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 ›

Table: Movies

Title	Year
Toy Story	1995
A Bug's Life	1998
Toy Story 2	1999
Monsters, Inc.	2001
Finding Nemo	2003

```
SELECT title, year FROM movies
WHERE year >= 1995 AND year <= 2003;
```

RESET

Exercise 2 — Tasks

1. Find the movie with a row `id` of 6 ✓
2. Find the movies released in the `year`s between 2000 and 2010 ✓
3. Find the movies **not** released in the `year`s between 2000 and 2010 ✓
4. 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.

Continue ›

Table: Movies

Title
WALL-E
WALL-G

```
SELECT title FROM movies
WHERE title IN ("WALL-E", "WALL-G");
```

RESET

Exercise 3 — Tasks

1. Find all the Toy Story movies ✓
2. Find all the movies directed by John Lasseter ✓
3. Find all the movies (and director) not directed by John Lasseter ✓
4. Find all the WALL-* movies ✓

Stuck? Read this task's Solution.
Solve all tasks to continue to the next lesson.

Continue ›

Table: Movies

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

```
SELECT DISTINCT title FROM movies
ORDER BY title ASC
LIMIT 5 OFFSET 5;
```

RESET

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 ›

Table: North_american_cities

City	Population
Chicago	2718782
Houston	2195914

```
SELECT city, population FROM north_american_cities
WHERE country = "United States" AND population <= 2718782 AND population >=
    2195914
ORDER BY population DESC
LIMIT 2;
```

RESET

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 ›

Query Results

Cars 2	191452396	368400000
Toy Story 2	245852179	239163000
The Incredibles	261441092	370001000
WALL-E	223808164	297503696
Toy Story 3	415004880	648167031
Toy Story	191796233	170162503
Cars	244082982	217900167
Up	293004164	438338580
Monsters, Inc.	289916256	272900000
A Bug's Life	162798565	200600000
Brave	237283207	301700000

```
SELECT title, Domestic_sales, International_sales
FROM movies
INNER JOIN Boxoffice
ON movies.Id = Boxoffice.Movie_id
```

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.

Finish above Tasks

Query Results

Title	Domestic_sales	International_sales
Finding Nemo	380843261	555900000
Monsters University	268492764	475066843
Ratatouille	206445654	417277164
Cars 2	191452396	368400000
The Incredibles	261441092	370001000
WALL-E	223808164	297503696
Toy Story 3	415004880	648167031
Up	293004164	438338580
A Bug's Life	162798565	200600000
Brave	237283207	301700000

```
SELECT title, Domestic_sales, International_sales
FROM movies
INNER JOIN Boxoffice
ON movies.Id = Boxoffice.Movie_id
WHERE Boxoffice.International_sales > Boxoffice.Domestic_sales
```

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.

Finish above Tasks

Query Results

Title	Rating
WALL-E	8.5
Toy Story 3	8.4
Toy Story	8.3
Up	8.3
Finding Nemo	8.2
Monsters, Inc.	8.1
Ratatouille	8
The Incredibles	8
Toy Story 2	7.9
Monsters University	7.4

```
SELECT title, Rating
FROM movies
INNER JOIN Boxoffice
ON movies.Id = Boxoffice.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 ›

Query Results

Building
1e
2w

```
SELECT DISTINCT Employees.Building
FROM employees
LEFT JOIN Buildings
ON employees.Building = Buildings.Building_name
WHERE employees.Building IS NOT NULL
```

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.

Finish above Tasks

Query Results

Building_name	Capacity
1e	24
1w	32
2e	16
2w	20

```
SELECT *
FROM Buildings
```

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.

Finish above Tasks

Query Results

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

```
SELECT DISTINCT Buildings.building_name, Employees.role
FROM Buildings
LEFT JOIN Employees
ON Buildings.Building_name = Employees.Building;
```

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 ›

Query Results

Name	Role	Building
Yancy I.	Engineer	
Oliver P.	Artist	

