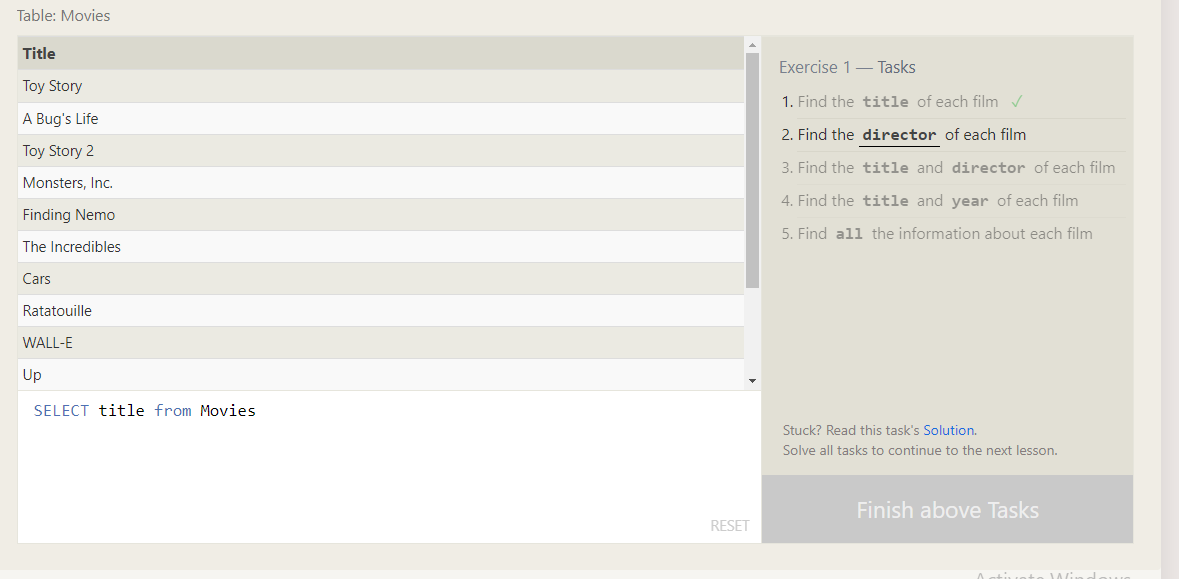
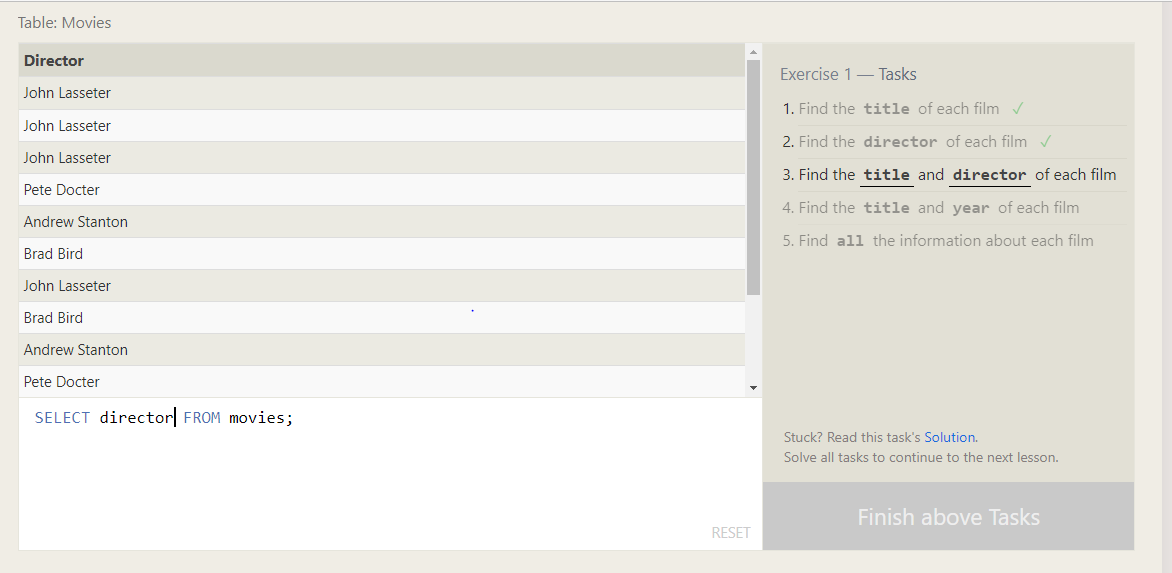
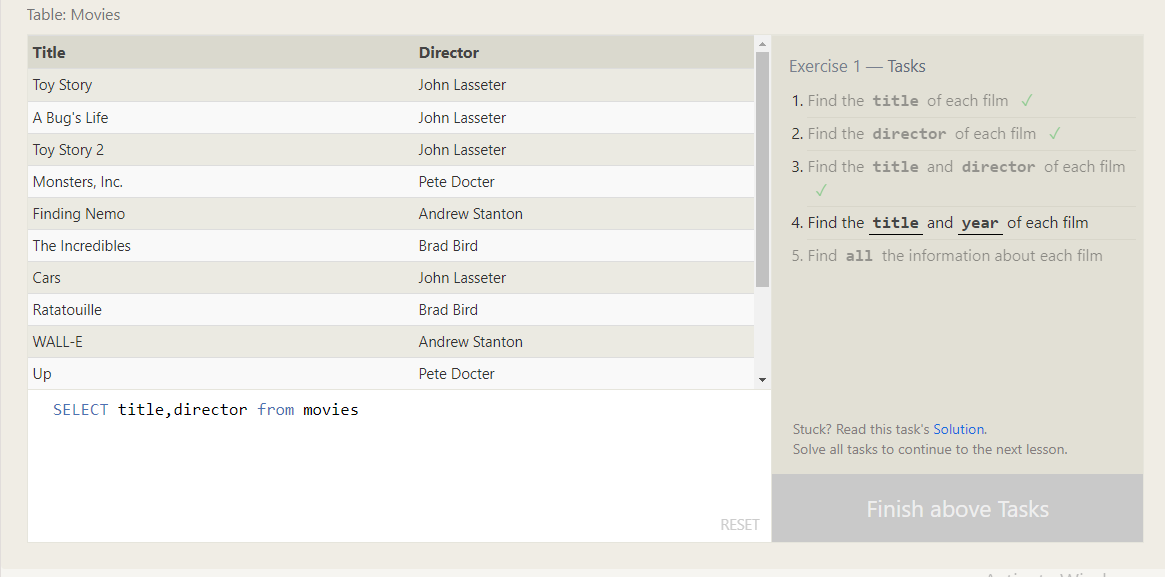
**SQL BOLT EXERCISES**

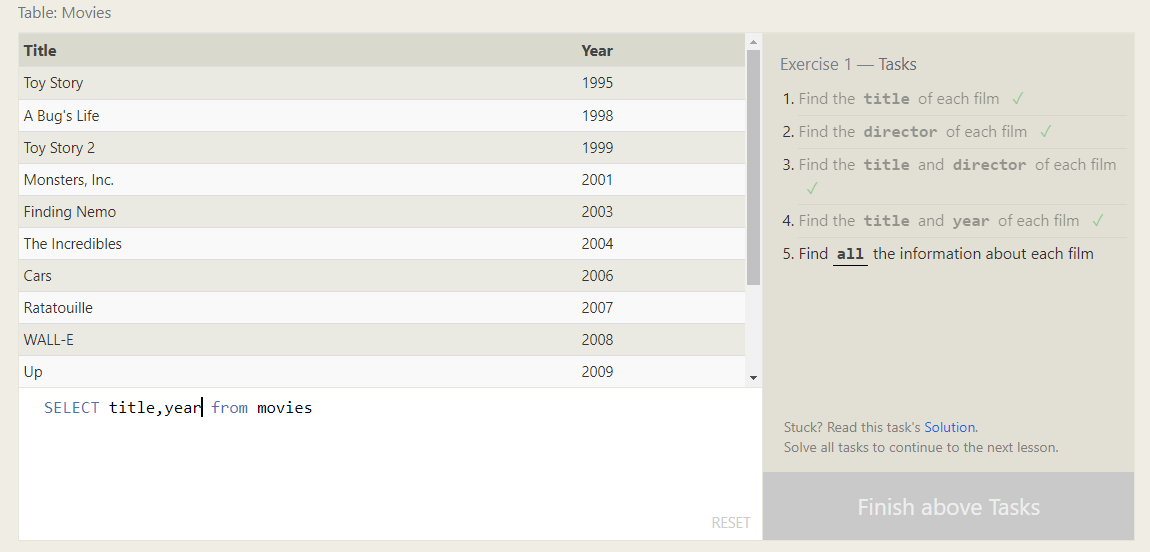
Exercise 1:

