, sum(Domestic_sales + International_Sales) as
  Total_Sales FROM Movies LEFT JOIN Boxoffice ON Id = Movie_ID
GROUP BY Director;

```

Exercise 12 – Tasks

- Find the number of movies each director has directed ✓
- Find the total domestic and international sales that can be attributed to each director ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 13: Inserting rows

- INSERT INTO Movies(id,title,director,year,length_minutes) VALUES (4, 'Toy Story 4', 'John Lasseter', 2017, 123);
- INSERT INTO Boxoffice VALUES (4, 8.7, 340000000, 270000000);

Movie_id	Rating	Domestic_sales	International_sales
3	7.9	245852179	239163000
1	8.3	191796233	170162503
2	7.2	162798565	200600000
4	8.7	340000000	270000000

Exercise 13 – Tasks

- Add the studio's new production, Toy Story 4 to the list of movies (you can use any director) ✓
- Toy Story 4 has been released to critical acclaim! It had a rating of 8.7, and made 340 million domestically and 270 million internationally. Add the record to the BoxOffice table. ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

RUN QUERY RESET

Continue >

Lesson 14: Updating rows

- UPDATE Movies SET Director = "John Lasseter" WHERE Id = 2;
- UPDATE Movies SET Year = "1999" WHERE Id = 3;
- UPDATE Movies SET Title = "Toy Story 3", Director = "Lee Unkrich" WHERE Id = 11;

Table: Movies

Id	Title	Director	Year	Length_minutes
1	Toy Story	Lee Unkrich	1995	81
2	A Bug's Life	John Lasseter	1998	95
3	Toy Story 2	Lee Unkrich	1999	93
4	Monsters, Inc.	Pete Docter	2001	92
5	Finding Nemo	Andrew Stanton	2003	107
6	The Incredibles	Brad Bird	2004	116
7	Cars	John Lasseter	2006	117
8	Ratatouille	Brad Bird	2007	115
9	WALL-E	Andrew Stanton	2008	104
10	Up	Pete Docter	2009	101

```
UPDATE Movies SET Title = "Toy Story 3", Director = "Lee Unkrich"
WHERE Id = 11;
```

[RUN QUERY](#) [RESET](#)

Exercise 14 – Tasks

1. The director for A Bug's Life is incorrect, it was actually directed by John Lasseter ✓
2. The year that Toy Story 2 was released is incorrect, it was actually released in 1999 ✓
3. Both the title and director for Toy Story 8 is incorrect! The title should be "Toy Story 3" and it was directed by Lee Unkrich ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

[Continue >](#)

Lesson 15: Deleting rows

- 1.DELETE FROM Movies WHERE Year < 2005;
- 2.DELETE FROM Movies WHERE director='Andrew Stanton';

Table: Movies

Id	Title	Director	Year	Length_minutes
7	Cars	John Lasseter	2006	117
8	Ratatouille	Brad Bird	2007	115
10	Up	Pete Docter	2009	101
11	Toy Story 3	Lee Unkrich	2010	103
12	Cars 2	John Lasseter	2011	120
13	Brave	Brenda Chapman	2012	102
14	Monsters University	Dan Scanlon	2013	110

Exercise 15 – Tasks

1. This database is getting too big, lets remove all movies that were released before 2005. ✓
2. Andrew Stanton has also left the studio, so please remove all movies directed by him. ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

[Continue >](#)

Lesson 16:Creating tables

- 1.CREATE TABLE Database (
 Name TEXT,
 Version FLOAT,
 Download_Count INTEGER);

Table: Database

Name	Version	Download_Count
SQLite	3.9	92000000
MySQL	5.5	512000000
Postgres	9.4	384000000

Incomplete SQL query

```
CREATE TABLE Database (
  Name TEXT,
  Version FLOAT,
  Download_Count INTEGER);
```

RUN QUERY RESET

Exercise 16 – Tasks

- Create a new table named **Database** with the following columns:
 - **Name** A string (text) describing the name of the database
 - **Version** A number (floating point) of the latest version of this database
 - **Download_count** An integer count of the number of times this database was downloaded

This table has no constraints. ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 17: Altering tables

- ALTER TABLE Movies ADD COLUMN Aspect_ratio FLOAT DEFAULT 3;
- ALTER TABLE Movies ADD COLUMN Language TEXT DEFAULT "English";

Table: Movies

Id	Title	Director	Year	Length_minutes	Aspect_ratio	Language
1	Toy Story	John Lasseter	1995	81	3	English
2	A Bug's Life	John Lasseter	1998	95	3	English
3	Toy Story 2	John Lasseter	1999	93	3	English
4	Monsters, Inc.	Pete Docter	2001	92	3	English
5	Finding Nemo	Andrew Stanton	2003	107	3	English
6	The Incredibles	Brad Bird	2004	116	3	English
7	Cars	John Lasseter	2006	117	3	English
8	Ratatouille	Brad Bird	2007	115	3	English
9	WALL-E	Andrew Stanton	2008	104	3	English
10	Up	Pete Docter	2009	101	3	English

|

RUN QUERY RESET

Exercise 17 – Tasks

- Add a column named **Aspect_ratio** with a **FLOAT** data type to store the aspect-ratio each movie was released in. ✓
- Add another column named **Language** with a **TEXT** data type to store the language that the movie was released in. Ensure that the default for this language is **English**. ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

Continue >

Lesson 18: Dropping tables

- drop table Movies
- drop table boxoffice

Query Results

Id	Title	Director	Year	Length_minutes
----	-------	----------	------	----------------

Exercise 18 — Tasks

1. We've sadly reached the end of our lessons, lets clean up by removing the **Movies** table ✓
2. And drop the **BoxOffice** table as well ✓

Stuck? Read this task's [Solution](#).
Solve all tasks to continue to the next lesson.

[RUN QUERY](#) [RESET](#)

[Continue >](#)