```
SELECT Name, Role, Building
FROM Employees
WHERE Building IS NULL
```

RESET

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.

Finish above Tasks

Query Results

Building_name	Role
1w	
2e	

```
SELECT Buildings.Building_name, Employees.Role
FROM Buildings
LEFT JOIN Employees
ON Employees.Building = Buildings.Building_name
WHERE Role IS NULL
```

RESET

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 ›

Query Results

Title	(Boxoffice.Domestic_sales + Boxoffice.International_sales) / 1000000
Toy Story	361.958736
A Bug's Life	363.398565
Toy Story 2	485.015179
Monsters, Inc.	562.816256
Finding Nemo	936.743261
The Incredibles	631.442092
Cars	461.983149
Ratatouille	623.722818
WALL-E	521.31186
Up	731.342744

```
SELECT Movies.Title, (Boxoffice.Domestic_sales + Boxoffice
    .International_sales) / 1000000
FROM Movies
LEFT JOIN Boxoffice
ON Movies.Id = Boxoffice.Movie_id;
```

RESET

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.

Finish above Tasks

Query Results

Title	Ratings_percent
Toy Story	83
A Bug's Life	72
Toy Story 2	79
Monsters, Inc.	81
Finding Nemo	82
The Incredibles	80
Cars	72
Ratatouille	80
WALL-E	85
Up	83

```
SELECT Movies.Title, Boxoffice.Rating * 10 AS ratings_percent
FROM Movies
LEFT JOIN Boxoffice
ON Movies.Id = Boxoffice.Movie_id;
```

RESET

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.

Finish above Tasks

Query Results

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
WHERE Year % 2 = 0
```

RESET

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 ›

Table: Employees

MAX(Years_employed)
9

```
SELECT MAX(Years_employed)
FROM employees;
```

RESET

Exercise 10 — Tasks

1. Find the longest time that an employee has been at the studio ✓
2. For each role, find the average number of years employed by employees in that role
3. 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.

Finish above Tasks

Table: Employees

Role	AVG(Years_employed)
Artist	6
Engineer	3.4
Manager	6

```
SELECT role, AVG(Years_employed)
FROM employees
GROUP BY role;
```

RESET

Exercise 10 — Tasks

1. Find the longest time that an employee has been at the studio ✓
2. For each role, find the average number of years employed by employees in that role ✓
3. 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.

Finish above Tasks

Table: Employees

SUM(Years_employed)	Building
29	1e
36	2w

```
SELECT SUM(Years_employed), Building
FROM employees
GROUP BY Building;
```

RESET

Exercise 10 — Tasks

1. Find the longest time that an employee has been at the studio ✓
2. For each role, find the average number of years employed by employees in that role ✓
3. 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 ›

Table: Employees

Role	COUNT(*)
Artist	5

```
SELECT role, COUNT(*)
FROM employees
GROUP BY Role
LIMIT 1;
```

RESET

Exercise 11 — Tasks

1. Find the number of Artists in the studio (without a **HAVING** clause) ✓
2. Find the number of Employees of each role in the studio
3. 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.

Finish above Tasks

Table: Employees

Role	COUNT (*)
Artist	5
Engineer	5
Manager	3

```
SELECT Role, COUNT (*)
FROM employees
GROUP BY Role;
```

RESET

Exercise 11 — Tasks

1. Find the number of Artists in the studio (without a **HAVING** clause) ✓
2. Find the number of Employees of each role in the studio ✓
3. 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.

Finish above Tasks

Table: Employees

Role	SUM(Years_employed)
Engineer	17

```
SELECT  Role, SUM(Years_employed)
FROM employees
WHERE Role = 'Engineer'
GROUP BY Role;
```

RESET

Exercise 11 — Tasks

1. Find the number of Artists in the studio (without a **HAVING** clause) ✓
2. Find the number of Employees of each role in the studio ✓
3. 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 ›

Query Results

Director	COUNT(Title)
Andrew Stanton	2
Brad Bird	2
Brenda Chapman	1
Dan Scanlon	1
John Lasseter	5
Lee Unkrich	1
Pete Docter	2

```
SELECT Director, COUNT(Title)
FROM movies
GROUP BY Director;
```