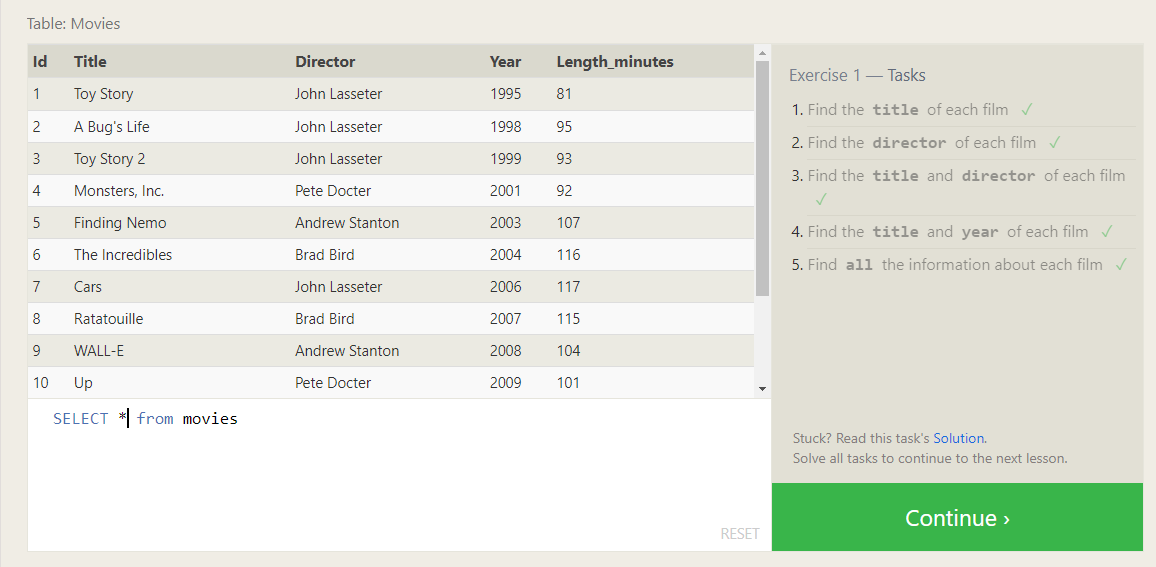
1) SELECT title from movies



2) SELECT director from movies

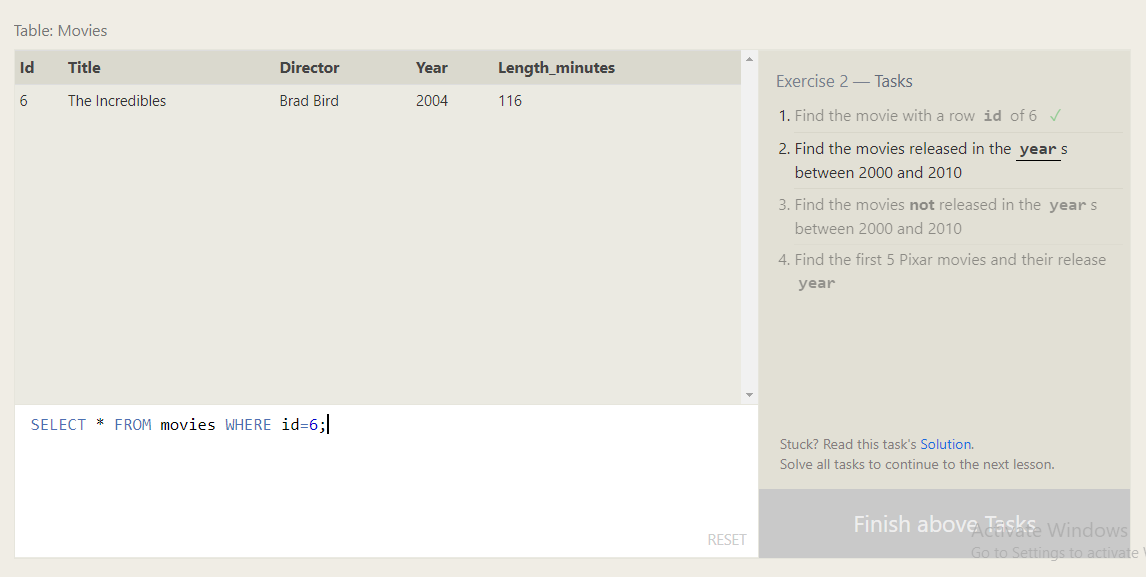
3) SELECT title,director from movies

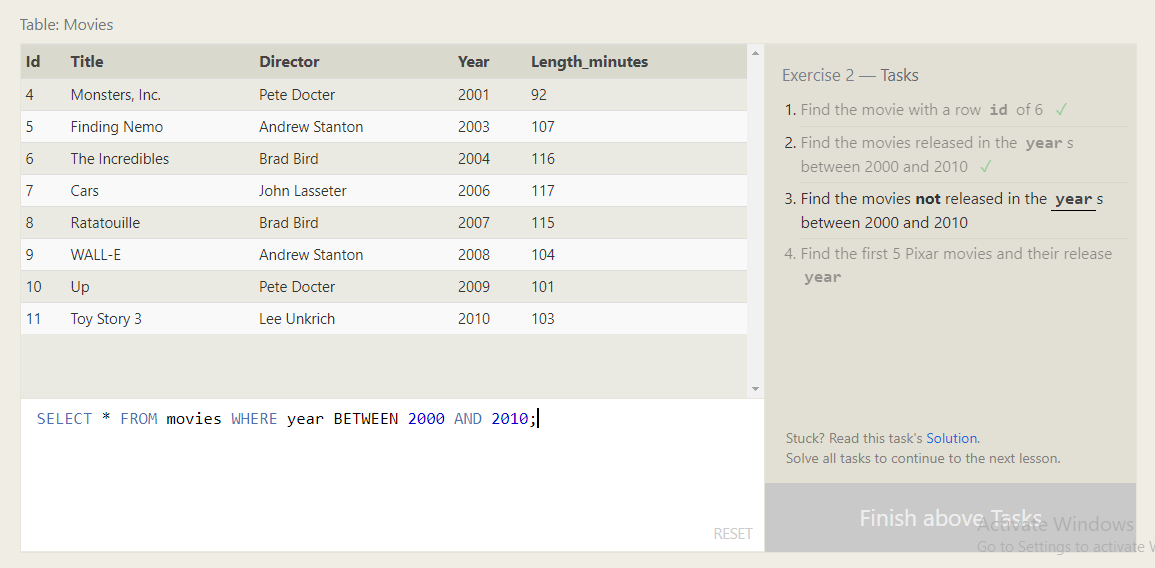
4) SELECT title,year from movies

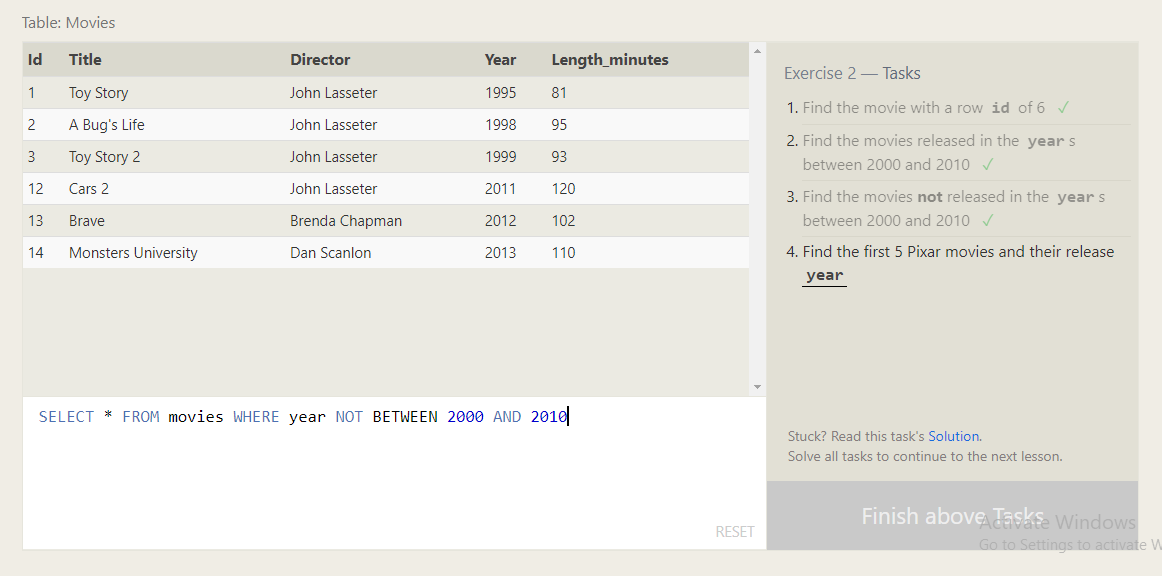
5) SELECT \* from movies

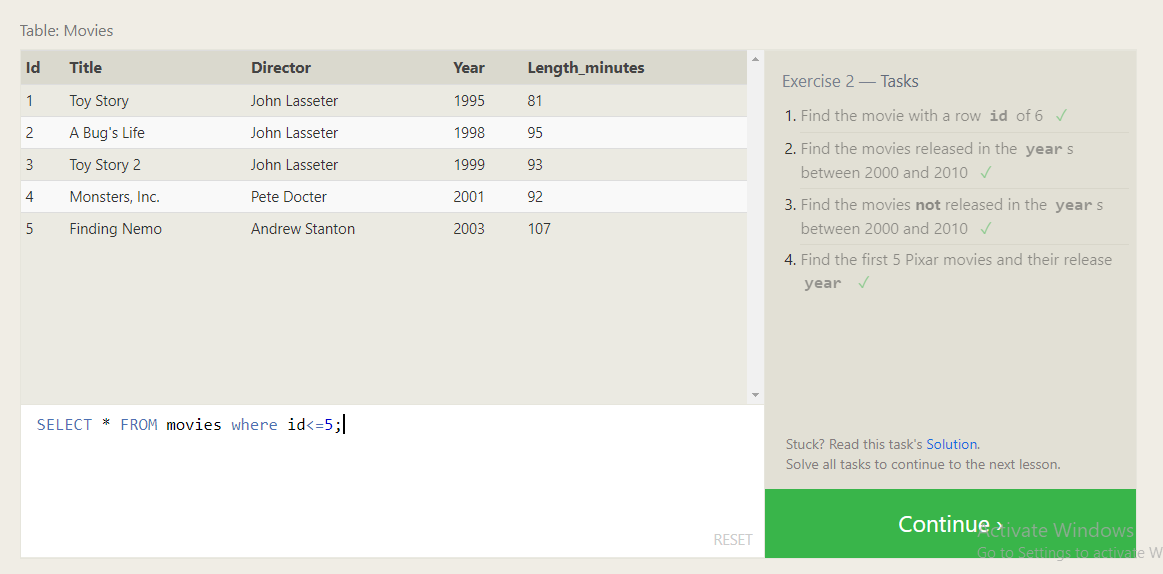
Exercise 2:

1. SELECT \* FROM movies WHERE id=6;



2) SELECT \* FROM movies WHERE year BETWEEN 2000 AND 2010;

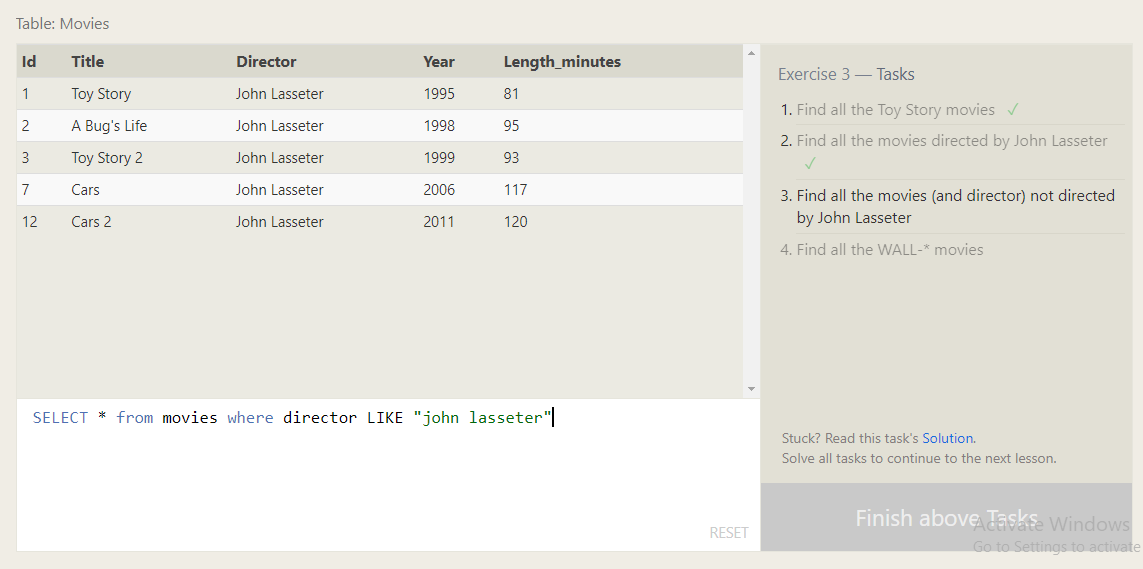
3) SELECT \* FROM movies WHERE year NOT BETWEEN 2000 AND 2010

