

W13-HW8

1.

- a. SELECT title
- b. FROM movies
- c. WHERE vote_count > 100 AND title NOT LIKE '%Star%';

	title
1	Avatar
2	Pirates of the Caribbean: At World's End
3	Spectre
4	The Dark Knight Rises
5	John Carter
6	Spider-Man 3

```
Execution finished without errors.  
Result: 3124 rows returned in 49ms  
At line 1:  
SELECT title  
FROM movies  
WHERE vote_count > 100 AND title NOT LIKE '%Star%';
```

d.

2.

- a. SELECT DISTINCT directors.name
- b. FROM directors
- c. JOIN movies ON directors.id = movies.director_id
- d. WHERE movies.revenue > 1000000 AND movies.vote_average > 7;

	name
1	James Cameron
2	Christopher Nolan
3	Byron Howard
4	Joss Whedon
5	David Yates
6	Peter Jackson

```
Execution finished without errors.  
Result: 379 rows returned in 6ms  
At line 5:  
SELECT DISTINCT directors.name  
FROM directors  
JOIN movies ON directors.id = movies.director_id  
WHERE movies.revenue > 1000000 AND movies.vote_average > 7;
```

e.

3.

- SELECT name
- FROM directors
- WHERE name LIKE '%Peter%';

	name
1	Peter Jackson
2	Peter Berg
3	Peter Sohn
4	Wolfgang Petersen
5	Peter Ramsey
6	Peter Weir

Execution finished without errors.
Result: 36 rows returned in 5ms
At line 10:
SELECT name
FROM directors
WHERE name LIKE '%Peter%';

d.

4.

- SELECT *
- FROM movies
- WHERE strftime('%Y', release_date) = '2000';

	id	original_title	budget	popularity	release_date	revenue	title	vote_average	vote_count	overview	
1	43791	Dinosaur	127500000	26	2000-05-19	354248063	Dinosaur	6.2	542	An orphaned dinosaur raised by lemurs joins an ...	You h
2	43808	How the Grinch Stole Christmas	123000000	45	2000-11-17	345141403	How the Grinch Stole Christmas	6.2	1386	Inside a snowflake exists the magical land of ...	He pu
3	43810	Mission: Impossible II	125000000	54	2000-05-24	546388105	Mission: Impossible II	5.9	1928	With computer genius Luther Stickell at his side ...	Expec
4	43811	The Perfect Storm	120000000	25	2000-03-15	325756637	The Perfect Storm	6.2	597	In October 1991, a confluence of weather ...	The s
5	43843	The Patriot	110000000	23	2000-06-28	215294342	The Patriot	6.8	1099	After proving himself on the field of battle in the ...	Some

Execution finished without errors.
Result: 165 rows returned in 4ms
At line 14:
SELECT *
FROM movies
WHERE strftime('%Y', release_date) = '2000';

d.

5.

- SELECT name
- FROM directors
- WHERE uid BETWEEN 500 AND 5500;

	name
1	James Cameron
2	Gore Verbinski
3	Christopher Nolan
4	Barry Sonnenfeld
5	Ridley Scott
6	Chris Weitz

Execution finished without errors.
Result: 209 rows returned in 5ms
At line 18:
SELECT name
FROM directors
WHERE uid BETWEEN 500 AND 5500;

d.

6.

- a. SELECT title
- b. FROM movies
- c. WHERE vote_average > 7
- d. AND vote_count BETWEEN 100 AND 500;

	title
1	The Flowers of War
2	Oceans
3	The Insider
4	A Very Long Engagement
5	Lolita
6	The Life of David Gale

Execution finished without errors.
Result: 188 rows returned in 3ms
At line 22:
SELECT title
FROM movies
WHERE vote_average > 7
AND vote_count BETWEEN 100 AND 500;

e.

7.

- a. SELECT DISTINCT directors.name
- b. FROM directors
- c. JOIN movies ON directors.id = movies.director_id
- d. WHERE movies.revenue > 1000000 AND movies.popularity < 5;

	name
1	Brett Ratner
2	Matthew Vaughn
3	Bryan Singer
4	Peter Chelsom
5	David Mamet
6	Joe Johnston

Execution finished without errors.
Result: 268 rows returned in 4ms
At line 27:
SELECT DISTINCT directors.name
FROM directors
JOIN movies ON directors.id = movies.director_id
WHERE movies.revenue > 1000000 AND movies.popularity < 5;

e.

