Exercise 1

A screenshot of a computer

Description automatically generated

Query

SELECT Title FROM movies;

SELECT director From movies;

SELECT Title, director From movies;

SELECT Title, year FROM movies;

SELECT \* FROM movies;

Exercise 2

A screenshot of a computer

Description automatically generated

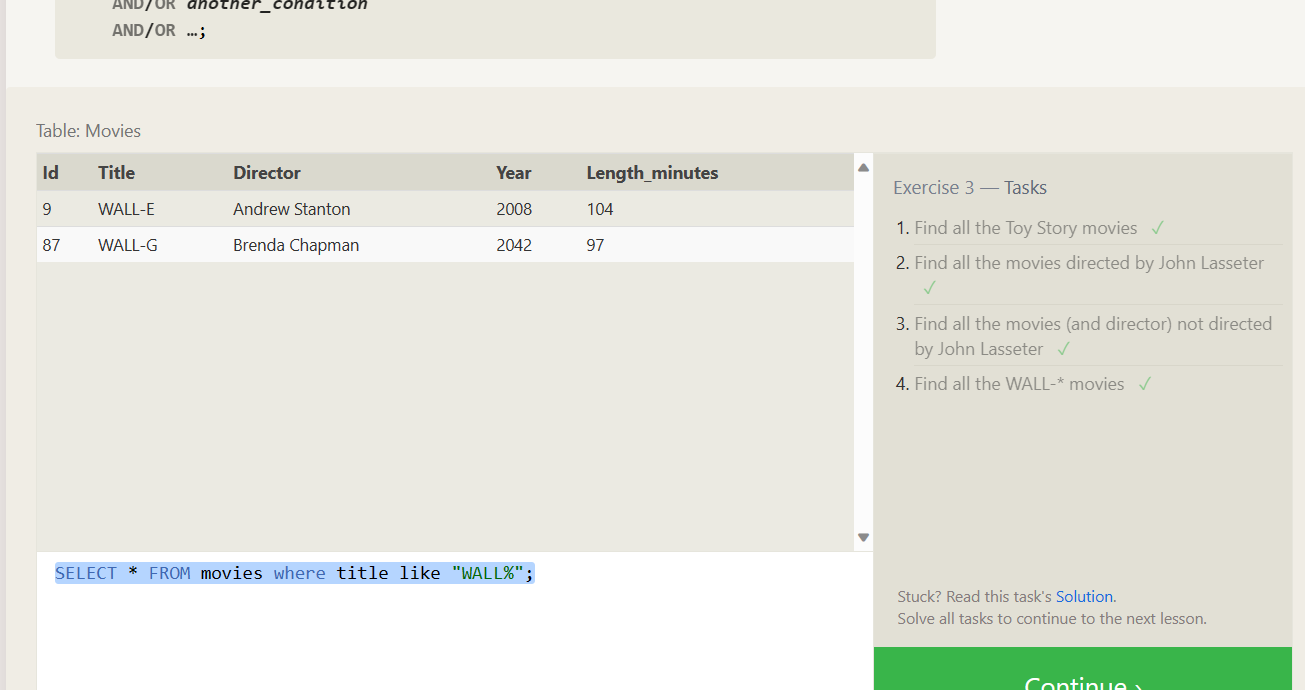
SELECT \* FROM movies where id=6;

SELECT Title FROM movies where year BETWEEN 2000 AND 2010;

SELECT Title FROM movies where year NOT BETWEEN 2000 AND 2010;

SELECT \* FROM movies where id IN(1,2,3,4,5)

Exercise 3



SELECT \* FROM movies where title like "Toy%";

SELECT \* FROM movies where director="John Lasseter";

SELECT title, director FROM movies where director!="John Lasseter";

SELECT \* FROM movies where title like "WALL%";

