Google Play Store Database For Dashboard

CSCI E-59
Waree Protprommart

Link to short-form video: https://youtu.be/5fRO668DAEM
Link to long-form video: https://youtu.be/uI2x8uRGWg0

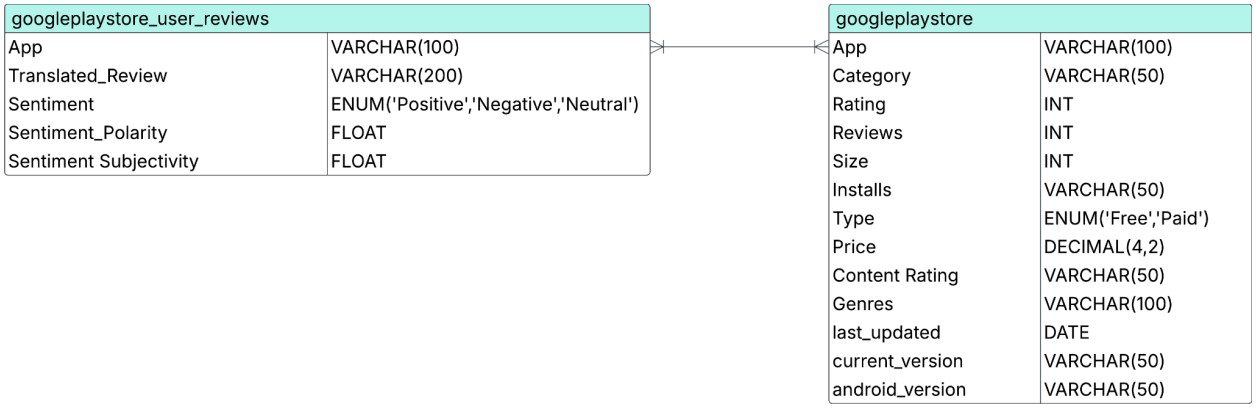
Overview

Objective

Google Play Store or Play Store offers the service as a platform to digitally distribute applications, books, movies, musical singles, television programs, and video games for Android and ChromeOS operating systems ("How Google Play Works", 2025). Android is an operating system that is being used worldwide and has a 71.42% market share worldwide ("IPhone vs. Android User & Revenue Statistics", 2025). Thus, the Google Play Store is an influential and essential platform to many people who are using Android and ChromeOS. In this project, we will use the database that is responsible for storing information related to the applications on the Google Play store, along with the application's user reviews, to reconstruct into the tables where the data analyst can use to construct a report about the performance of the application to the stakeholders of Google. The main things that stakeholders want to know from the Play Store platform are financial metrics such as total revenue of the application, User engagement such as download trends, total downloads, top-performing applications in each category, and customer feedback.

Description of Data

The database stores information related to the applications on the Google Play Store along with the application's user reviews. There are 2 CSV files, including googleplaystore.csv and googleplaystore_user_reviews.csv. Googleplaystore.csv stores information about applications on Google Play (entity) and their attributes including app, category, number of rating, number of reviews, size of the app, number of installs, Paid or Free, Price of app, Age group of user, Genres, last updated date, current version, and android version. Googleplaystore_user_reviews.csv stores the first 100 most relevant reviews for each application (entity) and five other attributes, including app, user review, sentiment, sentiment polarity score, and sentiment subjectivity score. The relationship between these 2 files is a many-to-many relationship between one app's information with many updated dates and many reviews.



The suitable database for this case would be a relational database. Relational database is the right choice because it is a highly structured format with clear entities and attributes, a clear relationship, and SQL queries can support the analysis. The Data for this is from Kaggle: https://www.kaggle.com/datasets/lava18/google-play-store-apps?select=googleplaystore_user_reviews.csv

