

## Week 14. Homework 9. Advance SQL-III

Please copy your codes and attach screenshots of your outputs under each question when you submit it in Canvas. Your submission file can be in Word or PDF format. The due date for this Homework assignment is April 26 at 1 PM.

### 1. Display the first 10 characters of each movie's title.

```
1 SELECT SUBSTRING(title, 1, 10) AS first_10_characters
2 FROM movies;
3
```

	first_10_characters
1	Avatar
2	Pirates of
3	Spectre
4	The Dark K
5	John Carte
6	Spider-Man
7	Tangled
8	Avengers:
9	Harry Pott
10	Batman v S
11	Superman R
12	Quantum of
13	Pirates of

Execution finished without errors.  
Result: 4773 rows returned in 89ms  
At line 1:  
SELECT SUBSTRING(title, 1, 10) AS first\_10\_characters  
FROM movies;

```
SELECT SUBSTRING(title, 1, 10) AS first_10_characters
FROM movies;
```

### 2. Retrieve the titles of all movies directed by James Cameron.

```
2 FROM movies
3 JOIN directors ON movies.director_id = directors.id
4 WHERE directors.name = 'James Cameron';
5
6
```

	title
1	Avatar
2	Titanic
3	Terminator 2: Judgment Day
4	True Lies
5	The Abyss
6	Aliens
7	The Terminator

Execution finished without errors.  
Result: 7 rows returned in 79ms  
At line 1:  
SELECT movies.title  
FROM movies

```
SELECT movies.title
FROM movies
JOIN directors ON movies.director_id = directors.id
WHERE directors.name = 'James Cameron';
```

### 3. Find the titles of movies with a vote count greater than the average vote count of all movies.

```

1 SELECT title
2 FROM movies
3 WHERE vote_count > (SELECT AVG(vote_count) FROM movies);
4
5
6

```

	title
1	Avatar
2	Pirates of the Caribbean: At ...
3	Spectre
4	The Dark Knight Rises
5	John Carter
6	Spider-Man 3
7	Tangled
8	Avengers: Age of Ultron
9	Harry Potter and the Half-Bloo...
10	Batman v Superman: Dawn of ...
11	Superman Returns
12	Quantum of Solace
13	Pirates of the Caribbean: Dead...

Execution finished without errors.  
Result: 1263 rows returned in 26ms  
At line 1:  
SELECT title  
FROM movies

```

SELECT title
FROM movies
WHERE vote_count > (SELECT AVG(vote_count) FROM movies);

```

4. List the names of directors who have never directed a movie with a budget of 0.

```

1 SELECT DISTINCT directors.name
2 FROM directors
3 WHERE NOT EXISTS (
4     SELECT 1
5     FROM movies
6     WHERE movies.director_id = directors.id
7     AND movies.budget = 0
8 );
9
10
11

```

	name
1	James Cameron
2	Gore Verbinski
3	Sam Mendes
4	Christopher Nolan
5	Andrew Stanton
6	Byron Howard
7	Joss Whedon
8	David Yates
9	Zack Snyder

Result: 1441 rows returned in 7318ms  
At line 1:  
SELECT DISTINCT directors.name  
FROM directors  
WHERE NOT EXISTS (  
SELECT 1  
FROM movies  
WHERE movies.director\_id = directors.id  
AND movies.budget = 0  
);

```

SELECT DISTINCT directors.name
FROM directors
WHERE NOT EXISTS (
    SELECT 1
    FROM movies
    WHERE movies.director_id = directors.id
    AND movies.budget = 0
);

```

5. List movies with 'Life' in the title and a popularity score above the average popularity of all movies.

```
1 SELECT title
2 FROM movies
3 WHERE title LIKE '%Life%'
4 AND popularity > (SELECT AVG(popularity) FROM movies);
5
6
7
```

	title
1	Life of Pi
2	The Secret Life of Walter Mitty
3	Lara Croft Tomb Raider: The ...
4	The Secret Life of Pets
5	The Book of Life
6	The Life Aquatic with Steve ...
7	The Life of David Gale
8	A Bug's Life
9	The Tree of Life

Execution finished without errors.  
Result: 10 rows returned in 54ms  
At line 1:  
SELECT title  
FROM movies

```
SELECT title
FROM movies
WHERE title LIKE '%Life%'
AND popularity > (SELECT AVG(popularity) FROM movies);
```