Exercise 4

A screenshot of a computer

Description automatically generated

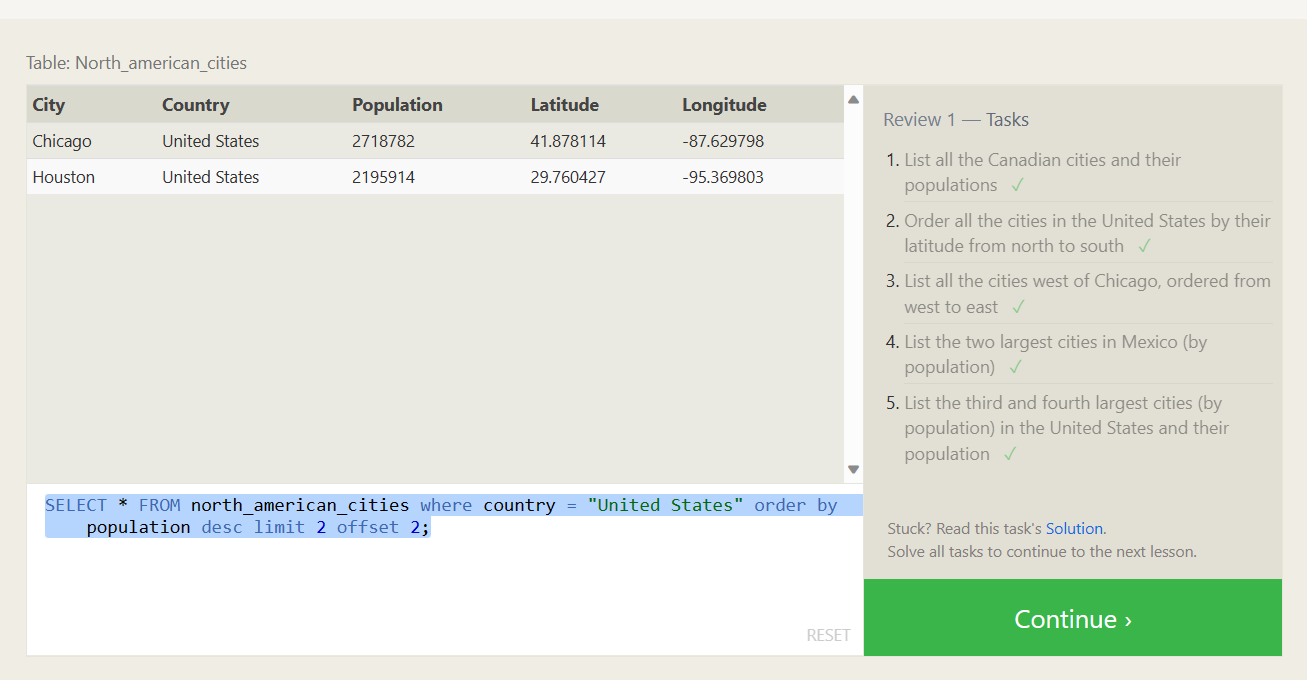
SELECT DISTINCT director FROM movies order by director ASC;

SELECT year, title FROM movies order by year desc limit 4;

SELECT year, title FROM movies order by title ASC limit 5;

SELECT year, title FROM movies order by title ASC limit 5 offset 5;

Exercise 5



SELECT city,country, population FROM north\_american\_cities where country like "canada";

SELECT \* FROM north\_american\_cities where country = "United States" order by latitude DESC;

SELECT \* FROM north\_american\_cities where longitude <-87.629798 order by longitude asc;

SELECT \* FROM north\_american\_cities where country = "Mexico" order by population desc limit 2;

SELECT \* FROM north\_american\_cities where country = "United States" order by population desc limit 2 offset 2;

Exercise 6

A screenshot of a computer

Description automatically generated

SELECT title, domestic\_sales, international\_sales

FROM movies

JOIN boxoffice

ON movies.id = boxoffice.movie\_id;

SELECT title, domestic\_sales, international\_sales

FROM movies

JOIN boxoffice

ON movies.id = boxoffice.movie\_id

WHERE international\_sales > domestic\_sales;

SELECT title, rating

FROM movies

JOIN boxoffice

ON movies.id = boxoffice.movie\_id

ORDER BY rating DESC;

Exercise 7

A screenshot of a computer

Description automatically generated

SELECT DISTINCT building FROM employees;

SELECT \* FROM buildings;

SELECT DISTINCT building\_name, role

FROM buildings

LEFT JOIN employees

ON building\_name = building;

Exercise 8

A screenshot of a computer

Description automatically generated

SELECT name, role FROM employees

WHERE building IS NULL;

SELECT DISTINCT building\_name

FROM buildings

LEFT JOIN employees

ON building\_name = building

WHERE role IS NULL;

Exercise 9

A screenshot of a computer

Description automatically generated

SELECT title, (domestic\_sales + international\_sales) / 1000000 AS gross\_sales\_millions

FROM movies

JOIN boxoffice

ON movies.id = boxoffice.movie\_id;

SELECT title, rating \* 10 AS rating\_percent

FROM movies

JOIN boxoffice

ON movies.id = boxoffice.movie\_id;

SELECT title, year

FROM movies

WHERE year % 2 = 0;

Exercise 10

A screenshot of a computer

Description automatically generated

SELECT MAX(years\_employed) as Max\_years\_employed

FROM employees;

SELECT role, AVG(years\_employed) as Average\_years\_employed

FROM employees

GROUP BY role;