Technology Used

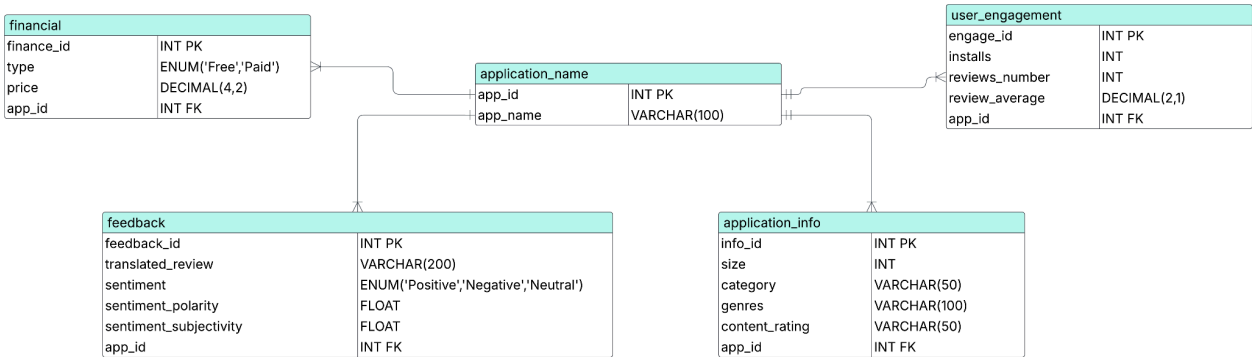
MySQL Workbench, EC2 Instance, Linux Command Line, Jupyter Notebook Python

Relational Database

Diagram the Normalized Schema

The end goal normalized schema that I designed is made to prevent data redundancy and anomalies while separating into three main tables for analysts to produce reports on financial, feedback, application info, and user engagement. The other two tables are for application information, which include the application name and the info tables. The application name table contains information for application names and their ID to prevent the duplication of application names. The application info table contains information regarding that application ID, including size, category, genres, targeted groups for the application (content rating), and info ID for unique information. The financial table includes the financial information of the application, including whether it's free or paid, what the price is if it is paid, and the finance ID. The user engagement table contains information about how engaged the user is in the application. The columns of the table include the engagement ID, installed number, number of reviews, and average review rating. Lastly, the feedback table contains the information about the first 100 most relevant reviews for each app, and their sentiment analysis, polarity score, and subjectivity score. The feedback table has two composite primary keys, which are application ID and feedback ID. This helps to identify each review and which application they are involved with. Application name, user engagement, application info, and financial tables have application ID as their foreign keys,

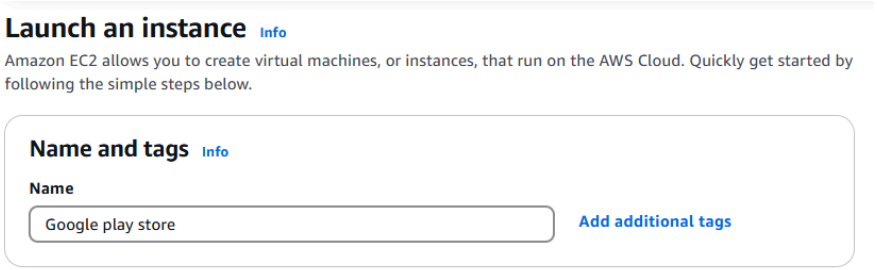
and their relationship between tables is only one application ID to at least one and at most many other tables relationships. The relationship between the application name and the feedback table is one application name to at least one and at most many feedback.



