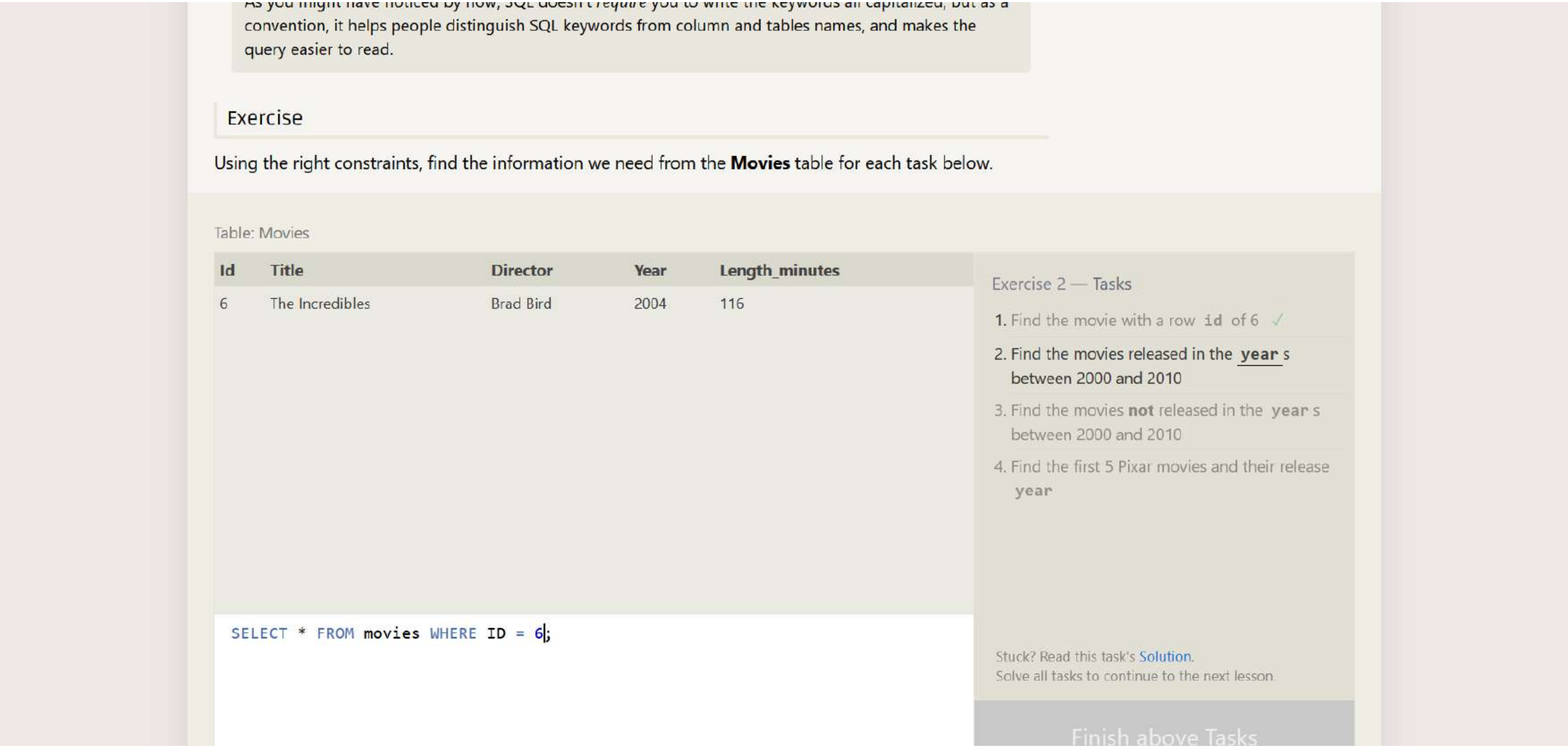


As you might have noticed by now, SQL doesn't *require* you to write the keywords all capitalized, but as a convention, it helps people distinguish SQL keywords from column and tables names, and makes the query easier to read.

Exercise

Using the right constraints, find the information we need from the **Movies** table for each task below.