SELECT building, SUM(years\_employed) as Total\_years\_employed

FROM employees

GROUP BY building;

Exercise 11

A screenshot of a computer

Description automatically generated

SELECT role, COUNT(\*) as Number\_of\_artists

FROM employees

WHERE role = "Artist";

SELECT role, COUNT(\*)

FROM employees

GROUP BY role;

SELECT role, SUM(years\_employed)

FROM employees

GROUP BY role

HAVING role = "Engineer";

Exercise 12

A screenshot of a computer

Description automatically generated

SELECT director, COUNT(id) as Num\_movies\_directed

FROM movies

GROUP BY director;

SELECT director, SUM(domestic\_sales + international\_sales) as Cumulative\_sales\_from\_all\_movies

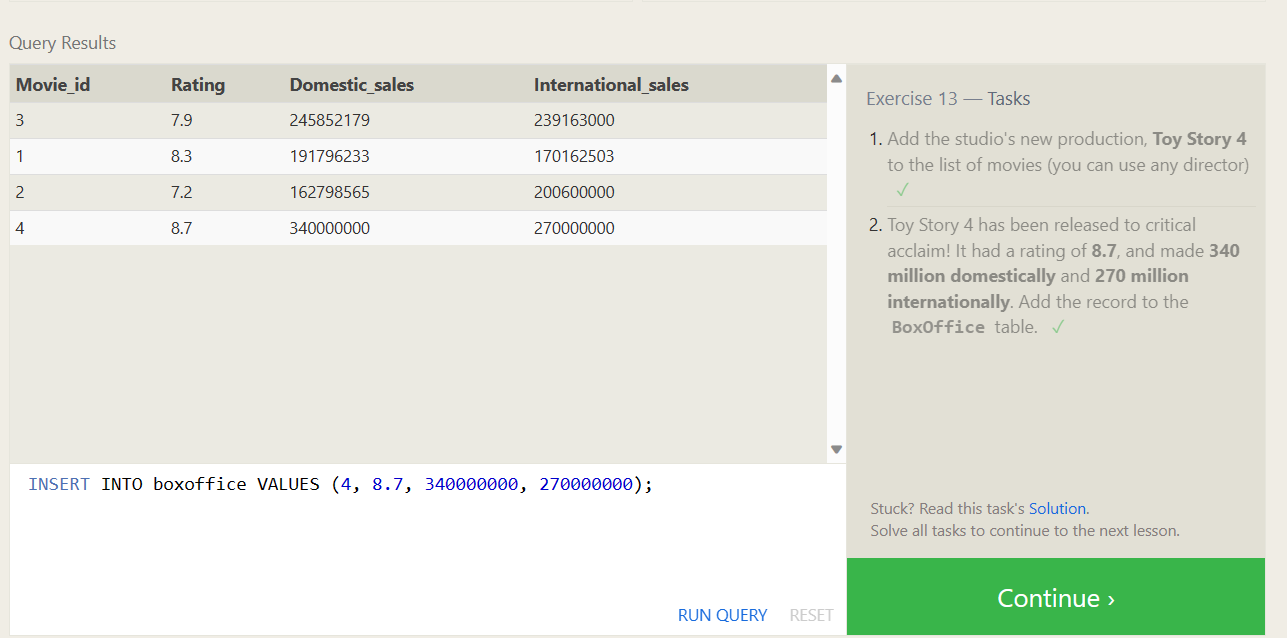
FROM movies

INNER JOIN boxoffice

ON movies.id = boxoffice.movie\_id

GROUP BY director;

Exercise 13



INSERT INTO movies VALUES (4, "Toy Story 4", "El Directore", 2015, 90);

INSERT INTO boxoffice VALUES (4, 8.7, 340000000, 270000000);

Exercise 14

A screenshot of a computer

Description automatically generated

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;