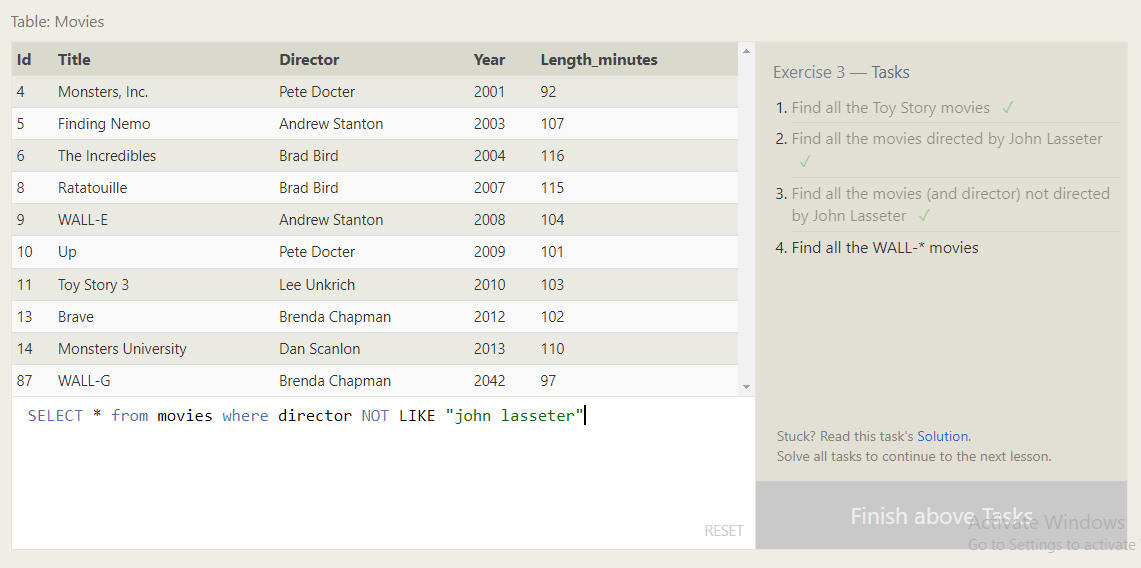
4) SELECT \* FROM movies where id<=5;

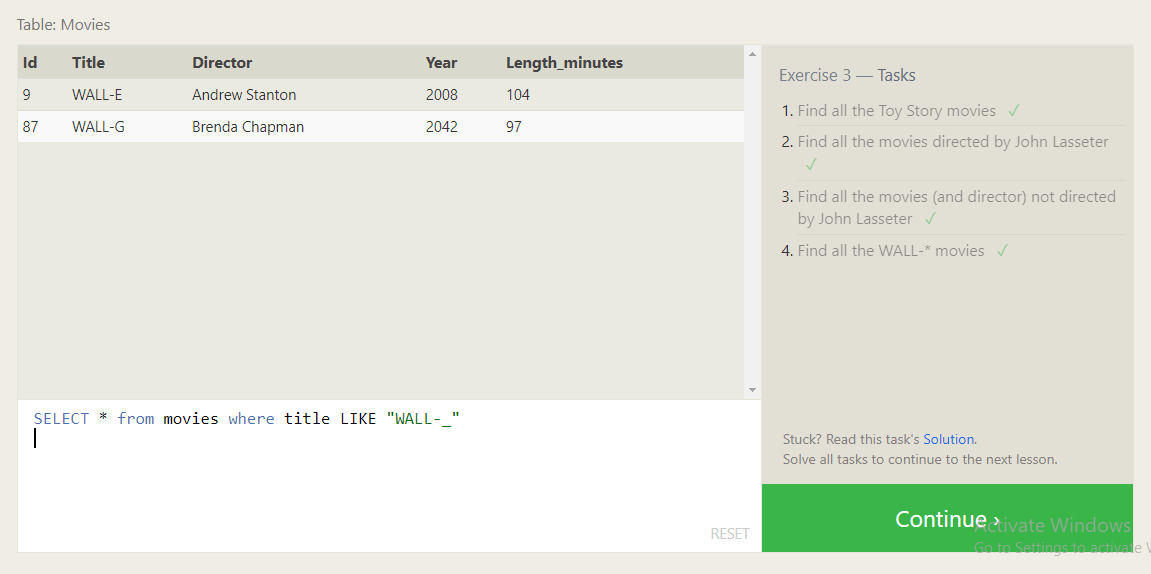
Exercise 3:

1. SELECT \* FROM movies WHERE title LIKE "toy story%";



2) SELECT \* from movies where director LIKE "john lasseter"

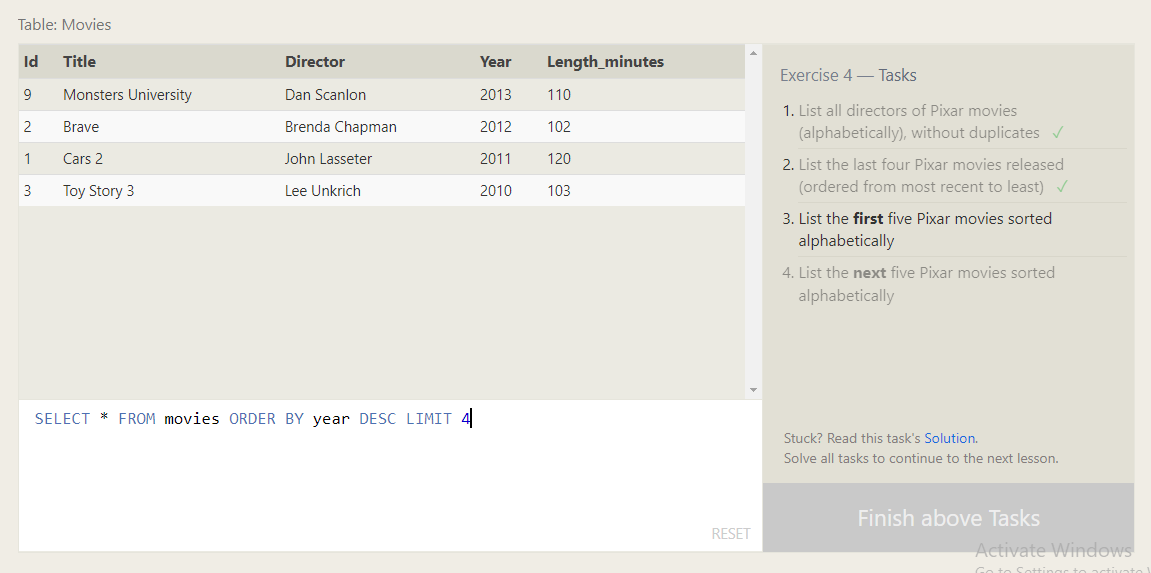
3) SELECT \* from movies where director NOT LIKE "john lasseter"

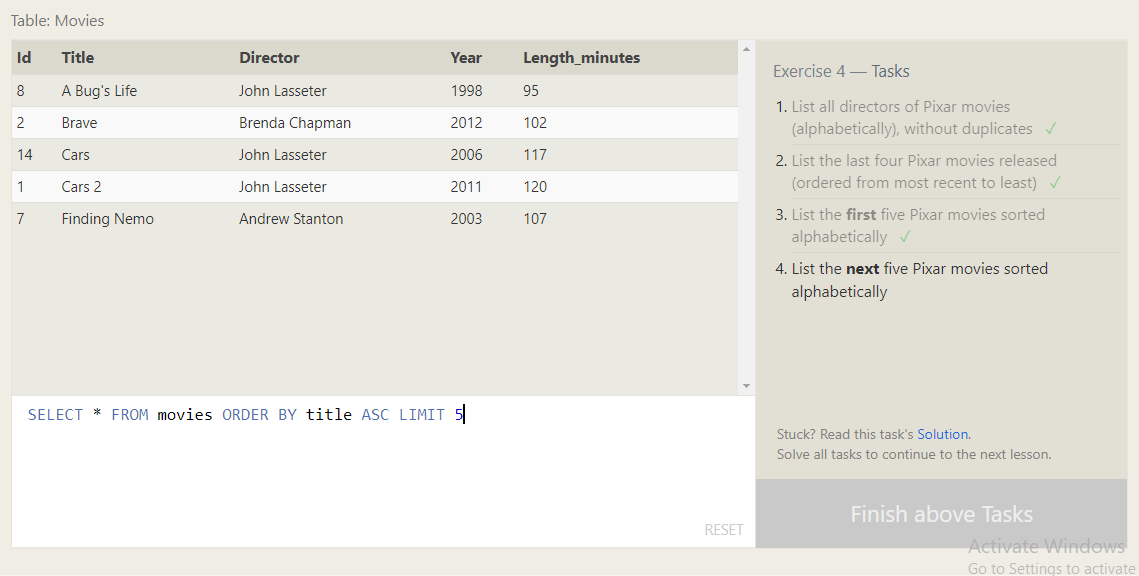
4) SELECT \* from movies where title LIKE "WALL-\_"

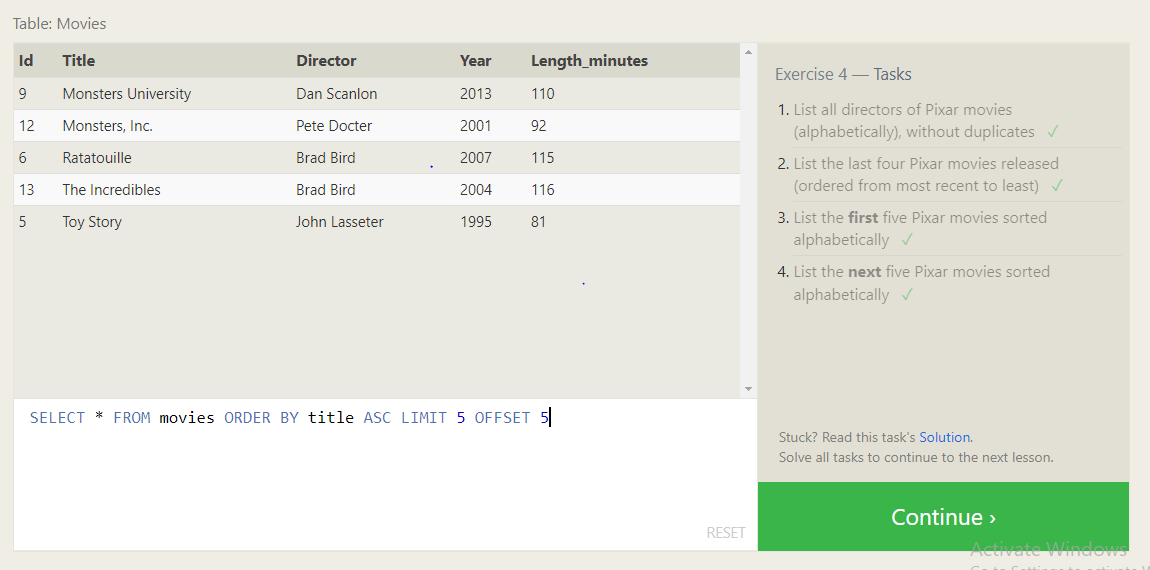
Exercise 4:

