Exercise 4: SQL Basics - DML

dr Szymon Murawski

Important! The following queries will permanently change the database. If you want to restore the database to default, execute the query in 'install.sql' file on Moodle.

Task 1: Insert new row into relation movies with the following data: movie_id equal to 11, title 'Madagascar', year of production 2005 and price 10.5

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	9,5	13
7	Ronin	1998	10,5	13
2	Ghostbusters	1984	6,5	12
4	Taxi Driver	1976	6	17
1	Star Wars Episode IV: A New Hope	1979	11,5	12
3	Terminator	1984	9,5	15
11	Madagascar	2005	10,5	

Task 2: Remove all movies produced in 2005

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	9,5	13
7	Ronin	1998	10,5	13
2	Ghostbusters	1984	6,5	12
4	Taxi Driver	1976	6	17
1	Star Wars Episode IV: A New Hope	1979	11,5	12
3	Terminator	1984	9,5	15

Task 3: Increase price of all movies produced before 1980 by 0.5

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	9,5	13
7	Ronin	1998	10,5	13
2	Ghostbusters	1984	6,5	12
3	Terminator	1984	9,5	15
4	Taxi Driver	1976	6,5	17
1	Star Wars Episode IV: A New Hope	1979	12	12

Task 4: Insert new row into relation movies with the following data: movie_id equal to 12, title 'The Incredibles', year of production 2004 and price 9.5

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	9,5	13
7	Ronin	1998	10,5	13
2	Ghostbusters	1984	6,5	12
3	Terminator	1984	9,5	15
4	Taxi Driver	1976	6,5	17
1	Star Wars Episode IV: A New Hope	1979	12	12
12	The Incredibles	2004	9,5	

Task 5: Delete all the movies in which no actor played

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	9,5	13
7	Ronin	1998	10,5	13
2	Ghostbusters	1984	6,5	12
3	Terminator	1984	9,5	15
4	Taxi Driver	1976	6,5	17
1	Star Wars Episode IV: A New Hope	1979	12	12

Task 6: Set the price of movie 'Taxi Driver' to 5

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	9,5	13
7	Ronin	1998	10,5	13
2	Ghostbusters	1984	6,5	12
3	Terminator	1984	9,5	15
1	Star Wars Episode IV: A New Hope	1979	12	12
4	Taxi Driver	1976	5	17

Task 7: Increase by 1 price of all movies, that were rented more than twice

Table movies should now look like this:

movie_id	title	year	price	age_restriction
5	Platoon	1986	5	18
6	Frantic	1988	8,5	15
8	Analyze This	1999	10,5	16
9	Leon: the Professional	1994	8,5	16
10	Mission Impossible	1996	10,5	13
7	Ronin	1998	11,5	13
2	Ghostbusters	1984	7,5	12
4	Taxi Driver	1976	6	17
1	Star Wars Episode IV: A New Hope	1979	13	12
3	Terminator	1984	10,5	15

Task 8: Insert new copy of a movie 'Platoon' into the relation. Do not explicitly write movie_id or assign copy_id, instead try getting them from some query

Table copies should now look like this:

copy_id	movie_id	available
1	1	1
2	1	0
3	2	1
4	3	1
5	3	0
6	3	1
7	4	1
8	5	0
9	6	1
10	6	0
11	6	1
12	7	1
13	7	1
14	8	0
15	9	1
16	10	1
17	10	0
18	10	1
19	10	1
20	10	1
21	5	1
22	8	1
23	4	1
24	9	1
25	2	1
26	5	1

Task 9: For every movie that has only one copy, create another one.
Hint: use row_number() over () in SELECT, to get row number in the result

Table copies should now look like this:

copy_id	movie_id	available
1	1	1
2	1	0
3	2	1
4	3	1
5	3	0
6	3	1
7	4	1
8	5	0
9	6	1
10	6	0
11	6	1
12	7	1
13	7	1
14	8	0
15	9	1
16	10	1
17	10	0
18	10	1
19	10	1
20	10	1
21	5	1
22	8	1
23	4	1
24	9	1
25	2	1
26	5	1