**HW3: Examining data using SQL**

**Individual assignment**

**(Total Points: 33 points)**

**Name:**

**Homework Objectives:**

* Practice advanced queries
* Examine data using queries (remember, real world databases are often imperfect and contain errors)

**Submission Guidelines:**

* Please note that this assignment contains 11 questions.
* Please name your file HW3\_lastname\_firstname (e.g., HW3\_Chen\_Peiyu.doc), and submit your file following appropriate hw submission link by the specified deadline.
* By submitting your homework, you acknowledge that it is submitted as individual work according to the Academic Integrity Policy and the W.P. Carey Honor Code to the best of your ability. If it is not true, explain why it is not true.

**Instructions:**

1. Use MySQL workbench and connect to the course server.
2. The database to use is apps\_and\_crunchbase. Please make sure you execute this statement: use apps\_and\_crunchbase. The relevant schema for your queries is the apps schema (provided as a separate file along with this homework).
3. It is ok to write more than one query in order to answer a question. You just need to document your queries and steps to get the results/answers to each question. Also, some questions ask for row counts only or for just first few entries, please provide just those information.

**Hints**

1. Use date() function to convert time to date, e.g., date(insert\_time). And functions max() and min() can be used on date data types.
2. In SQL, date can be written in the format of “YYYY-MM-DD”. For example, Jan 1, 2000 can be written “2000-01-01”.
3. The top300 table contains the top 300 apps in different lists on app store. And the rank can be calculated using idx column (this is because the data in top300 was populated in the sequence of rank). Specifically, use mod(idx, 300) = 1 to test if the rank is number 1, mod(idx, 300) = 5 to test if the rank is number 5.

**use apps\_and\_crunchbase;**

**1. Retrieve the name of the app that has the highest number of ratings (i.e., rating\_count) and the number of ratings it received.**

**Query:**

**SELECT**

**a.name as App\_Name, b.rating\_count as Number\_of\_Ratings**

**FROM**

**apps a,**

**app\_reviews b**

**WHERE**

**a.id = b.id**

**AND b.rating\_count IN (SELECT**

**MAX(rating\_count)**

**FROM**

**app\_reviews b1);**

**Answer:**

|  |  |
| --- | --- |
| App\_Name | Number\_of\_Ratings |
| Facebook | 2778691 |

**2. Retrieve the name and primary category for the apps that are game-center enabled and whose primary category is not "Games" (use apps table).**

**Query:**

**SELECT**

**distinct name as App\_Name, category\_primary as Primary\_Category, game\_center as Game\_Center**

**FROM**

**apps**

**WHERE**

**category\_primary NOT LIKE 'Games%'**

**AND game\_center >= 1;**

**Row count: 586**

**3. In which primary category, the apps have the highest mean average\_rating?**

**Query:**

**SELECT**

**AVG(a.average\_rating) Average, b.category\_primary as Primary\_Category**

**FROM**

**app\_reviews a,**

**apps b**

**WHERE**

**b.id = a.id**

**GROUP BY b.category\_primary**

**ORDER BY AVG(a.average\_rating) DESC**

**LIMIT 1;**

**Answer:**

|  |  |
| --- | --- |
| Average | Primary\_Category |
| 3.757181 | Book |

**4. List the total number of ratings (rating\_count) received for each primary category? Please list them in descending order of the total number of ratings.**

**Query:**

**SELECT**

**category\_primary AS Primary\_Category,**

**SUM(rating\_count) AS Sum\_Rating**

**FROM**

**apps a,**

**app\_reviews b**

**WHERE**

**a.id = b.id**

**GROUP BY Primary\_Category**

**ORDER BY SUM(rating\_count) DESC;**

**Results: (copy first 5 results)**

|  |  |
| --- | --- |
| Primary\_Category | Sum\_Rating |
| Games | 23142026 |
| Social Networking | 5238760 |
| Entertainment | 4790722 |
| Utilities | 1801520 |
| Lifestyle | 1718952 |

**5. List the primary category, number of ratings and average ratings for the app “Google Earth”.**

**Query:**

**SELECT**

**name AS App\_Name,**

**category\_primary AS Primary\_Category,**

**AVG(rating\_count) AS Average\_Number,**

**(Average\_Rating)**

**FROM**

**apps a,**

**app\_reviews b**

**WHERE**

**b.id = a.id AND name = 'Google Earth';**

**Result:**

|  |  |  |  |
| --- | --- | --- | --- |
| App\_Name | Primary\_Category | Average\_Number | Average\_Rating |
| Google Earth | Travel | 444607 | 3.5 |

