CSCI 585 - Assignment 2 Rubrics

SQL Query

Total Marks: **6 marks**
Distribution of marks is shown in the table below:

Submission Requirements	**Marks Allotted**
Q1.sql / q1.txt	1 mark
Q2.sql / q2.txt	1 mark
Q3.sql / q3.txt	1 mark
Q4.sql / q4.txt	1 mark
Q1.sql / q5.txt	1 mark
Q6.sql / q6.txt	1 mark
Bonus version of Q2	1 mark

FILE FORMAT AND NAMING CONVENTIONS

1. The queries should be either **.sql or .txt.**
2. Name of the IDE used by the student should be mentioned in the query file.

Rubrics - Queries

Following are the considerations we made during grading your assignments:

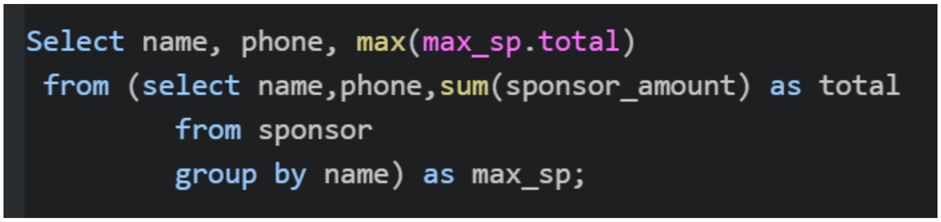
1. -0.25 for each missing criteria in the query selection (query selection criteria mentioned in the table below)
2. -0.25 for each missing table field mentioned in the display criteria (display criteria mentioned in the table below)
3. For the bonus questions, +1 only if implementation involves programming constructs other than the one mentioned in the student's Q2.sql/Q2.txt (example – JOINS in Q2.sql and INNER QUERIES in bonus version of Q2).
4. -0.25 if aggregate functions are not used in q1, q4, q5.
5. -0.25 if division operation is not used in at least one solution version of q2.
6. -0.25 if LIKE keyword is not used in at least one solution version of q2.

Rubrics – Query Analysis

QUERY	QUERY CRITERIA	DISPLAY CRITERIA
Find the sponsor who has sponsored the highest amount in YouTube. Display the sponsor's name, phone number and the total amount sponsored.	1. Sum of sponsor amount 2. Max of the sponsor sum	1. Sponsor name 2. Phone Number 3. Total Sponsored Amount
Find the ratio of likes to view count of each video belonging to any of the channels owned by users having the word "Marvel Entertainment" in them. Display the Video Title, channel name and the ratio in the ascending order of the title.	1. Channel owner name like "%Marvel Entertainment%" 2. Divide likes by views from YT Statistics	1. Video title 2. Channel name 3. Ratio(likes/view)
Find unique user/s with the total number of paid subscribers greater than 100 for their channel/s created on 01.01.2023. Display the username, email, channel name and the subscriber count.	1. Channel created on 01.01.2023. 2. Count (subscription type=" PAID") is greater than 100	1. Username 2. Email 3. Channel name 4. Channel subscription/user subscription count (both acceptable)
Find the average sentiment score for each keyword category. Display the keyword name along with average score such that the highest score is displayed first.	1. Group by keyword 2. Order by keyword in descending order 3. Average of sentiment score	1. Keyword 2. Average score (highest score)
Find the minimum and maximum age of viewers who watched the most commented on video on Taylor Swift's channel. Display the video title, minimum age and the maximum age.	1. Channel owner = Taylor Swift 2. Viewership table to track consumers. 3. Min and max age per video	1. Video title 2. Minimum age 3. Maximum age
Find all the content creators living in the US who have consistently posted at least 1 video each week of the last month. Display their username, channel/s they own and their total subscriber count.	1. Users who live in the US 2. Users who are consistently posting to their channel/s during every week of January.	1. Username 2. Consistent channel name 3. Total subscriber count of the consistent channel/channel subscription count (both acceptable)

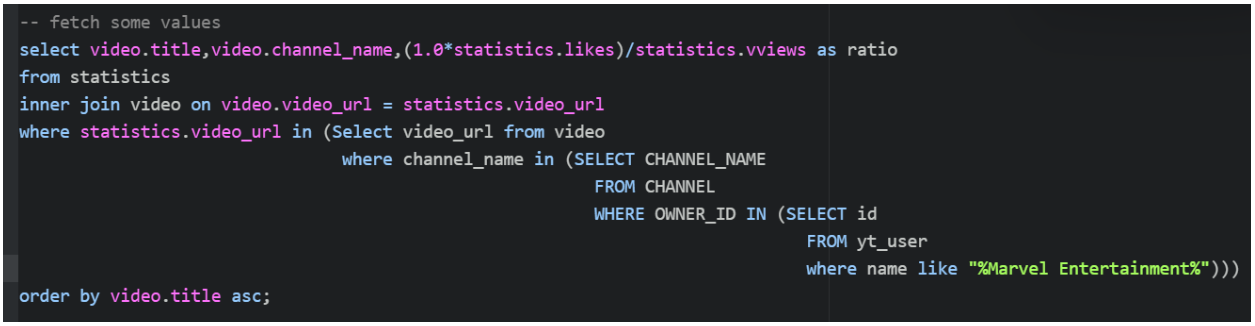
Homework 2 – Sample Solution

Query – 1:



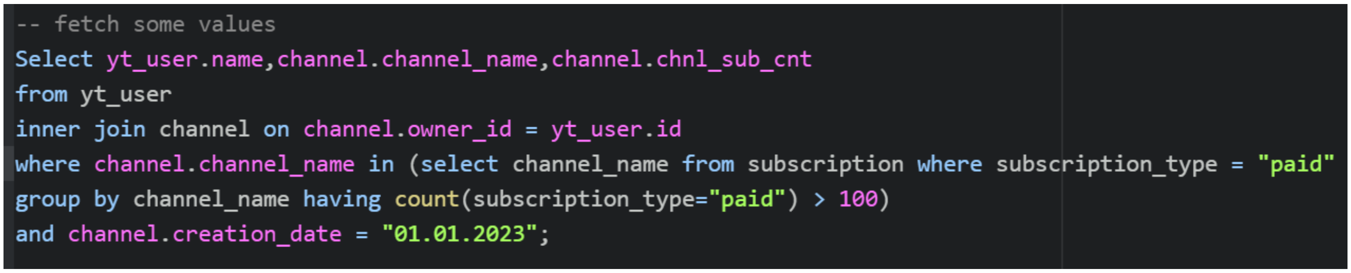
```
Select name, phone, max(max_sp.total)
 from (select name,phone,sum(sponsor_amount) as total
         from sponsor
         group by name) as max_sp;
```

Query – 2:



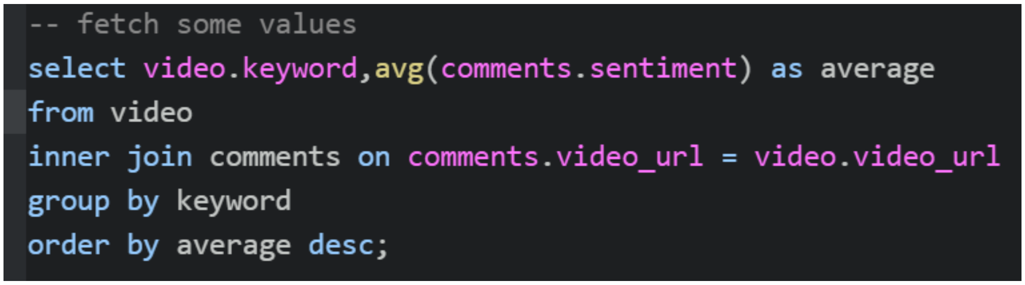
```
-- fetch some values
select video.title,video.channel_name,(1.0*statistics.likes)/statistics.vviews as ratio
from statistics
inner join video on video.video_url = statistics.video_url
where statistics.video_url in (Select video_url from video
                                where channel_name in (SELECT CHANNEL_NAME
                                                         FROM CHANNEL
                                                         WHERE OWNER_ID IN (SELECT id
                                                                              FROM yt_user
                                                                              where name like "%Marvel Entertainment%")))
order by video.title asc;
```

Query – 3:



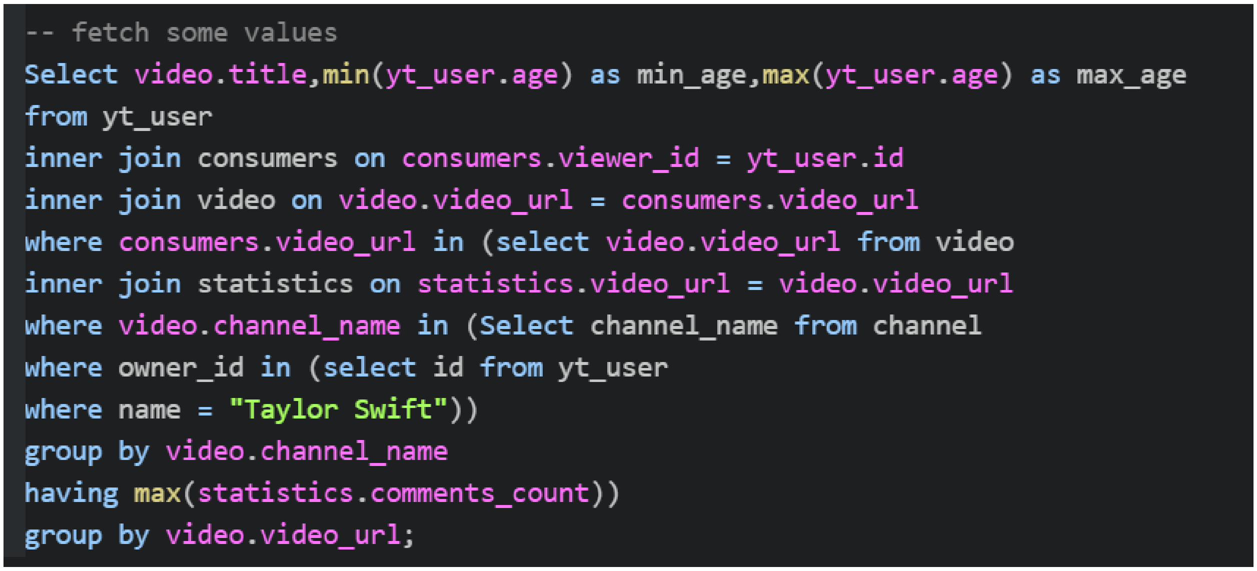
```
-- fetch some values
Select yt_user.name,channel.channel_name,channel.chnl_sub_cnt
from yt_user
inner join channel on channel.owner_id = yt_user.id
where channel.channel_name in (select channel_name from subscription where subscription_type = "paid"
group by channel_name having count(subscription_type="paid") > 100)
and channel.creation_date = "01.01.2023";
```