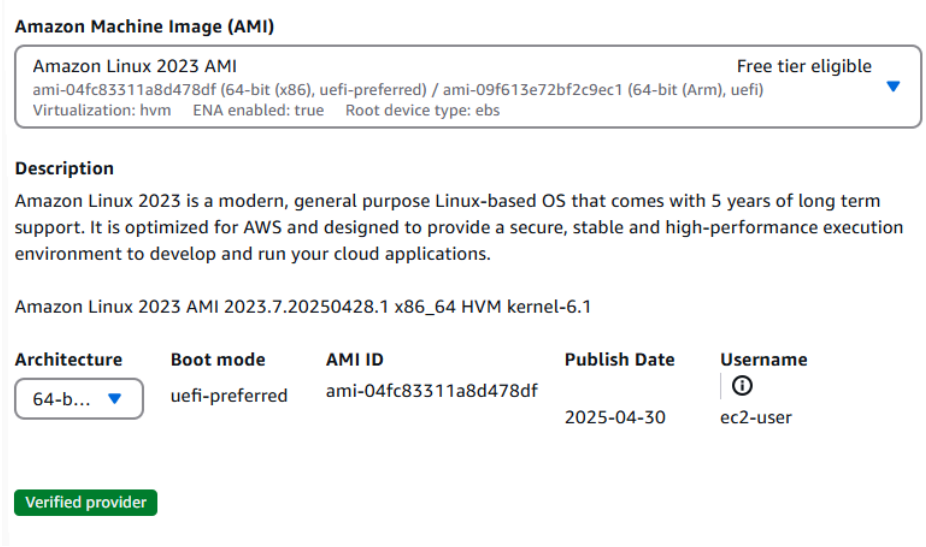
Steps for Creating Normalized Schema

Launch EC2 Instance

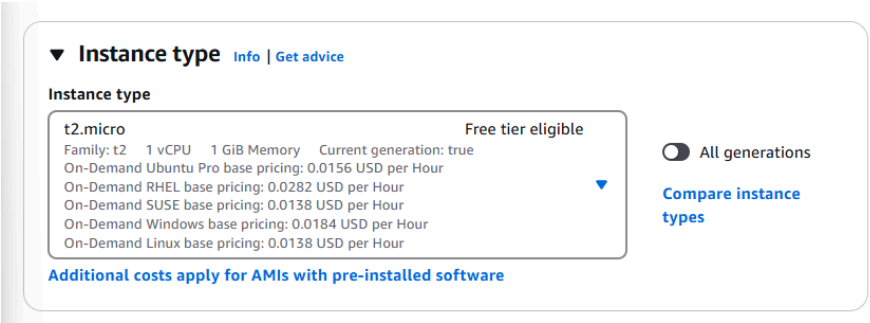
1. Log into AWS account and go to EC2 and instance
2. Click launch instances
3. Name the instance “Google play store”



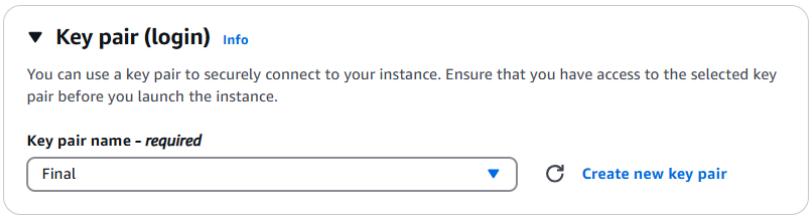
4. For Amazon Machine Image (AMI), choose Amazon Linux 2023 AMI