RESET

Exercise 12 — Tasks

1. Find the number of movies each director has directed ✓
2. 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.

Finish above Tasks

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 Movies.Director,SUM(Boxoffice.Domestic_sales + Boxoffice
    .International_sales) AS total_sales
FROM movies
LEFT JOIN Boxoffice
ON Movies.Id = Boxoffice.Movie_id
GROUP BY Director;
```

RESET

Exercise 12 — Tasks

1. Find the number of movies each director has directed ✓
2. 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 ›

Query Results

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	Toy Story 4	John Lasseter	2005	90

```
INSERT INTO Movies
VALUES (4, 'Toy Story 4', 'John Lasseter', 2005, 90);
```

RUN QUERY RESET

Exercise 13 — Tasks

1. Add the studio's new production, **Toy Story 4** to the list of movies (you can use any director)
   ✓
2. 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.

Finish above Tasks

Query Results

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	340	270

```
INSERT INTO Boxoffice
VALUES (4, 8.7, 340, 270);
```

RUN QUERY RESET

Exercise 13 — Tasks

1. Add the studio's new production, **Toy Story 4** to the list of movies (you can use any director) ✓
2. 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.

Continue ›

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	1899	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 Director = 'John Lasseter'
WHERE Id = 2;
```

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.

Finish above Tasks

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

```
UPDATE Movies
SET Year = 1999
WHERE Id = 3;
```

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.

Finish above Tasks

Table: Movies

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
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

```
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 ›

Table: Movies

Id	Title	Director	Year	Length_minutes
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
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

```
DELETE FROM movies
WHERE Year < 2005;
```

RUN QUERY RESET

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.

Finish above Tasks

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

```
DELETE FROM movies
WHERE Director = 'Andrew Stanton';
```

RUN QUERY RESET

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 ›

Table: Database

Missing table...

```
CREATE TABLE Database (
id INTEGER PRIMARY KEY,
Name TEXT,
Version number,
Download_count number)
```

RUN QUERY RESET

Exercise 16 — Tasks

1. 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 ›

Table: Movies

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

```
ALTER TABLE Movies
ADD column Aspect_ratio
DEFAULT FLOAT;
```

RUN QUERY RESET

Exercise 17 — Tasks

1. Add a column named **Aspect_ratio** with a **FLOAT** data type to store the aspect-ratio each movie was released in. ✓
2. 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.

Finish above Tasks

Table: Movies

4	Monsters, Inc.	Pete Docter	2001	92	FLOAT	English
5	Finding Nemo	Andrew Stanton	2003	107	FLOAT	English
6	The Incredibles	Brad Bird	2004	116	FLOAT	English
7	Cars	John Lasseter	2006	117	FLOAT	English
8	Ratatouille	Brad Bird	2007	115	FLOAT	English
9	WALL-E	Andrew Stanton	2008	104	FLOAT	English
10	Up	Pete Docter	2009	101	FLOAT	English
11	Toy Story 3	Lee Unkrich	2010	103	FLOAT	English
12	Cars 2	John Lasseter	2011	120	FLOAT	English
13	Brave	Brenda Chapman	2012	102	FLOAT	English
14	Monsters University	Dan Scanlon	2013	110	FLOAT	English

```
ALTER TABLE Movies
ADD column Language
TEXT DEFAULT 'English'
```

RUN QUERY RESET

Exercise 17 — Tasks

1. Add a column named **Aspect_ratio** with a **FLOAT** data type to store the aspect-ratio each movie was released in. ✓
2. 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 ›

Query Results

Id	Title	Director	Year	Length_minutes

```
DROP TABLE IF EXISTS Movies;
```

RUN QUERY RESET

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.

Finish above Tasks

Query Results

Id	Title	Director	Year	Length_minutes

```
DROP TABLE IF EXISTS Boxoffice;
```

RUN QUERY RESET

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.

Continue ›