6. Find directors who have directed movies with a budget of 0.

```
1 SELECT DISTINCT directors.name
2 FROM directors
3 JOIN movies ON directors.id = movies.director_id
4 WHERE movies.budget = 0;
5
6
7
8
```

	name
1	Bo Welch
2	Jay Roach
3	Walt Becker
4	Barry Cook
5	Bob Fosse
6	Shawn Levy
7	Henry Jaglom
8	Lucile Hadzihalilovic
9	Lee Tamahori

Execution finished without errors.  
Result: 908 rows returned in 10ms  
At line 1:  
SELECT DISTINCT directors.name  
FROM directors

```
SELECT DISTINCT directors.name
FROM directors
JOIN movies ON directors.id = movies.director_id
WHERE movies.budget = 0;
```

7. Display the maximum number of votes received by a movie for each year since 2010.

```

1 SELECT SUBSTR(release_date, 1, 4) AS year, MAX(vote_count) AS max_votes
2 FROM movies
3 WHERE SUBSTR(release_date, 1, 4) >= '2010'
4 GROUP BY SUBSTR(release_date, 1, 4);
5
6
7
8
9
10

```

	year	max_votes
1	2010	13752
2	2011	7047
3	2012	11776
4	2013	8806
5	2014	10867
6	2015	9427
7	2016	10995
8	2017	7

```

Execution finished without errors.
Result: 8 rows returned in 58ms
At line 1:
SELECT SUBSTR(release_date, 1, 4) AS year, MAX(vote_count) AS max_votes
FROM movies

```

```

SELECT SUBSTR(release_date, 1, 4) AS year, MAX(vote_count) AS max_votes
FROM movies
WHERE SUBSTR(release_date, 1, 4) >= '2010'
GROUP BY SUBSTR(release_date, 1, 4);

```

8. For movies released after the year 2000, display the year and the minimum revenue generated in that year for movies with more than 10 votes. Exclude years where the minimum revenue was 0.

```

1 SELECT SUBSTR(release_date, 1, 4) AS year, MIN(revenue) AS min_revenue
2 FROM movies
3 WHERE CAST(SUBSTR(release_date, 1, 4) AS INTEGER) > 2000
4 AND vote_count > 10
5 GROUP BY SUBSTR(release_date, 1, 4)
6 HAVING MIN(revenue) > 0;
7
8
9
10
11

```

```

Execution finished without errors.
Result: 0 rows returned in 112ms
At line 1:
SELECT SUBSTR(release_date, 1, 4) AS year, MIN(revenue) AS min_revenue
FROM movies

```

```

SELECT SUBSTR(release_date, 1, 4) AS year, MIN(revenue) AS min_revenue
FROM movies
WHERE CAST(SUBSTR(release_date, 1, 4) AS INTEGER) > 2000
AND vote_count > 10
GROUP BY SUBSTR(release_date, 1, 4)
HAVING MIN(revenue) > 0;

```

9. List the average budget of the movies for each year since 2000, showing only those years where the average budget was more than 2 million.

```

1 SELECT SUBSTR(release_date, 1, 4) AS year, AVG(budget) AS average_budget
2 FROM movies
3 WHERE CAST(SUBSTR(release_date, 1, 4) AS INTEGER) >= 2000
4 GROUP BY SUBSTR(release_date, 1, 4)
5 HAVING AVG(budget) > 2000000;
6
7

```

	year	average_budget
1	2000	33049697.1515152
2	2001	30349816.5300546
3	2002	29173861.5373134
4	2003	31927773.1497006
5	2004	31649250.0049261
6	2005	29666282.8224299
7	2006	28129364.309322
8	2007	30731525.7589744
9	2008	31768662.6622222

Execution finished without errors.  
Result: 17 rows returned in 561ms  
At line 1:  
SELECT SUBSTR(release\_date, 1, 4) AS year, AVG(budget) AS average\_budget

```

SELECT SUBSTR(release_date, 1, 4) AS year, AVG(budget) AS average_budget
FROM movies
WHERE CAST(SUBSTR(release_date, 1, 4) AS INTEGER) >= 2000
GROUP BY SUBSTR(release_date, 1, 4)
HAVING AVG(budget) > 2000000;