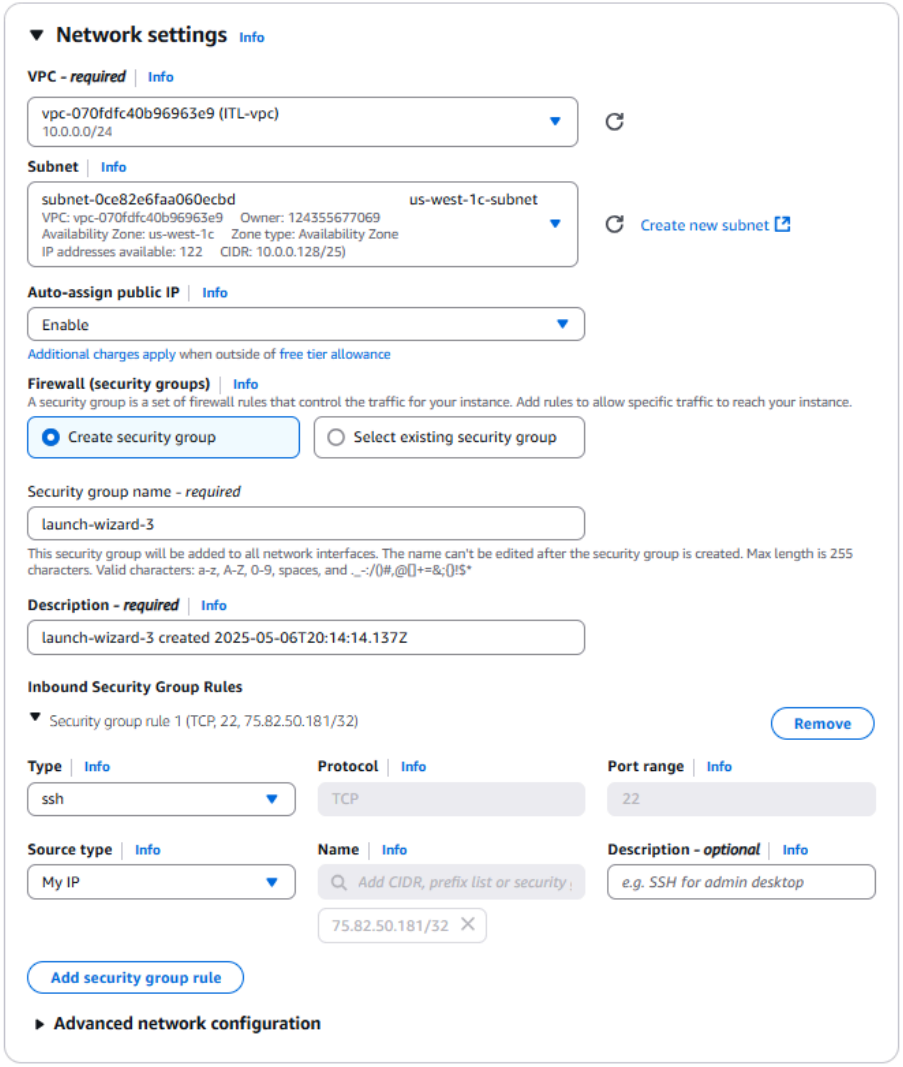
5. For Instance type choose t2.micro



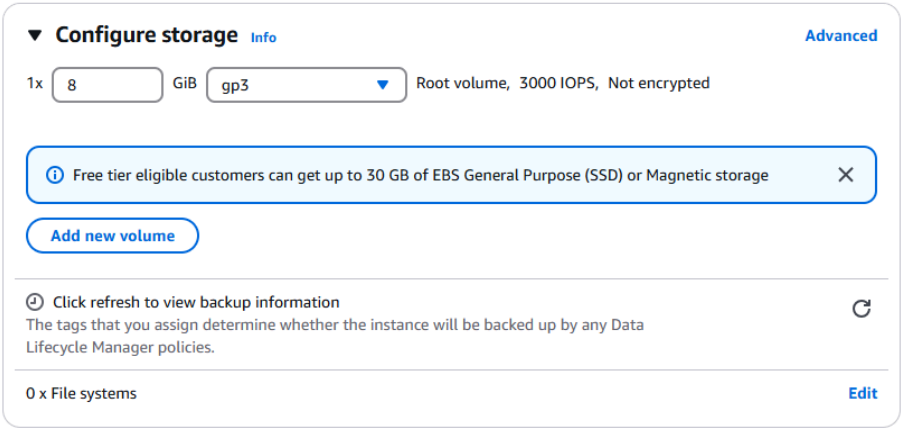
6. Create a new key pair name Final
    a. key pair type ED25519
    b. Private key file format .pem
    c. The private key will be downloaded into computer



7. For Network settings, you can choose default vpc or pre-existing one and subnet is optional.
    a. Enable Auto-assign public IP
    b. For inbound security Group Rules, include ssh port range 22 and your IP with your IP address



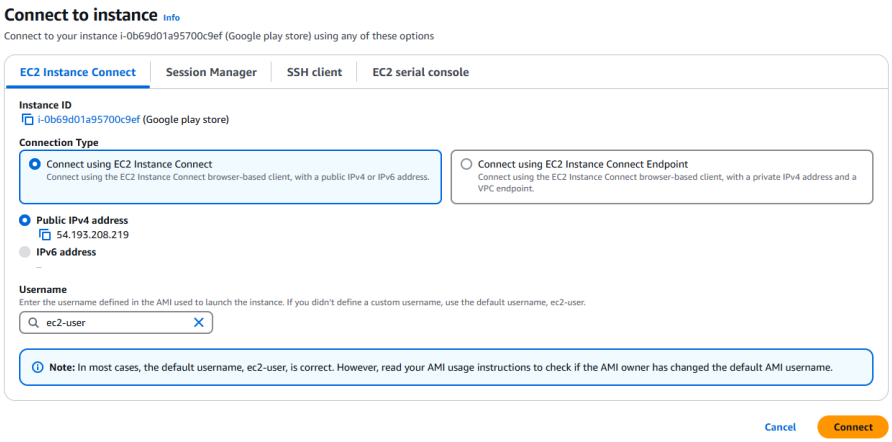
8. For configure storage, choose 8 GiB and gp3 root volume