**6. (3 pts) Show the top list and list the total number of apps in each top list from the table top300 table. Anything you can say about the findings?**

**Query:**

**SELECT**

**list as Top\_List, COUNT(\*) AS Apps\_Number**

**FROM**

**top300**

**GROUP BY list;**

**Results:**

|  |  |
| --- | --- |
| Top\_List | Apps\_Number |
| Top Free | 109500 |
| Top Grossing | 109500 |
| Top iPad Free | 52200 |
| Top iPad Grossing | 51900 |
| Top iPad Paid | 52200 |
| Top Paid | 109500 |

**Analysis:**

What I can say about the findings is that individuals prefer free apps since the number of free apps is the greatest. This means that the company should advertise free apps more than paid ones and receive profits from the free apps in a different way.

**7. (3 pts) In the "Top Free" list, which two primary categories appear most often?**

**Query:**

**SELECT**

**apps.category\_primary AS Primary\_Category,**

**COUNT(\*) AS Count\_Number**

**FROM**

**top300**

**JOIN**

**apps ON top300.id = apps.id**

**WHERE**

**list = 'Top Free'**

**GROUP BY apps.category\_primary**

**ORDER BY Count\_Number DESC**

**LIMIT 2;**

**Answer:**

|  |  |
| --- | --- |
| Primary\_Category | Count\_Number |
| Games | 6085 |
| Social Networking | 2715 |

**8. (4 pts) What is the shortest time in number of days between an app’s release date and the date an app makes to the top list? What do you think about this information?**

**Query:**

**SELECT**

**DATEDIFF(DATE(insert\_time), release\_date) AS Number\_Days**

**FROM**

**apps a,**

**top300 b**

**WHERE**

**a.id = b.id**

**ORDER BY Number\_Days ASC**

**LIMIT 1;**

**Answer:**

|  |
| --- |
| Number\_Days |
| -1 |

**Analysis:**

The shortest time of days between an app’s release date and when the app makes it to the top list is one day. Since the release date is the day that individuals purchase the app, it is very important to advertise the app on the day that it is first released.

**9. (3 pts) On Aug 31, do we miss any data for any of the top lists? Please provide evidence to support your answer.**

**Query:**

**SELECT DISTINCT**

**list as List, insert\_time as Date\_Time**

**FROM**

**top300**

**WHERE**

**list NOT IN (SELECT DISTINCT**

**list**

**FROM**

**top300**

**WHERE**

**DATE(insert\_time) = '2013-08-31')**

**LIMIT 1;**

**Answer:**

|  |  |
| --- | --- |
| List | Date\_Time |
| Top iPad Grossing | 7/10/2013 16:00 |

**10. (3 pts) Is the apps table complete? That is, do we have data on all apps that appear in the top 300 list? Please provide evidence supporting your answer.**

**Query:**

**SELECT**

**COUNT(\*) as CT**

**FROM**

**top300**

**WHERE**

**id NOT IN (SELECT**

**t.id**

**FROM**

**apps a**

**JOIN**

**top300 t ON a.id = t.id);**

**Answer:**

The apps table is not complete, and we do not have data on all apps that appear in the top 300 list. In an excel spreadsheet or using select all, there are missing values, so the count number is an error.

|  |
| --- |
| CT |
| 383121 |

**11. You have learned that data redundancy is generally bad in a database setting, and you notice that there is redundant information on developer information that exists in both the apps table and the app\_categories table, and you decide to check if the information matches. If the information does not match, it indicates inconsistency and potential problem down the road. Please write a query to check if there is any inconsistency. If your answer points out inconsistency, please report the error rate in terms of the percentage of data not matched on this particular piece of information (developer).**

**Query:**

**SELECT**

**((SELECT**

**COUNT(\*)**

**FROM**

**app\_categories a,**

**apps b**

**WHERE**

**a.id = b.id**

**AND b.developer != a.developer) / (SELECT**

**COUNT(\*)**

**FROM**

**app\_categories a,**

**apps b**

**WHERE**

**a.id = b.id) \* 100) AS Percentage\_Error;**

**Answer:**

|  |
| --- |
| Percentage\_Error |
| 2.5926 |