1. SELECT distinct director FROM movies ORDER BY director ASC



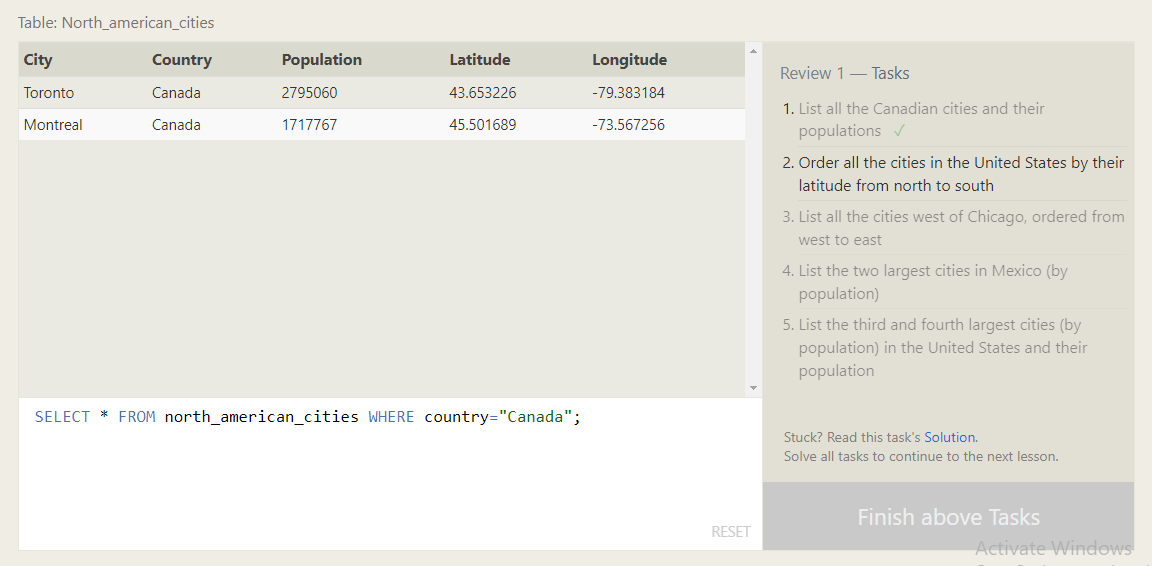
2) SELECT \* FROM movies ORDER BY year DESC LIMIT 4

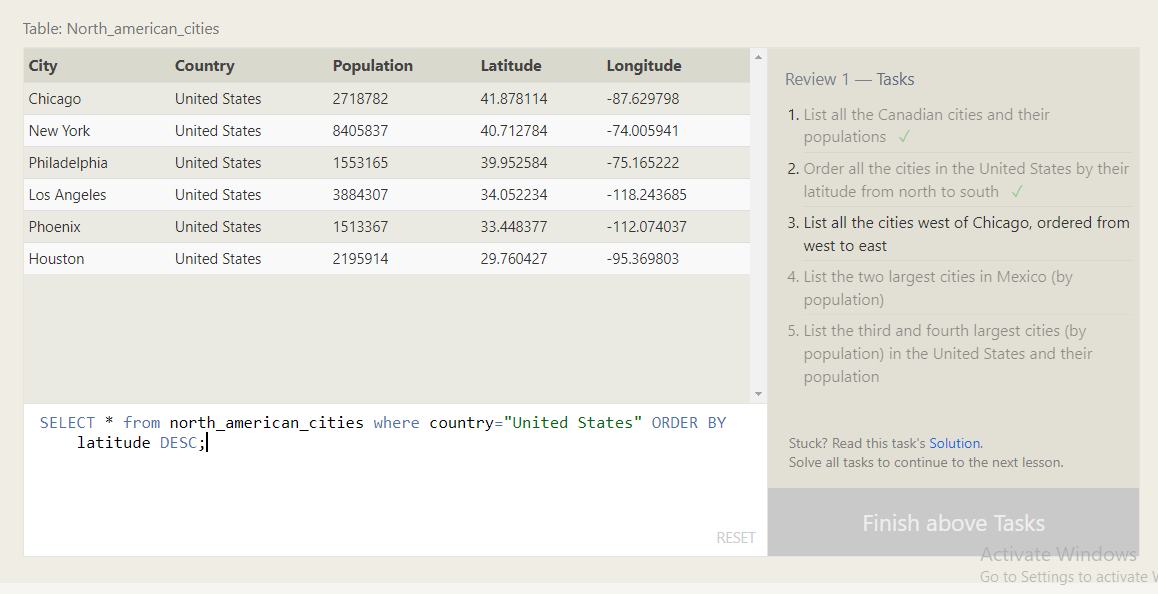
3) SELECT \* FROM movies ORDER BY title ASC LIMIT 5

4) SELECT \* FROM movies ORDER BY title ASC LIMIT 5 OFFSET 5

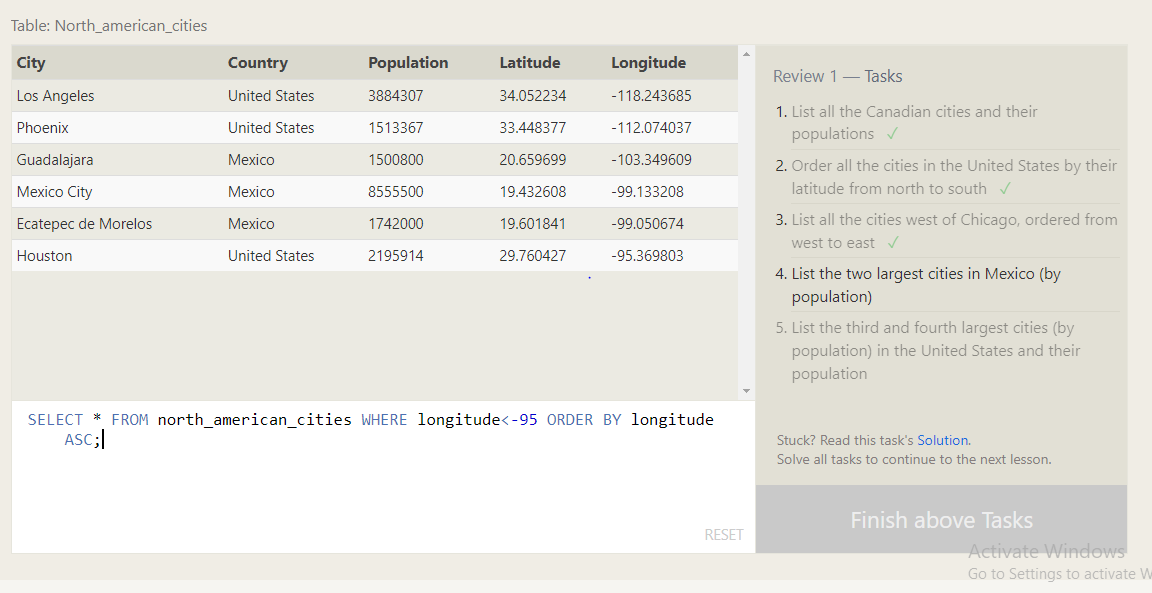
Exercise 5:

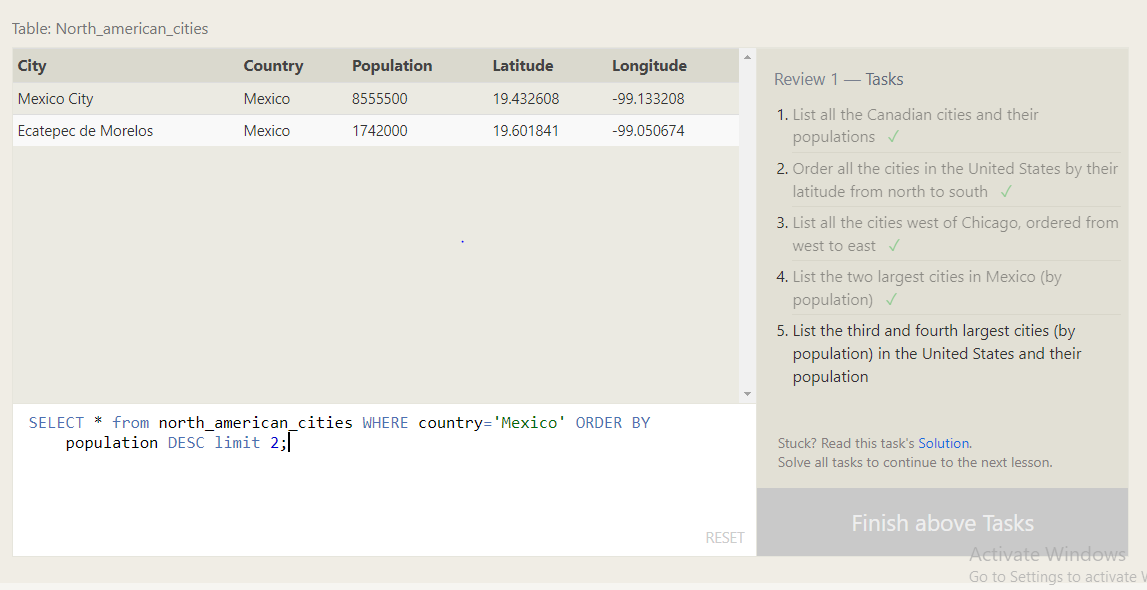
1. SELECT \* FROM north\_american\_cities WHERE country="Canada";

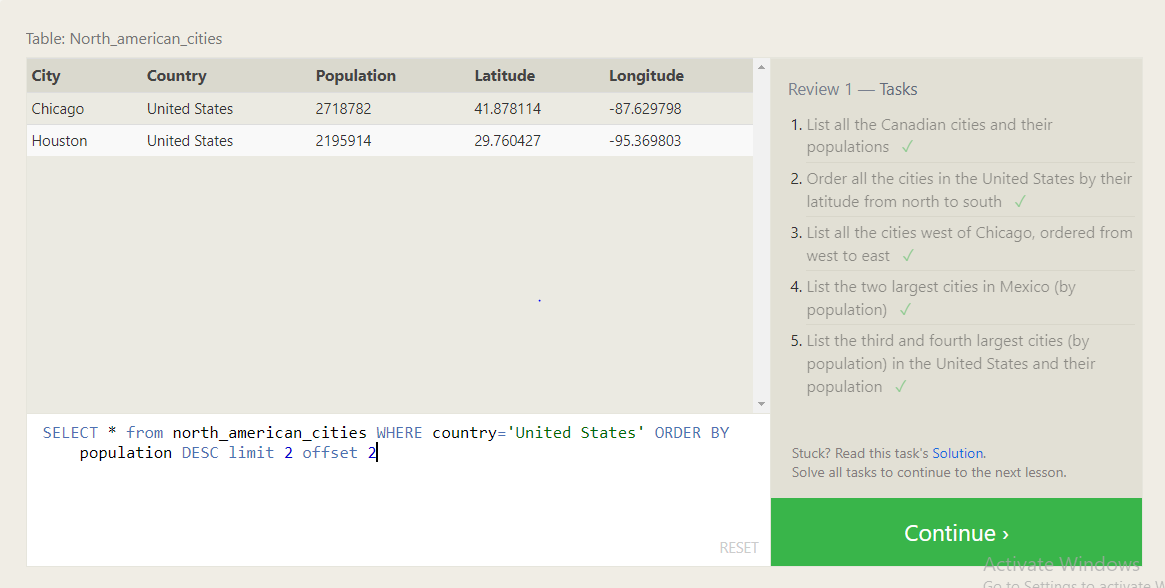


2) SELECT \* from north\_american\_cities where country="United States" ORDER BY latitude DESC;

3) SELECT \* FROM north\_american\_cities WHERE longitude<-95 ORDER BY longitude ASC;