9. Click launch instance and wait for the status check to be complete

Install MySQL on Instance

1. Connect to instance through EC2 Instance Connect



2. Install MySQL through the command:

sudo yum install wget
wget https://dev.mysql.com/get/mysql84-community-release-el9-1.noarch.rpm
sudo dnf install mysql84-community-release-el9-1.noarch.rpm -y
sudo dnf install mysql-community-server -y

```
[ec2-user@ip-10-0-0-146 ~]$ sudo yum install wget
Amazon Linux 2023 Kernel Livepatch repository                                   148 kB/s |  16 kB     00:00
Package wget-1.21.3-1.amzn2023.0.4.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-0-146 ~]$ wget https://dev.mysql.com/get/mysql84-community-release-el9-1.noarch.rpm
--2025-05-06 20:46:33--  https://dev.mysql.com/get/mysql84-community-release-el9-1.noarch.rpm
Resolving dev.mysql.com (dev.mysql.com)... 23.5.250.172, 2600:1406:2e00:38f::2e31, 2600:1406:2e00:3a1::2e31
Connecting to dev.mysql.com (dev.mysql.com)|23.5.250.172|:443... connected.
HTTP request sent, awaiting response... 302 Moved Temporarily
Location: https://repo.mysql.com//mysql84-community-release-el9-1.noarch.rpm [following]
--2025-05-06 20:46:34--  https://repo.mysql.com//mysql84-community-release-el9-1.noarch.rpm
Resolving repo.mysql.com (repo.mysql.com)... 23.203.223.154, 2600:1406:2e00:880::1d68, 2600:1406:2e00:89c::1d68
Connecting to repo.mysql.com (repo.mysql.com)|23.203.223.154|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13139 (13K) [application/x-redhat-package-manager]
Saving to: 'mysql84-community-release-el9-1.noarch.rpm'

mysql84-community-release-el9-1.noarch.r 100%[=====================================================================>]  12.83K  --.-KB/s    in 0s

2025-05-06 20:46:34 (140 MB/s) - 'mysql84-community-release-el9-1.noarch.rpm' saved [13139/13139]
```

```
[ec2-user@ip-10-0-0-146 ~]$ sudo dnf install mysql84-community-release-el9-1.noarch.rpm -y
Last metadata expiration check: 0:01:07 ago on Tue May  6 20:46:17 2025.
Dependencies resolved.
================================================================================
 Package                    Architecture        Version          Repository     Size
================================================================================
Installing:
 mysql84-community-release  noarch              el9-1            @commandline   13 k

Transaction Summary
================================================================================
Install  1 Package

Total size: 13 k
Installed size: 14 k
Downloading Packages:
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Installing       : mysql84-community-release-el9-1.noarch                 1/1
  Verifying        : mysql84-community-release-el9-1.noarch                 1/1

Installed:
  mysql84-community-release-el9-1.noarch

Complete!
```

```
[ec2-user@ip-10-0-0-146 ~]$ sudo dnf install mysql-community-server -y
MySQL 8.4 LTS Community Server                                   22 MB/s | 1.0 MB     00:00
MySQL Connectors Community                                      2.5 MB/s |  80 kB     00:00
MySQL Tools 8.4 LTS Community                                    12 MB/s | 534 kB     00:00
Dependencies resolved.
================================================================================
 Package                    Architecture        Version              Repository      Size
================================================================================
Installing:
 mysql-community-server     x86_64              8.4.5-1.el9          mysql-8.4-lts-community  50 M
Installing dependencies:
 mysql-community-client     x86_64              8.4.5-1.el9          mysql-8.4-lts-community  3.1 M
 mysql-community-client-plugins x86_64          8.4.5-1.el9          mysql-8.4-lts-community  1.5 M
 mysql-community-common     x86_64              8.4.5-1.el9          mysql-8.4-lts-community  578 k
 mysql-community-icu-data-files x86_64          8.4.5-1.el9          mysql-8.4-lts-community  2.3 M
 mysql-community-libs       x86_64              8.4.5-1.el9          mysql-8.4-lts-community  1.5 M

Transaction Summary
================================================================================
Install  6 Packages

Total download size: 59 M
Installed size: 321 M
Downloading Packages:
(1/6): mysql-community-common-8.4.5-1.el9.x86_64.rpm            7.3 MB/s | 578 kB     00:00
(2/6): mysql-community-client-plugins-8.4.5-1.el9.x86_64.rpm     13 MB/s | 1.5 MB     00:00
(3/6): mysql-community-client-8.4.5-1.el9.x86_64.rpm             24 MB/s | 3.1 MB     00:00
(4/6): mysql-community-icu-data-files-8.4.5-1.el9.x86_64.rpm     29 MB/s | 2.3 MB     00:00
(5/6): mysql-community-libs-8.4.5-1.el9.x86_64.rpm               28 MB/s | 1.5 MB     00:00
(6/6): mysql-community-server-8.4.5-1.el9.x86_64.rpm             74 MB/s |  50 MB     00:00
--------------------------------------------------------------------------------
Total                                                            72 MB/s |  59 MB     00:00
MySQL 8.4 LTS Community Server                                  3.0 MB/s | 3.1 kB     00:00
Importing GPG key 0xA8D3785C:
 Userid     : "MySQL Release Engineering <mysql-build@oss.oracle.com>"
 Fingerprint: BCA4 3417 C3B4 85DD 128E C6D4 B7B3 B788 A8D3 785C
 From       : /etc/pki/rpm-gpg/RPM-GPG-KEY-mysql-2022
Key imported successfully
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Installing       : mysql-community-common-8.4.5-1.el9.x86_64              1/6
  Installing       : mysql-community-client-plugins-8.4.5-1.el9.x86_64      2/6
  Installing       : mysql-community-libs-8.4.5-1.el9.x86_64                3/6
  Running scriptlet: mysql-community-libs-8.4.5-1.el9.x86_64                3/6
  Installing       : mysql-community-client-8.4.5-1.el9.x86_64              4/6
  Installing       : mysql-community-icu-data-files-8.4.5-1.el9.x86_64      5/6
  Running scriptlet: mysql-community-server-8.4.5-1.el9.x86_64              6/6
  Installing       : mysql-community-server-8.4.5-1.el9.x86_64              6/6
  Running scriptlet: mysql-community-server-8.4.5-1.el9.x86_64              6/6
  Verifying        : mysql-community-client-8.4.5-1.el9.x86_64              1/6
  Verifying        : mysql-community-client-plugins-8.4.5-1.el9.x86_64      2/6
  Verifying        : mysql-community-common-8.4.5-1.el9.x86_64              3/6
  Verifying        : mysql-community-icu-data-files-8.4.5-1.el9.x86_64      4/6
  Verifying        : mysql-community-libs-8.4.5-1.el9.x86_64                5/6
  Verifying        : mysql-community-server-8.4.5-1.el9.x86_64              6/6

Installed:
  mysql-community-client-8.4.5-1.el9.x86_64          mysql-community-client-plugins-8.4.5-1.el9.x86_64      mysql-community-common-8.4.5-1.el9.x86_64
  mysql-community-icu-data-files-8.4.5-1.el9.x86_64  mysql-community-libs-8.4.5-1.el9.x86_64                mysql-community-server-8.4.5-1.el9.x86_64

Complete!
```