Table: Movies

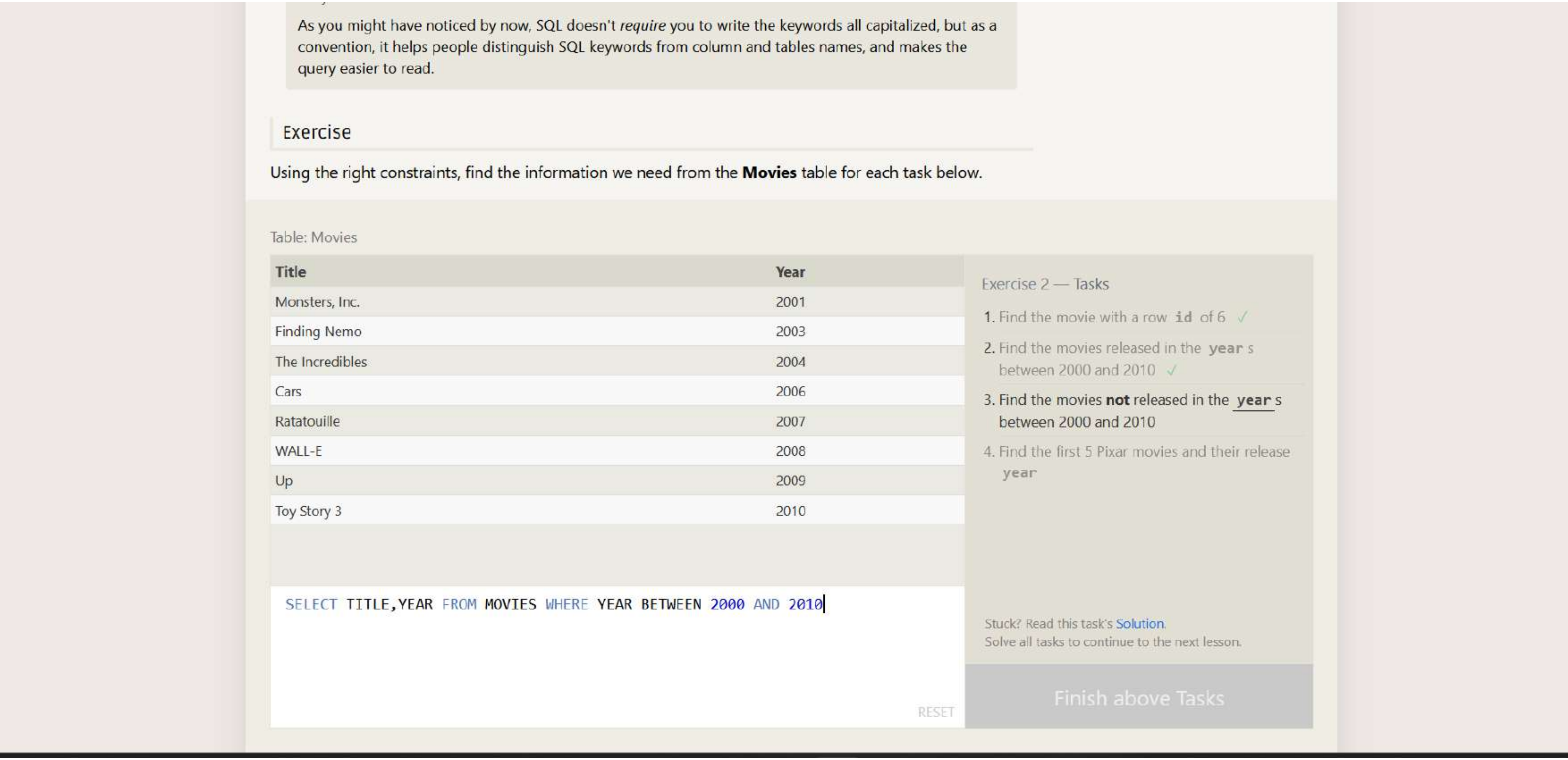
Id	Title	Director	Year	Length_minutes
6	The Incredibles	Brad Bird	2004	116

```
SELECT * FROM movies WHERE ID = 6;
```

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.



As you might have noticed by now, SQL doesn't *require* you to write the keywords all capitalized, but as a convention, it helps people distinguish SQL keywords from column and tables names, and makes the query easier to read.

Exercise

Using the right constraints, find the information we need from the **Movies** table for each task below.