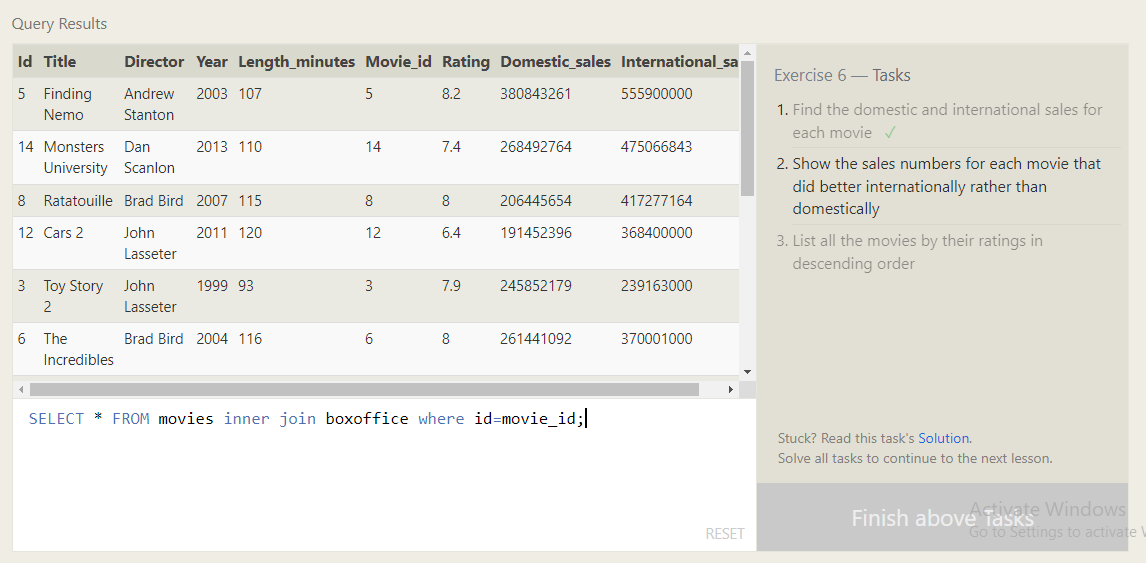


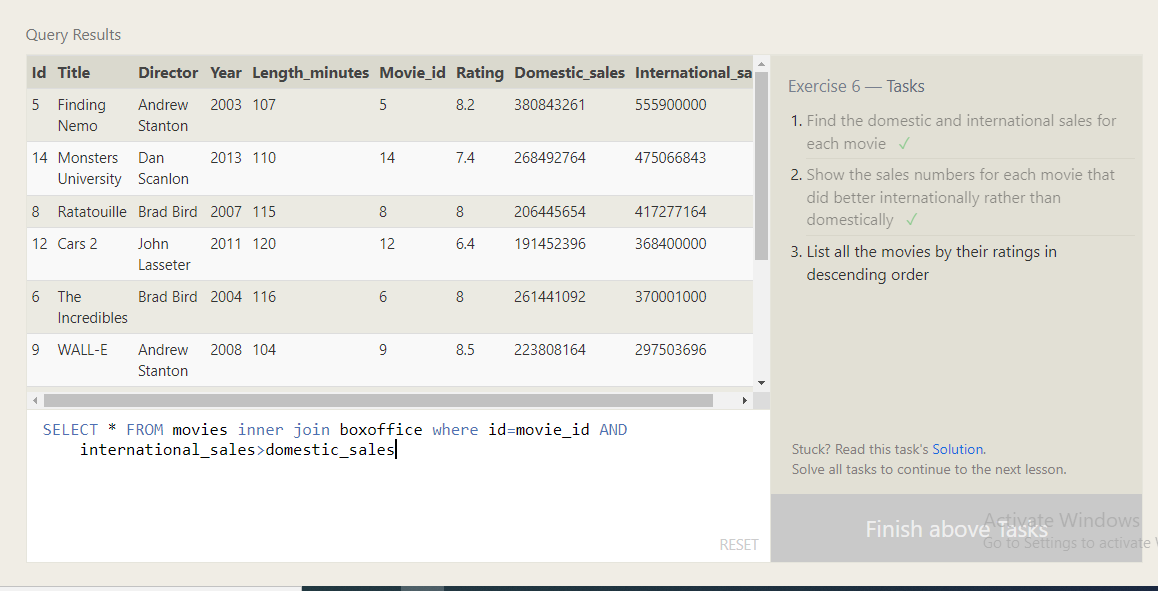
4) SELECT \* from north\_american\_cities WHERE country='Mexico' ORDER BY population DESC limit 2;

5) SELECT \* from north\_american\_cities WHERE country='United States' ORDER BY population DESC limit 2 offset 2

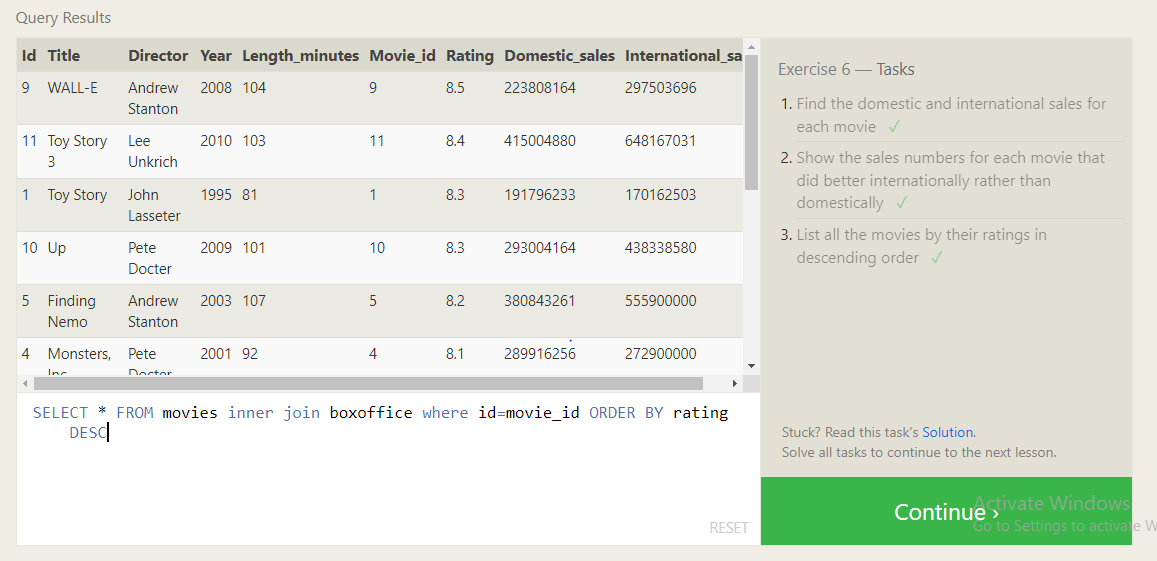
Exercise 6:

1) SELECT \* FROM movies inner join boxoffice where id=movie\_id;

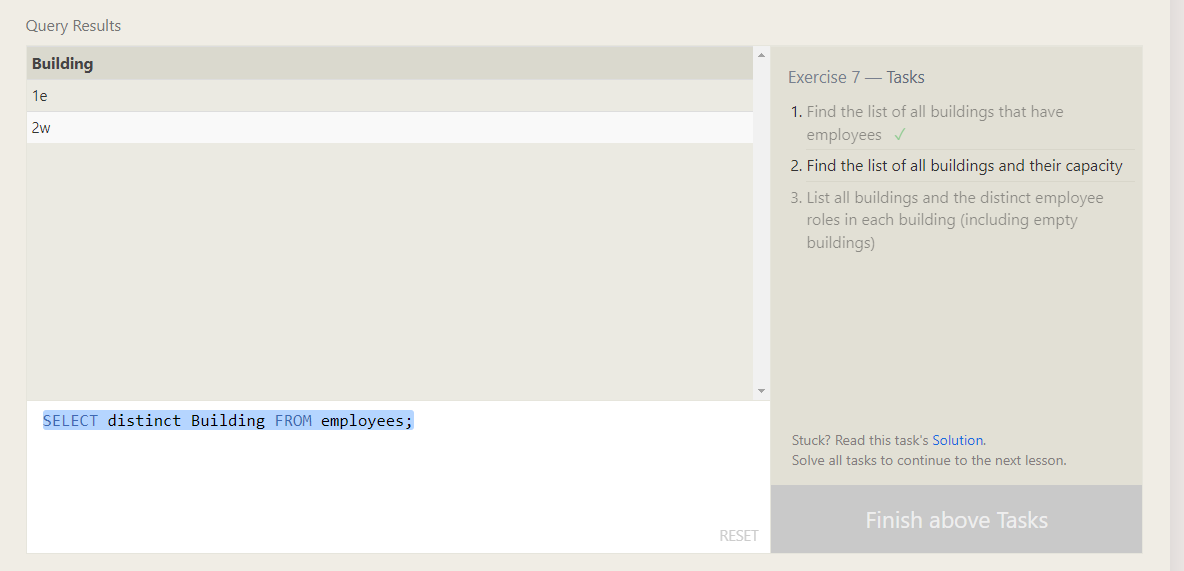


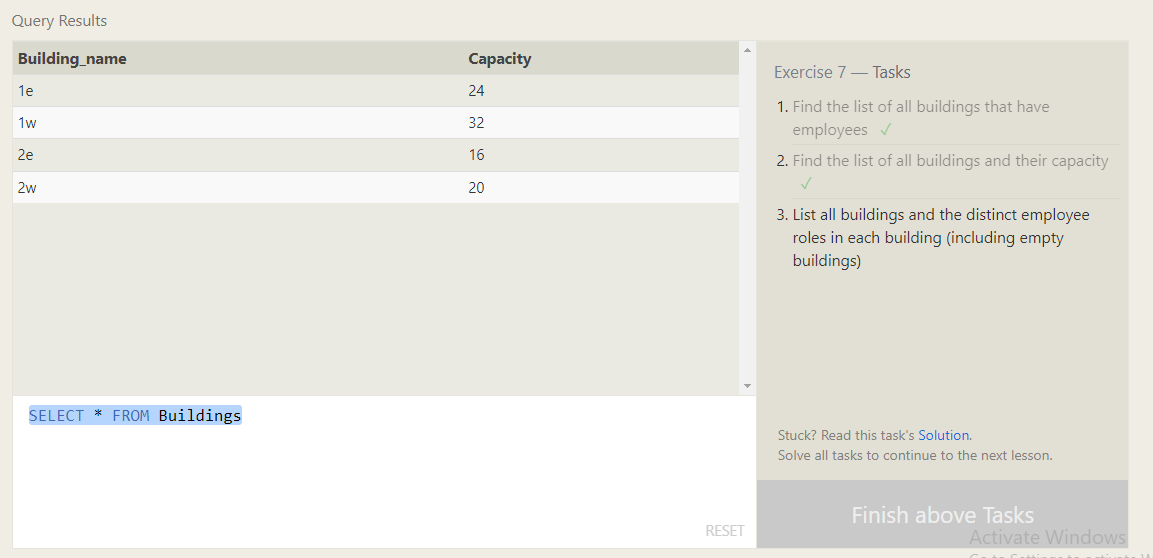
2) SELECT \* FROM movies inner join boxoffice where id=movie\_id AND international\_sales>domestic\_sales