3. Start MySQL server by the following command:

sudo service mysqld start

```
[ec2-user@ip-10-0-0-146 ~]$ sudo service mysqld start
Redirecting to /bin/systemctl start mysqld.service
```

4. Check the status of MySQL by the following command:

sudo service mysqld status

```
[ec2-user@ip-10-0-0-146 ~]$ sudo service mysqld status
Redirecting to /bin/systemctl status mysqld.service
● mysqld.service - MySQL Server
     Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-05-06 20:53:42 UTC; 13s ago
       Docs: man:mysqld(8)
             http://dev.mysql.com/doc/refman/en/using-systemd.html
    Process: 26743 ExecStartPre=/usr/bin/mysqld_pre_systemd (code=exited, status=0/SUCCESS)
   Main PID: 26858 (mysqld)
     Status: "Server is operational"
      Tasks: 35 (limit: 1111)
     Memory: 456.1M
        CPU: 3.764s
     CGroup: /system.slice/mysqld.service
             └─26858 /usr/sbin/mysqld

May 06 20:53:33 ip-10-0-0-146.us-west-1.compute.internal systemd[1]: Starting mysqld.service - MySQL Server...
May 06 20:53:42 ip-10-0-0-146.us-west-1.compute.internal systemd[1]: Started mysqld.service - MySQL Server.
```

5. Get a temporary password using the following command:

```
sudo grep 'temporary password' /var/log/mysqld.log
```

```
[ec2-user@ip-10-0-0-146 ~]$ sudo grep 'temporary password' /var/log/mysqld.log
2025-05-06T20:53:37.961054Z 6 [Note] [MY-010454] [Server] A temporary password is generated for root@localhost: P<_(i#aaC61U
```

6. Log into mysql with temporary password with the following command:

```
mysql -uroot -p
```

Once logged in, it should be looking like this:

```
[ec2-user@ip-10-0-0-146 ~]$ mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.4.5

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

7. You can reset the password from temporary to your own password with the following command:

```
Mysql_secure_installation
```

```
[ec2-user@ip-10-0-0-146 ~]$ mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

The existing password for the user account root has expired. Please set a new password.