```

10. Find the title and tagline of movies where the tagline contains the word 'life', and the vote average is greater than or equal to the average vote average of all movies.

```

3 -- 10. Find the title and tagline of movies where the tagline contains the word
4 SELECT title, tagline
5 FROM movies
6 WHERE tagline LIKE '%life%'
7 AND vote_average >= (
8     SELECT AVG(vote_average)
9     FROM movies
10 );

```

	title	tagline
1	The Fast and the Furious	Live life 1/4 mile at a time.
2	The Curious Case of Benjamin Button	Life isn't measured in minutes, but in moments.
3	Night at the Museum	Where History Comes To Life
4	Django Unchained	Life, liberty and the pursuit of vengeance.
5	The Lovely Bones	The story of a life and everything that came after...
6	Zodiac	There's more than one way to lose your life to a ...
7	What Dreams May Come	After life there is more. The end is just the ...
8	The Lion King	Life's greatest adventure is finding your place in ...
9	The Terminal	Life is waiting.
10	The Majestic	Sometimes your life comes into focus one frame ...
11	Big Fish	An adventure as big as life itself.
12	You've Got Mail	Someone you pass on the street may already be ...
13	Trouble with the Curve	Whatever Life Throws at You
14	The Soloist	Life has a mind of its own

11. Show the difference between the highest and lowest budget for movies released in '1997'.

```

12 -- 11. Show the difference between the highest and lowest
13 SELECT MAX(budget) - MIN(budget) AS budget_difference
14 FROM movies
15 WHERE strftime('%Y', release_date) = '1997';

```

	budget_difference
1	200000000

12. Find the movies with a vote\_average score within the top 10% of all movies, and order them by the number of votes they received.

```

18 SELECT title, vote_average, vote_count
19 FROM movies
20 WHERE vote_average >= (
21     SELECT AVG(vote_average) + 1.645 * (SELECT AVG(vote_average) - MIN(vote_average) FROM movies)
22     FROM movies
23 )
24 ORDER BY vote_count;
25
26 13. Show the title of movies and their corresponding director's name, where the movie has

```

13. Show the title of movies and their corresponding director's name, where the movie has more than 100 votes and its director has directed more than 5 movies in total.

```

27 SELECT m.title, d.name
28 FROM movies m
29 JOIN (
30     SELECT director_id, COUNT(*) as movie_count
31     FROM movies
32     GROUP BY director_id
33     HAVING COUNT(*) > 5
34 ) AS d_count
35 ON m.director_id = d_count.director_id
36 JOIN directors d ON m.director_id = d.id
37 WHERE m.vote_count > 100;
38

```

	title	name
1	Avatar	James Cameron
2	Pirates of the Caribbean: At World's End	Gore Verbinski
3	Spectre	Sam Mendes
4	The Dark Knight Rises	Christopher Nolan
5	Spider-Man 3	Sam Raimi
6	Batman v Superman: Dawn of Justice	Zack Snyder
7	Superman Returns	Bryan Singer
8	Quantum of Solace	Marc Forster
9	Pirates of the Caribbean: Dead Man's Chest	Gore Verbinski
10	The Lone Ranger	Gore Verbinski
11	Man of Steel	Zack Snyder
12	Men in Black 3	Barry Sonnenfeld
13	The Hobbit: The Battle of the Five Armies	Peter Jackson
14	Robin Hood	Ridley Scott

14. List all movies where the second word in the title is 'Love', considering titles with more than one word.

```

40 SELECT title
41 FROM movies
42 WHERE title LIKE '% Love %' OR title LIKE 'Love %';
43

```

	title
1	The Love Guru
2	For Love of the Game
3	Love Actually
4	Love in the Time of Cholera
5	I Love You, Man
6	Must Love Dogs
7	P.S. I Love You
8	Love & Other Drugs
9	Love Ranch
10	Capitalism: A Love Story
11	Everyone Says I Love You
12	Love Happens
13	Love the Coopers
14	I Love You, Beth Cooper

15. Display the titles of movies that contain the director's last name within the movie title. Assume the last name starts after the last space in the director's name.

```

45 SELECT title
46 FROM movies
47 JOIN directors ON movies.director_id = directors.id
48 WHERE title LIKE '%' || SUBSTR(directors.name, INSTR(directors.name, ' ') + 1) || '%';