Exercise 15

A screenshot of a computer

Description automatically generated

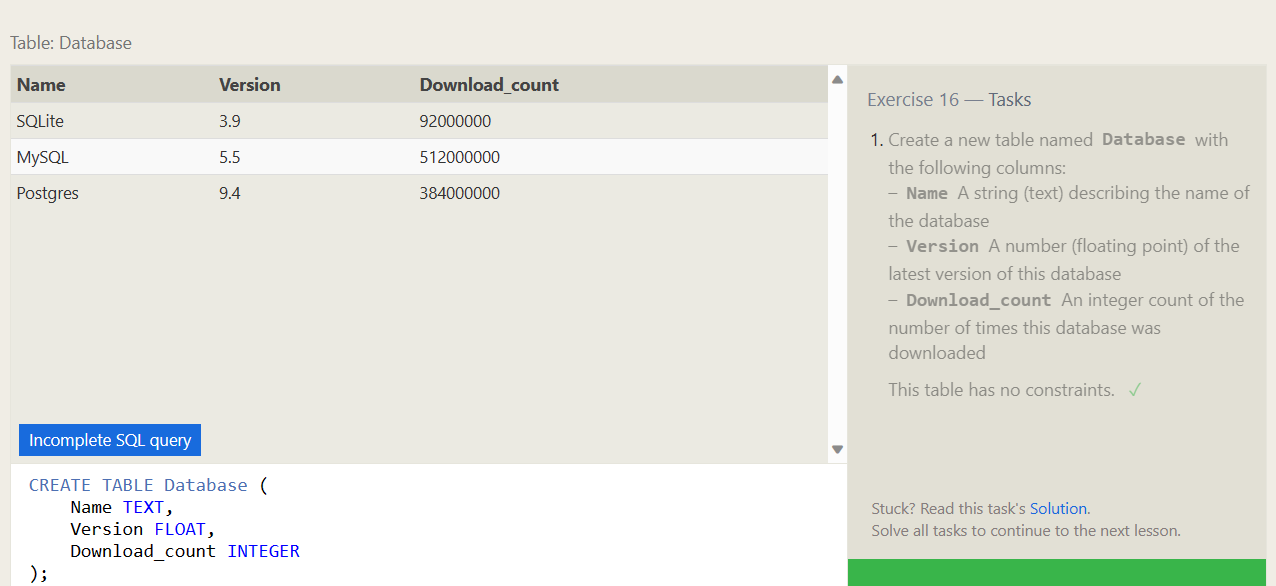
DELETE FROM movies

where year < 2005;

DELETE FROM movies

where director = "Andrew Stanton";

Exercise 16



CREATE TABLE Database (

Name TEXT,

Version FLOAT,

Download\_count INTEGER

);

Exercise 17

A screenshot of a computer

Description automatically generated

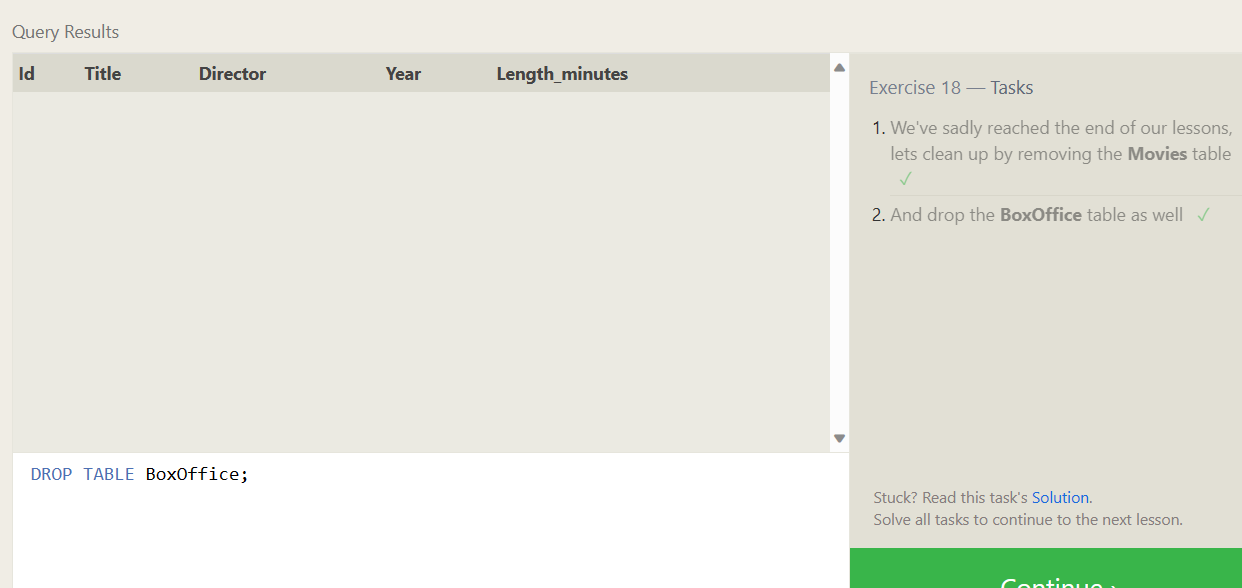
ALTER TABLE Movies

ADD COLUMN Aspect\_ratio FLOAT DEFAULT 2.39;

ALTER TABLE Movies

ADD COLUMN Language TEXT DEFAULT "English";

Exercise 18



DROP TABLE Movies;

DROP TABLE BoxOffice;

Certification

A screenshot of a computer

Description automatically generated