Table: Movies

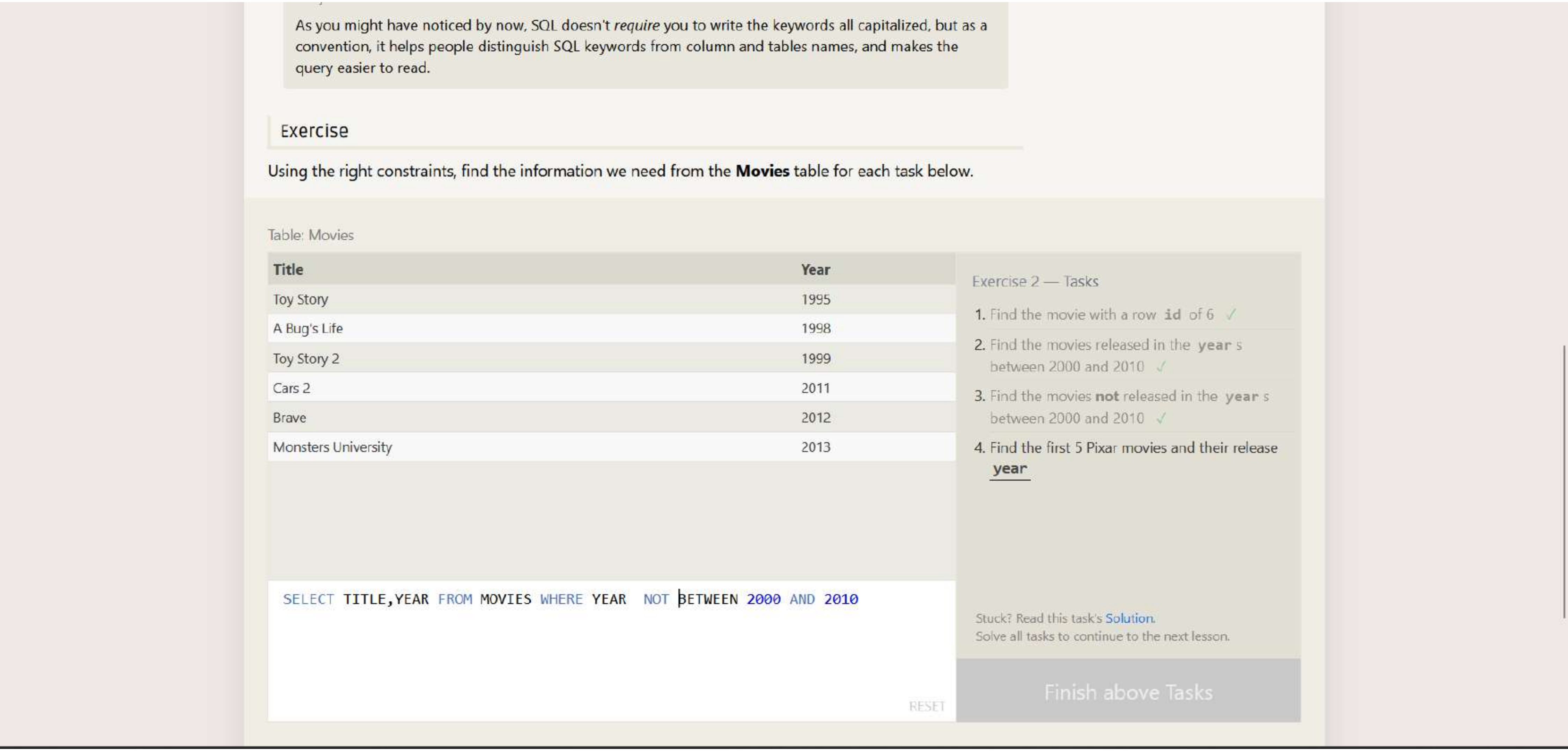
Title	Year
Monsters, Inc.	2001
Finding Nemo	2003
The Incredibles	2004
Cars	2006
Ratatouille	2007
WALL-E	2008
Up	2009
Toy Story 3	2010

```
SELECT TITLE,YEAR FROM MOVIES WHERE YEAR BETWEEN 2000 AND 2010
```

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.



As you might have noticed by now, SQL doesn't *require* you to write the keywords all capitalized, but as a convention, it helps people distinguish SQL keywords from column and tables names, and makes the query easier to read.

Exercise

Using the right constraints, find the information we need from the **Movies** table for each task below.