8.

- SELECT SUM(revenue) AS total_revenue
- FROM movies
- WHERE director_id IN (
- SELECT id
- FROM directors
- WHERE name LIKE '%Christopher Nolan%');

	total_revenue
1	4227483234

Execution finished without errors.
Result: 1 rows returned in 8ms
At line 32:
SELECT SUM(revenue) AS total_revenue
FROM movies
WHERE director_id IN (
SELECT id
FROM directors
WHERE name LIKE '%Christopher Nolan%'

g.

9.

- SELECT title, release_date
- FROM movies
- WHERE vote_average IS NULL;

```
Execution finished without errors.  
Result: 0 rows returned in 16ms  
At line 40:  
SELECT title, release_date  
FROM movies  
WHERE vote_average IS NULL;
```

10. d.
- a. SELECT title
 - b. FROM movies
 - c. WHERE (budget IS NULL OR revenue IS NULL) AND vote_count > 10;

```
Execution finished without errors.  
Result: 0 rows returned in 16ms  
At line 44:  
SELECT title  
FROM movies  
WHERE (budget IS NULL OR revenue IS NULL) AND vote_count > 10;
```

11. d.
- a. SELECT movies.title
 - b. FROM movies
 - c. JOIN directors ON movies.director_id = directors.id
 - d. WHERE movies.vote_average BETWEEN 6 AND 8
 - e. AND directors.name LIKE 'S%'

	title
1	Spectre
2	Skyfall
3	Spider-Man 2
4	Iron Man 3
5	Spider-Man
6	The BFG

```
Execution finished without errors.
Result: 254 rows returned in 10ms
At line 48:
SELECT movies.title
FROM movies
JOIN directors ON movies.director_id = directors.id
WHERE movies.vote_average BETWEEN 6 AND 8
AND directors.name LIKE 'S%'
```

- f.
- 12.
- SELECT directors.name, COUNT(*) AS num_movies
 - FROM directors
 - JOIN movies ON directors.id = movies.director_id
 - WHERE movies.revenue > 100000 AND movies.popularity < 10
 - GROUP BY directors.name;

	name	num_movies
1	Abel Ferrara	1
2	Adam Rapp	1
3	Adam Rifkin	1
4	Adam Shankman	1
5	Adrienne Shelly	1
6	Agustín Díaz Yanes	1

```
Execution finished without errors.
Result: 637 rows returned in 9ms
At line 54:
SELECT directors.name, COUNT(*) AS num_movies
FROM directors
JOIN movies ON directors.id = movies.director_id
WHERE movies.revenue > 100000 AND movies.popularity < 10
GROUP BY directors.name;
```

- f.
- 13.
- SELECT name
 - FROM directors
 - WHERE id NOT IN (
 - SELECT DISTINCT director_id

- e. FROM movies
- f. WHERE budget <= 1000000);

	name
1	James Cameron
2	Gore Verbinski
3	Sam Mendes
4	Christopher Nolan
5	Andrew Stanton
6	Byron Howard

Execution finished without errors.
Result: 1250 rows returned in 11ms
At line 60:
SELECT name
FROM directors
WHERE id NOT IN (
 SELECT DISTINCT director_id
 FROM movies
 WHERE budget <= 1000000
);

- g.
- 14.
- a. SELECT directors.name, SUM(movies.vote_count) AS total_votes
- b. FROM directors
- c. JOIN movies ON directors.id = movies.director_id
- d. GROUP BY directors.name
- e. HAVING COUNT(*) >= 4
- f. ORDER BY total_votes DESC;

	name	total_votes
1	Christopher Nolan	62653
2	Steven Spielberg	59771
3	Peter Jackson	45533
4	Quentin Tarantino	43339
5	David Fincher	36586
6	Ridley Scott	36243

Execution finished without errors.
Result: 339 rows returned in 17ms
At line 68:
SELECT directors.name, SUM(movies.vote_count) AS total_votes
FROM directors
JOIN movies ON directors.id = movies.director_id
GROUP BY directors.name
HAVING COUNT(*) >= 4
ORDER BY total_votes DESC;

- g.

15.