Query – 4:



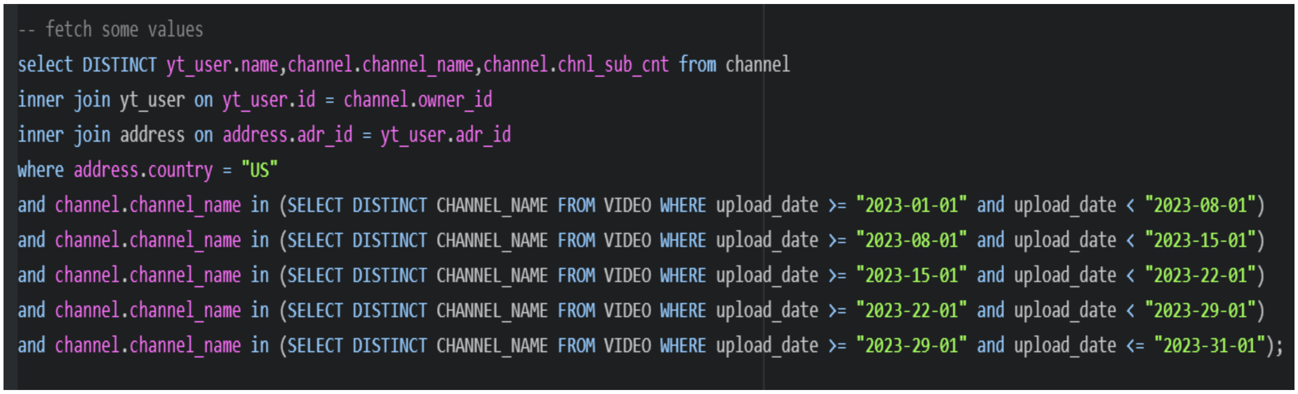
```
-- fetch some values
select video.keyword,avg(comments.sentiment) as average
from video
inner join comments on comments.video_url = video.video_url
group by keyword
order by average desc;
```

Query – 5:



```sql
-- fetch some values
Select video.title,min(yt_user.age) as min_age,max(yt_user.age) as max_age
from yt_user
inner join consumers on consumers.viewer_id = yt_user.id
inner join video on video.video_url = consumers.video_url
where consumers.video_url in (select video.video_url from video
inner join statistics on statistics.video_url = video.video_url
where video.channel_name in (Select channel_name from channel
where owner_id in (select id from yt_user
where name = "Taylor Swift"))
group by video.channel_name
having max(statistics.comments_count))
group by video.video_url;
```

Query – 6:



```sql
-- fetch some values
select DISTINCT yt_user.name,channel.channel_name,channel.chnl_sub_cnt from channel
inner join yt_user on yt_user.id = channel.owner_id
inner join address on address.adr_id = yt_user.adr_id
where address.country = "US"
and channel.channel_name in (SELECT DISTINCT CHANNEL_NAME FROM VIDEO WHERE upload_date >= "2023-01-01" and upload_date < "2023-08-01")
and channel.channel_name in (SELECT DISTINCT CHANNEL_NAME FROM VIDEO WHERE upload_date >= "2023-08-01" and upload_date < "2023-15-01")
and channel.channel_name in (SELECT DISTINCT CHANNEL_NAME FROM VIDEO WHERE upload_date >= "2023-15-01" and upload_date < "2023-22-01")
and channel.channel_name in (SELECT DISTINCT CHANNEL_NAME FROM VIDEO WHERE upload_date >= "2023-22-01" and upload_date < "2023-29-01")
and channel.channel_name in (SELECT DISTINCT CHANNEL_NAME FROM VIDEO WHERE upload_date >= "2023-29-01" and upload_date <= "2023-31-01");
```

Homework 2 – Executable SQL Files

query1.sql

query2.sql

query 3.sql

query4.sql

query 5.sql

query 6.sql