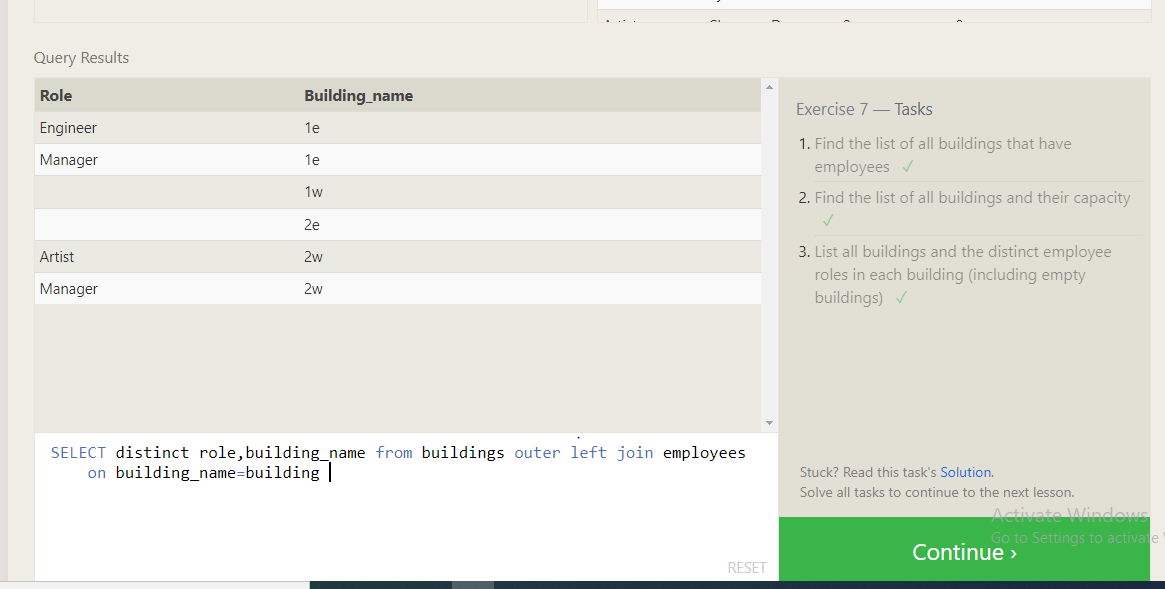
3) SELECT \* FROM movies inner join boxoffice where id=movie\_id ORDER BY rating DESC



Exercise 7:

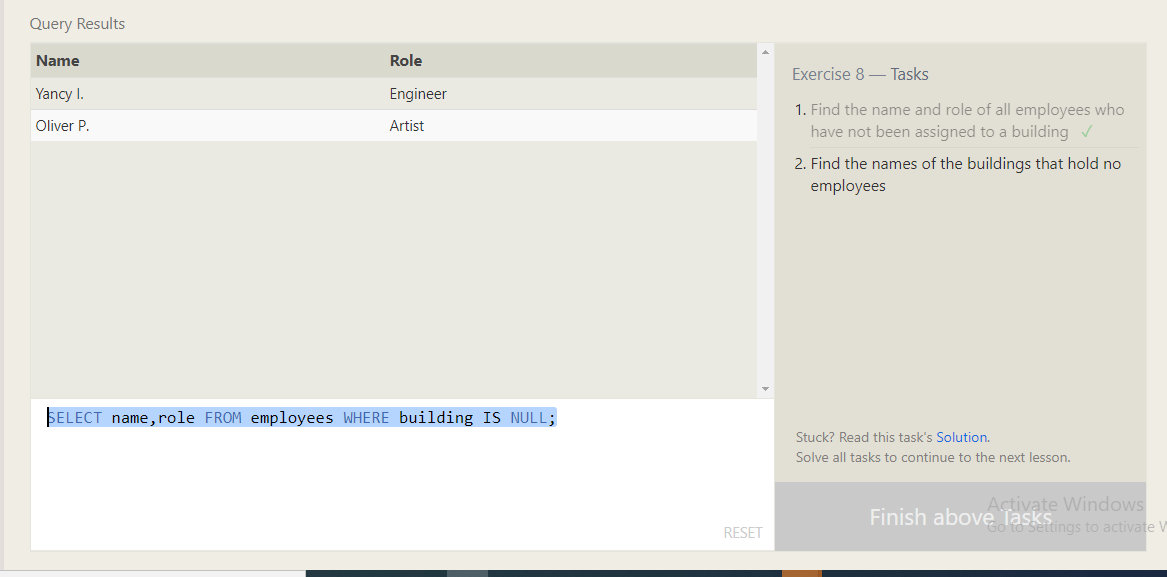
1) SELECT distinct Building FROM employees;

2) SELECT \* FROM Buildings;

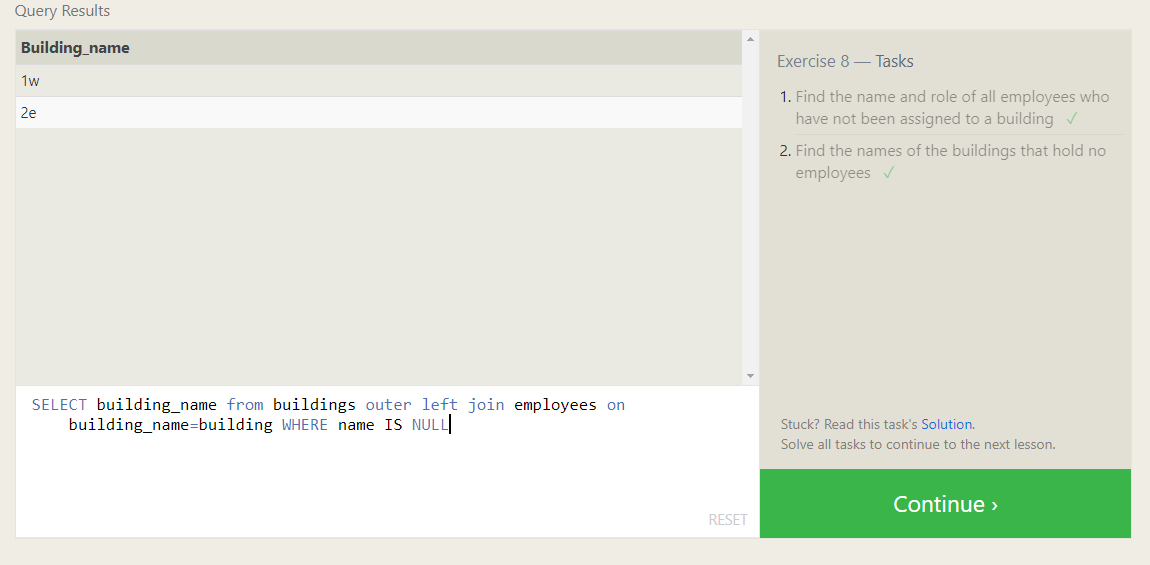
3) SELECT distinct role,building\_name from buildings outer left join employees on building\_name=building 

Exercise 8:

1) SELECT name,role FROM employees WHERE building IS NULL;



2) SELECT building\_name from buildings outer left join employees on building\_name=building WHERE name IS NULL

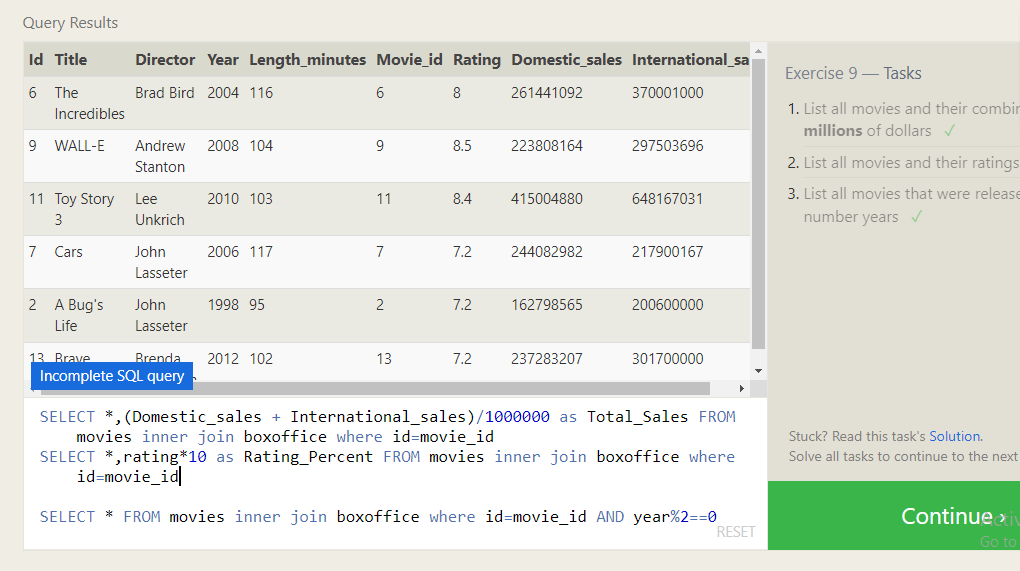


Exercise 9:

SELECT \*,(Domestic\_sales + International\_sales)/1000000 as Total\_Sales FROM movies inner join boxoffice where id=movie\_id

SELECT \*,rating\*10 as Rating\_Percent FROM movies inner join boxoffice where id=movie\_id

SELECT \* FROM movies inner join boxoffice where id=movie\_id AND year%2==0

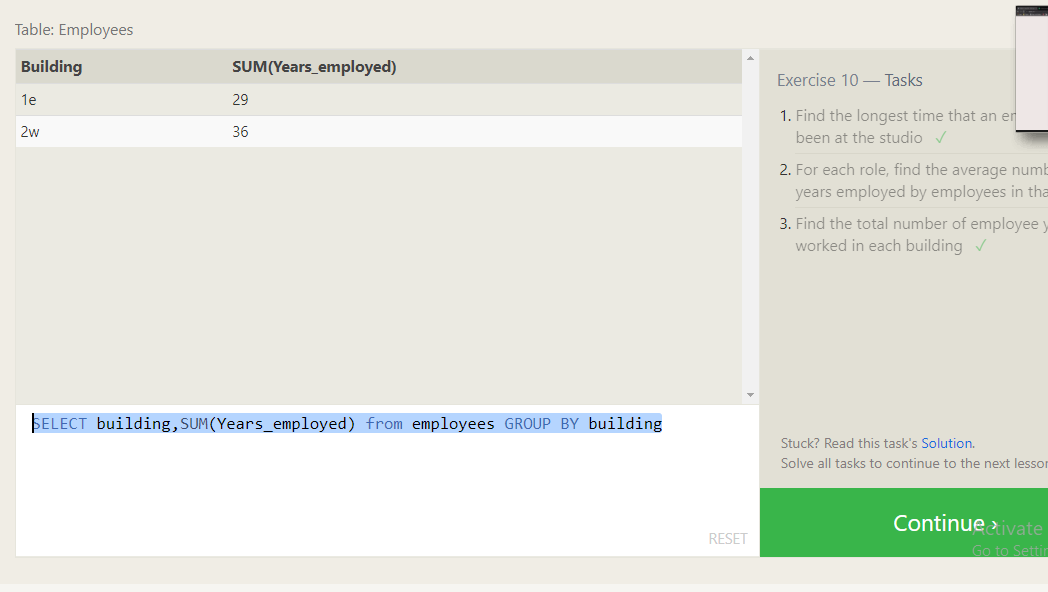


Exercise 10:

1) SELECT MAX(Years\_Employed) FROM employees;