- a. SELECT directors.name, COUNT(*) AS num_highRated_movies
- b. FROM directors
- c. JOIN movies ON directors.id = movies.director_id
- d. WHERE movies.vote_average > 7.0
- e. GROUP BY directors.name
- f. HAVING num_highRated_movies > 2;

	name	num_highRated_movies
1	Alejandro González Iñárritu	4
2	Alfonso Cuarón	4
3	Alfred Hitchcock	4
4	Ang Lee	4
5	Antoine Fuqua	3
6	Barry Levinson	3

```
Execution finished without errors.  
Result: 67 rows returned in 7ms  
At line 75:  
SELECT directors.name, COUNT(*) AS num_highRated_movies  
FROM directors  
JOIN movies ON directors.id = movies.director_id  
WHERE movies.vote_average > 7.0  
GROUP BY directors.name  
HAVING num_highRated_movies > 2;
```

16.

- g.
- a. SELECT directors.name, AVG(movies.revenue) AS avg_revenue
- b. FROM directors
- c. JOIN movies ON directors.id = movies.director_id
- d. GROUP BY directors.name
- e. HAVING COUNT(*) > 2
- f. ORDER BY avg_revenue DESC
- g. LIMIT 3;

	name	avg_revenue
1	Joss Whedon	987943689.333333
2	James Cameron	840509919.857143
3	David Yates	742971665.333333

Execution finished without errors.
Result: 3 rows returned in 17ms
At line 82:
SELECT directors.name, AVG(movies.revenue) AS avg_revenue
FROM directors
JOIN movies ON directors.id = movies.director_id
GROUP BY directors.name
HAVING COUNT(*) > 2
ORDER BY avg_revenue DESC
LIMIT 3;

h.

17.

- SELECT *
- FROM movies
- WHERE title LIKE '%War%'
- AND release_date BETWEEN '2000-01-01' AND '2009-12-31'
- AND vote_count >= 100
- ORDER BY vote_average DESC;

	id	original_title	budget	popularity	release_date	revenue	title	vote_average	vote_count	overview
1	46403	태국기 휘날리며	0	9	2004-02-05	15	Tae Guk Gi: The Brotherhood of War	7.4	133	In 1950, in South Korea, shoe-shiner Jin-tae Lee
2	43826	Star Wars: Episode III - Revenge of the Sith	113000000	44	2005-05-17	850000000	Star Wars: Episode III - Revenge of the Sith	7.1	4116	Years after the onset of the Clone Wars, the no
3	44732	Lord of War	42000000	34	2005-09-16	24127895	Lord of War	7.0	1335	Yuri Orlov is a globetrotting arms dealer and, ...
4	44823	Pay It Forward	40000000	24	2000-10-12	55707411	Pay It Forward	7.0	432	Like some other kids, 12-year-old Trevor McKin
5	45268	The Assassination of Jesse James by the Coward...	30000000	35	2007-09-02	14711793	The Assassination of Jesse James by the Coward...	7.0	776	Outlaw Jesse James is rumored be the 'fastest t

Execution finished without errors.
Result: 16 rows returned in 7ms
At line 90:
SELECT *
FROM movies
WHERE title LIKE '%War%'
AND release_date BETWEEN '2000-01-01' AND '2009-12-31'
AND vote_count >= 100
ORDER BY vote_average DESC;

g.

18.

- SELECT directors.name, COUNT(*) AS num_movies, AVG(movies.vote_average) AS avg_vote_average
- FROM directors
- JOIN movies ON directors.id = movies.director_id
- WHERE strftime('%Y', release_date) >= '2000'
- GROUP BY directors.name
- ORDER BY num_movies DESC, avg_vote_average DESC
- LIMIT 5;

	name	num_movies	avg_vote_average
1	Clint Eastwood	13	6.9
2	Steven Spielberg	12	6.75
3	Woody Allen	12	6.54166666666667
4	Steven Soderbergh	12	6.25833333333333
5	Robert Rodriguez	12	5.625

Execution finished without errors.

Result: 5 rows returned in 18ms

At line 97:

```
SELECT directors.name, COUNT(*) AS num_movies, AVG(movies.vote_average) AS avg_vote_average
FROM directors
JOIN movies ON directors.id = movies.director_id
WHERE strftime('%Y', release_date) >= '2000'
GROUP BY directors.name
ORDER BY num_movies DESC, avg_vote_average DESC
LIMIT 5;
```

h.