Table: Movies

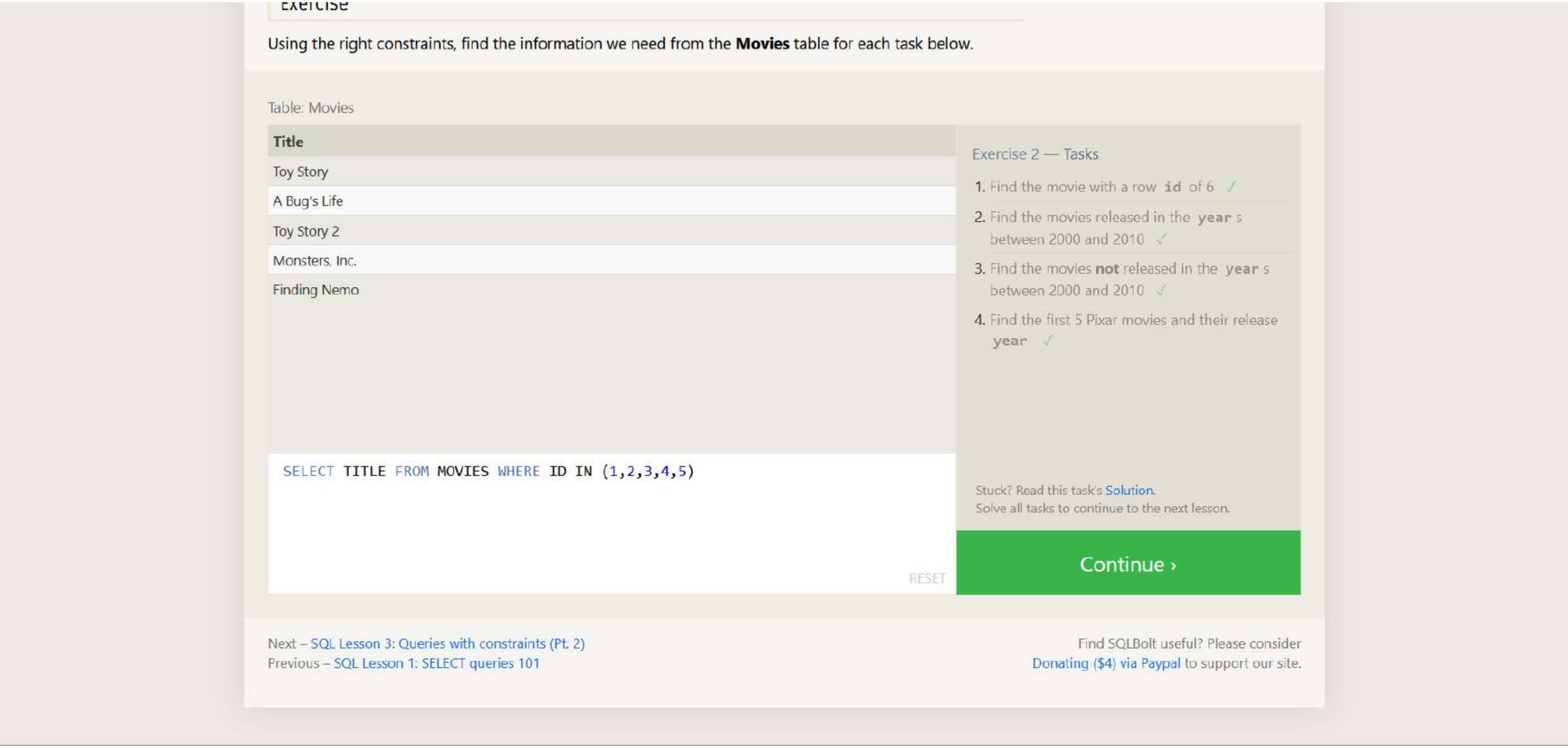
Title	Year
Toy Story	1995
A Bug's Life	1998
Toy Story 2	1999
Cars 2	2011
Brave	2012
Monsters University	2013

```
SELECT TITLE,YEAR FROM MOVIES WHERE YEAR  NOT BETWEEN 2000 AND 2010
```

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.



Exercise

Using the right constraints, find the information we need from the **Movies** table for each task below.

Table: Movies

Title
Toy Story
A Bug's Life
Toy Story 2
Monsters, Inc.
Finding Nemo

```
SELECT TITLE FROM MOVIES WHERE ID IN (1,2,3,4,5)
```

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.

Next – SQL Lesson 3: Queries with constraints (Pt. 2)
Previous – SQL Lesson 1: SELECT queries 101

Find SQLBolt useful? Please consider
Donating ($4) via Paypal to support our site.