2) SELECT distinct role,AVG(Years\_employed) AS AVG\_Year FROM employees GROUP BY role;

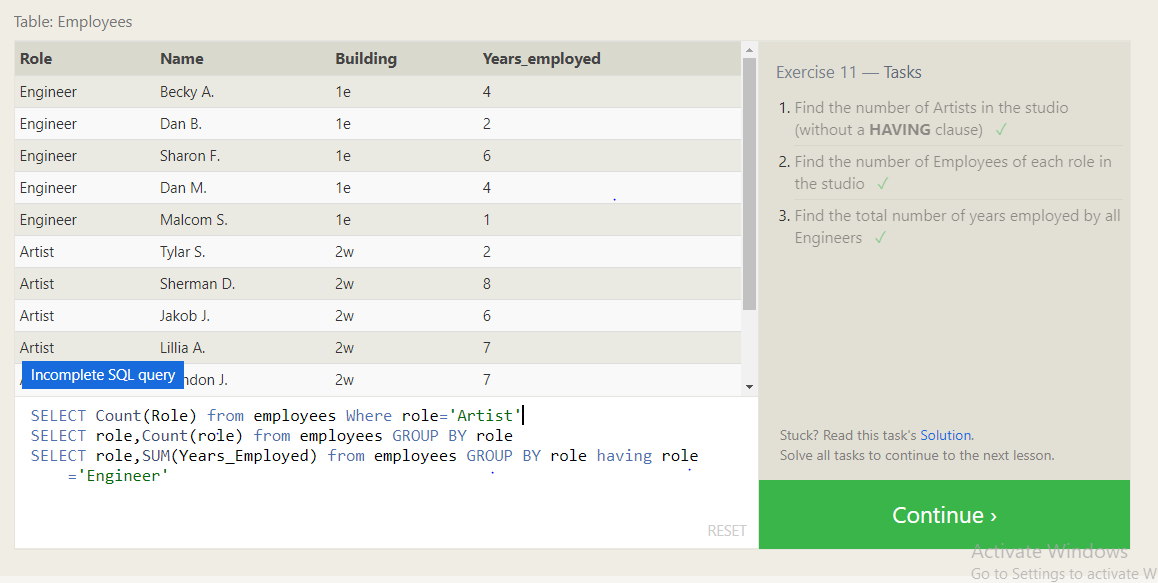
3) SELECT building,SUM(Years\_employed) from employees GROUP BY building

Exercise 11:

SELECT Count(Role) from employees Where role='Artist'

SELECT role,Count(role) from employees GROUP BY role

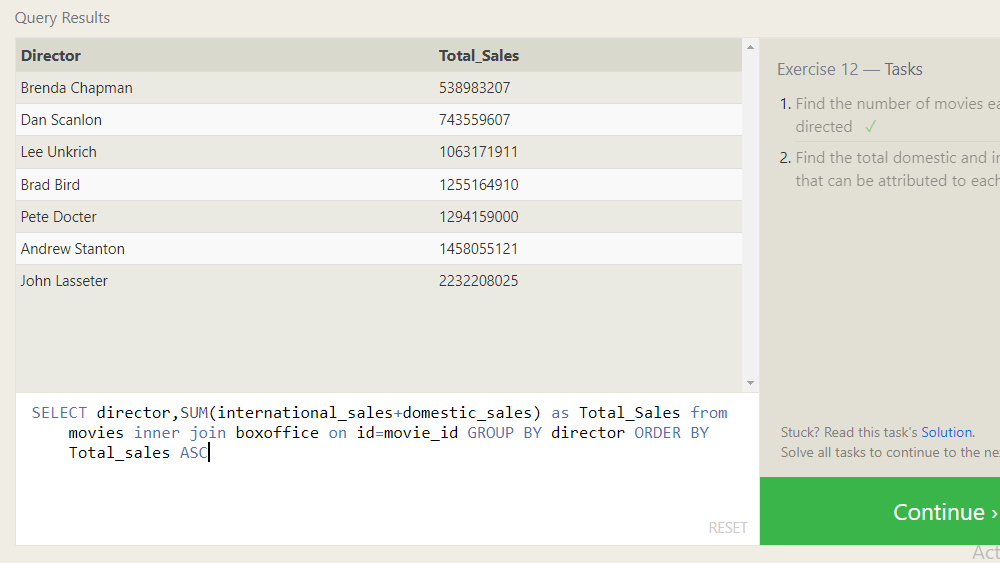
SELECT role,SUM(Years\_Employed) from employees GROUP BY role having role='Engineer'



EXERCISE 12:

SELECT director,COUNT(title) as NO\_OF\_MOVIES from movies GROUP BY director

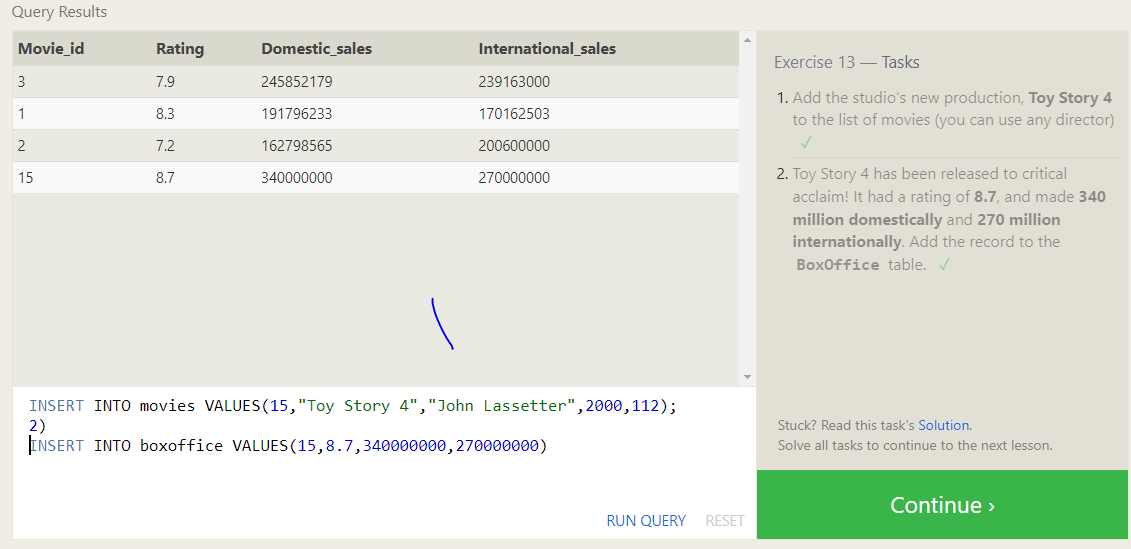
SELECT director,SUM(international\_sales+domestic\_sales) as Total\_Sales from movies inner join boxoffice on id=movie\_id GROUP BY director



EXERCISE 13:

INSERT INTO movies VALUES(15,"Toy Story 4","John Lassetter",2000,112);

INSERT INTO boxoffice VALUES(15,8.7,340000000,270000000)

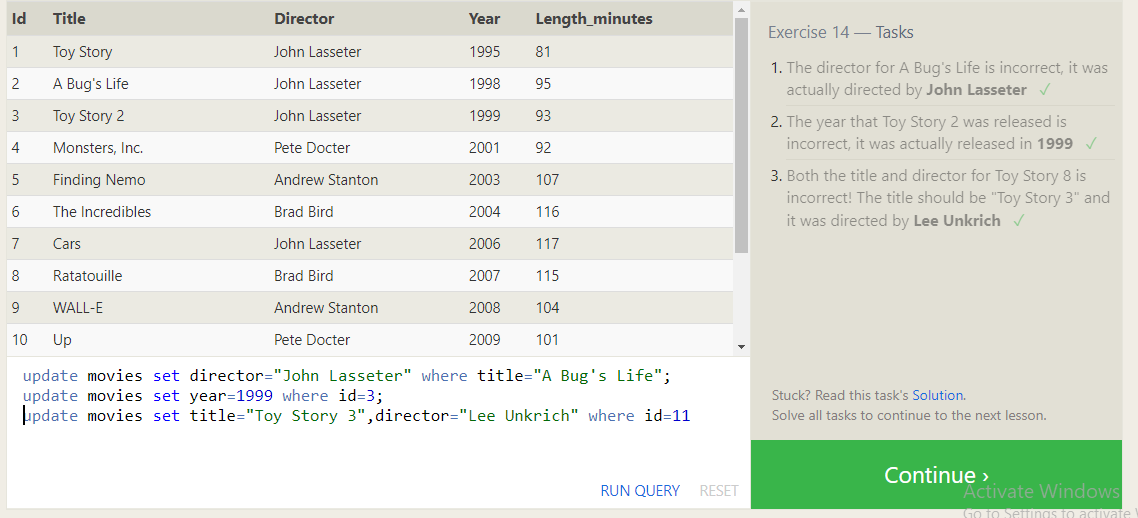


Exercise 14:

update movies set director="John Lasseter" where title="A Bug's Life";

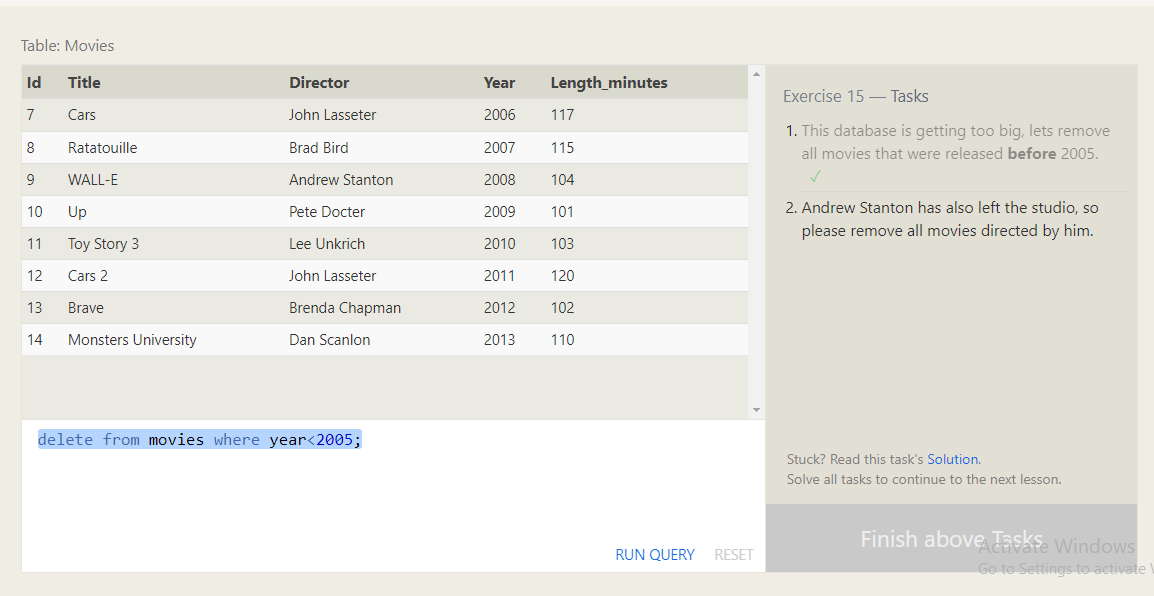
update movies set year=1999 where id=3;