```

	title
1	Coach Carter
2	An Alan Smithee Film: Burn, Hollywood, Burn
3	Martin Lawrence Live: Runteldat

16. List movies with 'NULL' in their overview or tagline, replacing 'NULL' with 'No data provided'.

```

51 SELECT
52     id,
53     title,
54     COALESCE(overview, 'No data provided') AS overview,
55     COALESCE(tagline, 'No data provided') AS tagline
56 FROM movies
57 WHERE overview IS NULL OR tagline IS NULL;
58
59

```

	id	title	overview	tagline
1	43607	Superman Returns	Superman returns to discover his 5-year absenc...	No data provided
2	43653	Star Trek Beyond	The USS Enterprise crew explores the furthest ...	No data provided
3	43663	Up	Carl Fredricksen spent his entire life dreaming of...	No data provided
4	43675	The Jungle Book	After a threat from the tiger Shere Khan forces ...	No data provided
5	43725	Angels & Demons	Harvard symbologist Robert Langdon investigates...	No data provided
6	43763	G.I. Joe: Retaliation	Framed for crimes against the country, the G.I. J...	No data provided
7	43832	Astérix at the Olympic Games	Astérix and Obélix have to win the Olympic Game...	No data provided
8	43885	The Hunchback of Notre Dame	When Quasi defies the evil Frollo and ventures o...	No data provided
9	43914	The Flowers of War	A Westerner finds refuge with a group of women...	No data provided
10	43978	The Nutcracker: The Untold Story	Set in 1920's Vienna, this a tale of a little girl, ...	No data provided
11	44062	Fantasia 2000	Blending lively music and brilliant animation, this ...	No data provided
12	44071	Evolution	11-year-old Nicolas lives with his mother in a ...	No data provided
13	44089	Top Cat Begins	Top Cat has arrived to charm his way into your ...	No data provided
14	44133	Anna and the King	The story of the romance between the King of ...	No data provided

17. Show the movie titles along with a modified overview where 'a' is replaced by '@', and if the overview is 'NULL', display 'Overview not available'.

```

1 SELECT
2     title AS movie_title,
3     COALESCE(REPLACE(overview, 'a', '@'), 'Overview not available') AS modified_overview
4 FROM movies;
5

```

	movie_title	modified_overview
1	Avatar	In the 22nd century, @ ...
2	Pirates of the Caribbean: At ...	C@pt@in B@rboss@, long ...
3	Spectre	A cryptic mess@ge from Bond...
4	The Dark Knight Rises	Following the de@th of District...
5	John Carter	John C@rter is @ w@r-we@ry,...
6	Spider-Man 3	The seemingly invincible Spide...
7	Tangled	When the kingdom's most ...
8	Avengers: Age of Ultron	When Tony St@rk tries to ...
9	Harry Potter and the Half-Bloo...	As H@rry begins his sixth ye@...

Execution finished without errors.  
Result: 4773 rows returned in 44ms  
At line 1:  
SELECT  
title AS movie\_title,

```

SELECT
    title AS movie_title,
    COALESCE(REPLACE(overview, 'a', '@'), 'Overview not available') AS
modified_overview
FROM movies;

```

18. Find directors with a last name starting with 'Mc' or 'Mac' and list their movies, replacing any occurrence of 'Mc' or 'Mac' in movie titles with 'Mc(Mac)'.

```

1 SELECT directors.name AS director_name, REPLACE(title, 'Mc', 'Mc(Mac)') AS movie_title
2 FROM directors
3 JOIN movies ON directors.id = movies.director_id
4 WHERE directors.name LIKE 'Mc%' OR directors.name LIKE 'Mac%'
5

```

director_name	movie_title
1 McG	Terminator Salvation
2 McG	Charlie's Angels: Full Throttle
3 McG	Charlie's Angels
4 McG	This Means War
5 McG	We Are Marshall
6 McG	3 Days to Kill

```

Execution finished without errors.
Result: 6 rows returned in 50ms
At line 1:
SELECT directors.name AS director_name, REPLACE(title, 'Mc', 'Mc(Mac)') AS movie_title
FROM directors

```

```

SELECT directors.name AS director_name, REPLACE(title, 'Mc', 'Mc(Mac)') AS
movie_title
FROM directors
JOIN movies ON directors.id = movies.director_id
WHERE directors.name LIKE 'Mc%' OR directors.name LIKE 'Mac%';

```