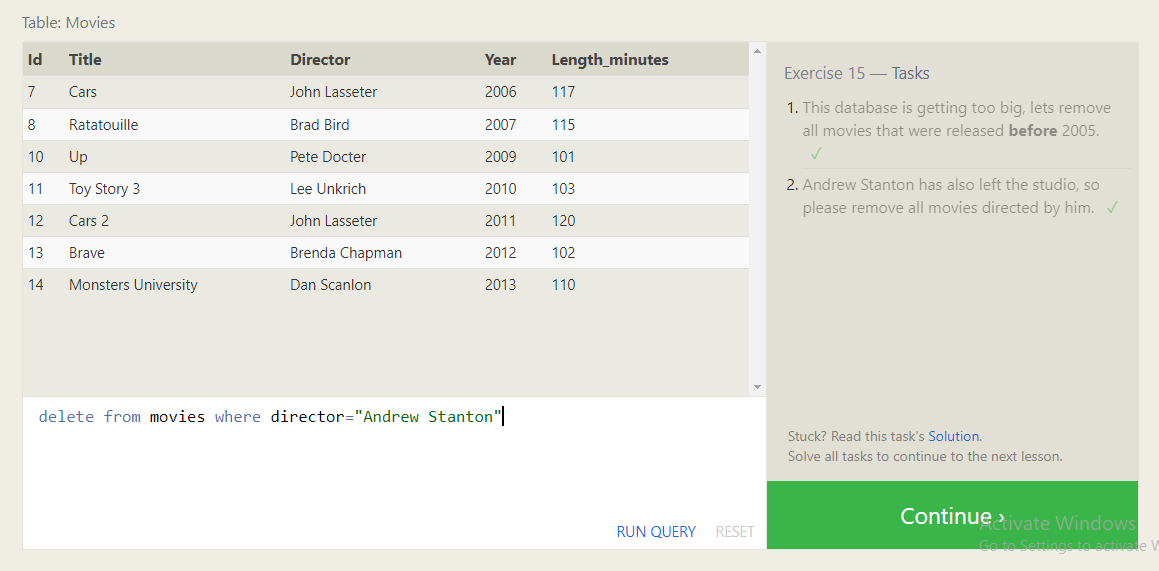
update movies set title="Toy Story 3",director="Lee Unkrich" where id=11



Exercise 15:

1. delete from movies where year<2005;



2) delete from movies where director="Andrew Stanton"

Exercise 16:

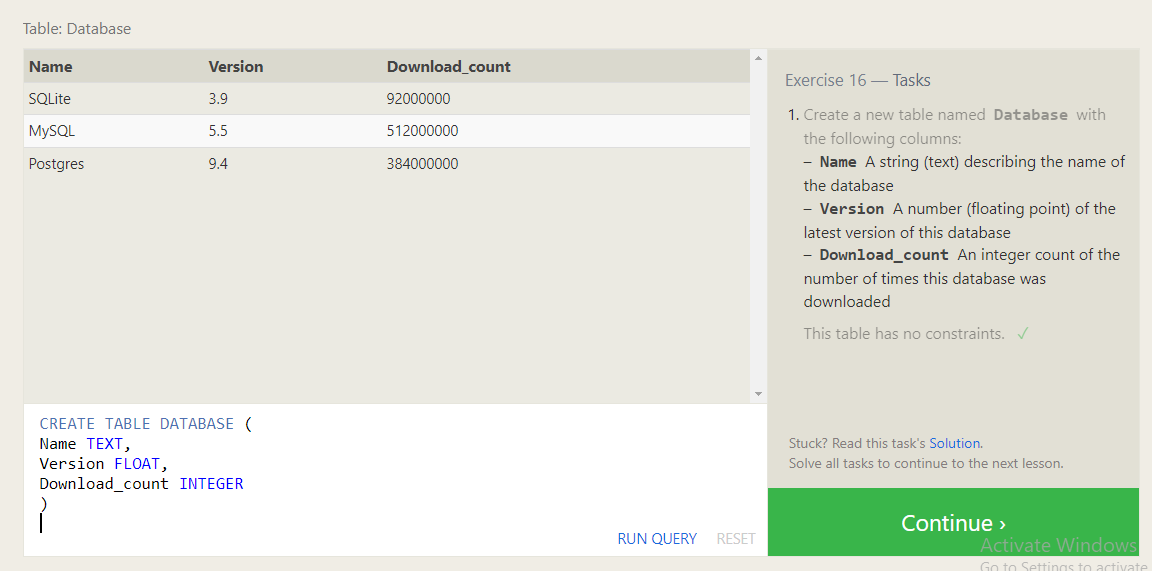
CREATE TABLE DATABASE (

Name TEXT,

Version FLOAT,

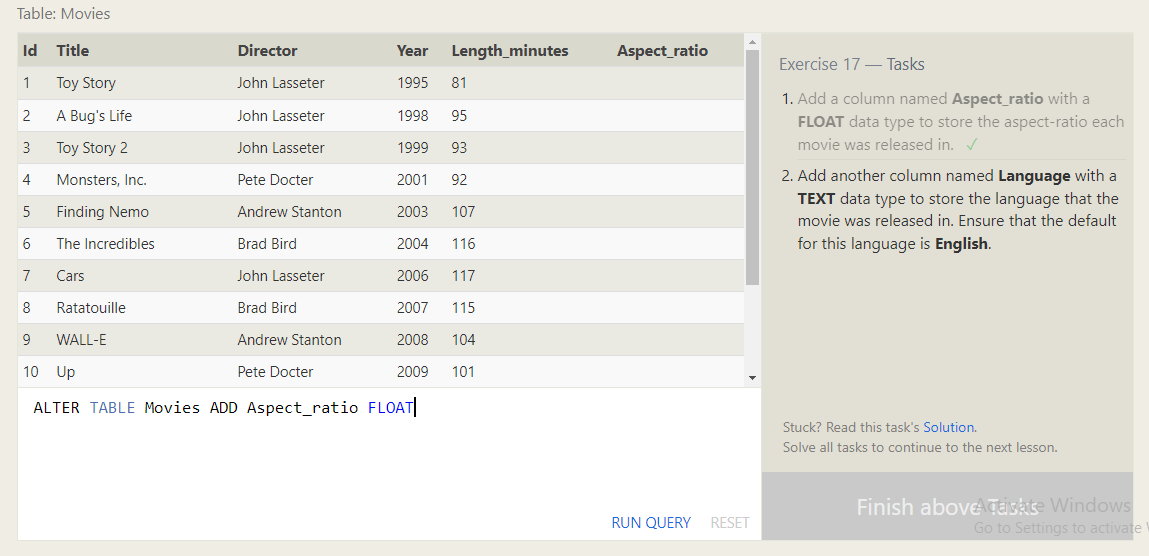
Download\_count INTEGER

)

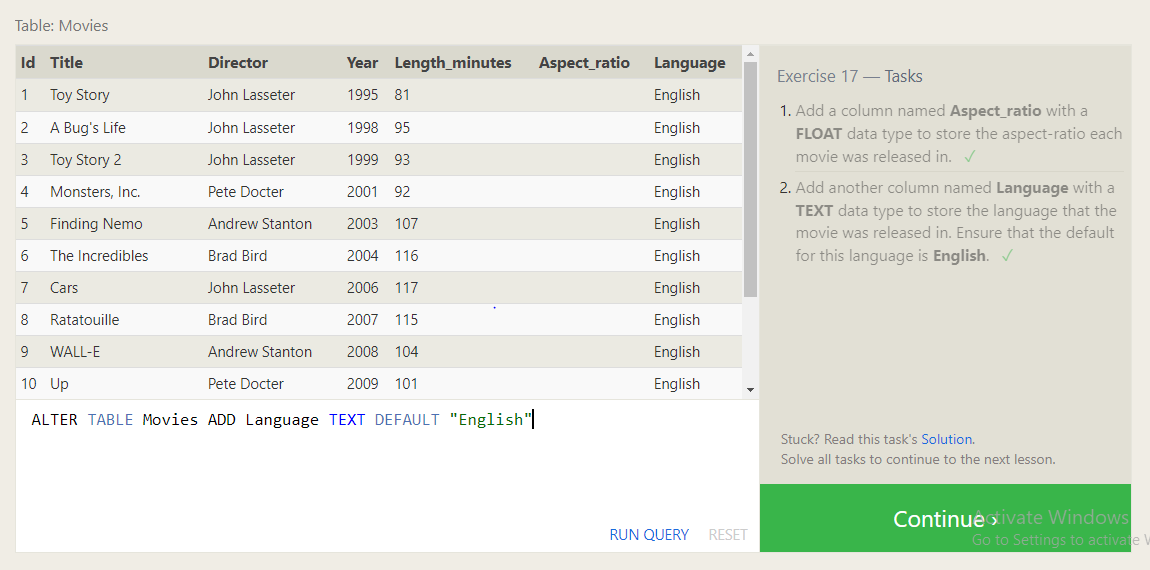


Exercise 17:

1. ALTER TABLE Movies ADD Aspect\_ratio FLOAT

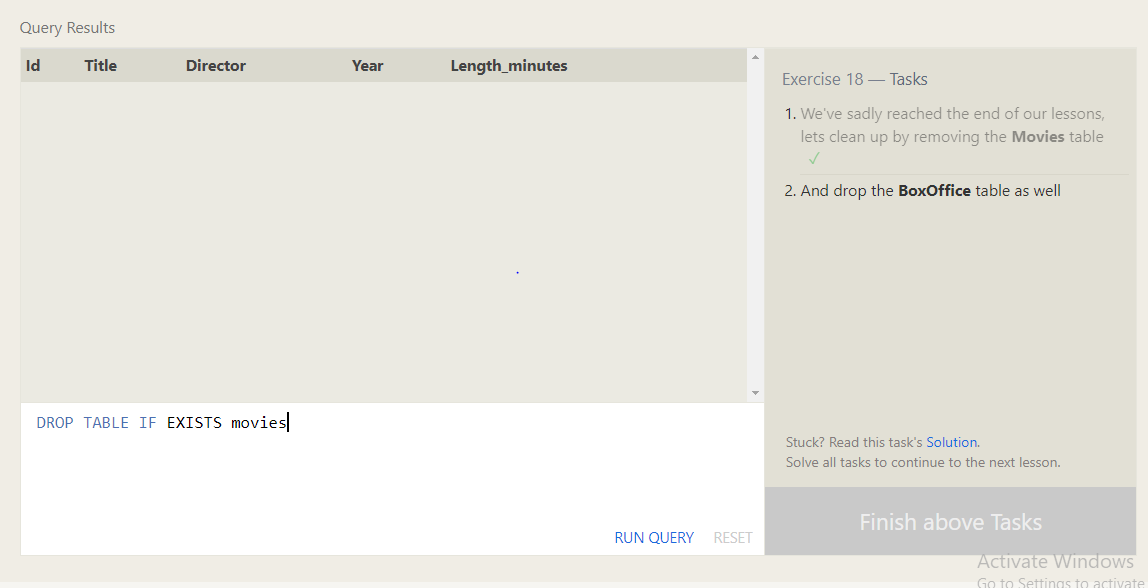


1. ALTER TABLE Movies ADD Language TEXT DEFAULT "English"



Exercise 18:

1. DROP TABLE IF EXISTS movies



2) DROP TABLE IF EXISTS BoxOffice