New password:

Re-enter new password:
 ... Failed! Error: Your password does not satisfy the current policy requirements

New password:

Re-enter new password:
The 'validate_password' component is installed on the server.
The subsequent steps will run with the existing configuration
of the component.
Using existing password for root.

Estimated strength of the password: 100
Change the password for root ? ((Press y|Y for Yes, any other key for No) : y
```

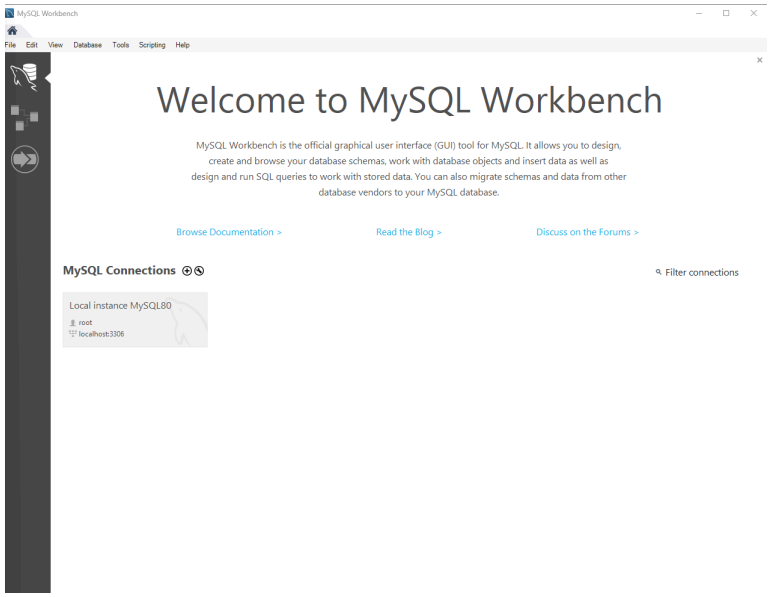
8. Check the database in MySQL with this command:

```
SHOW DATABASES;
```

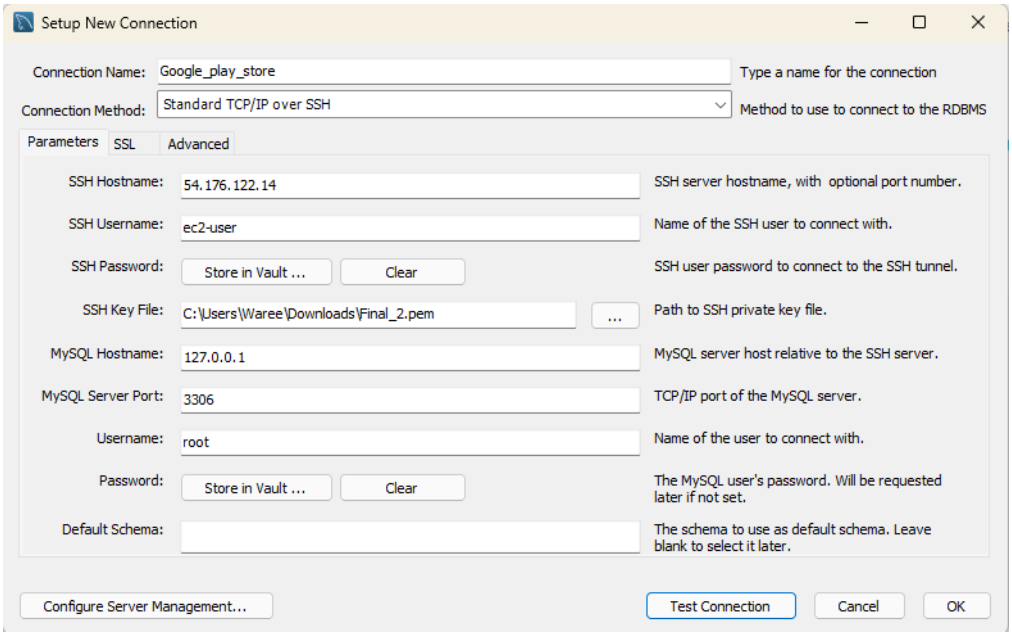
```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)
```

Connect to MySQL Workbench

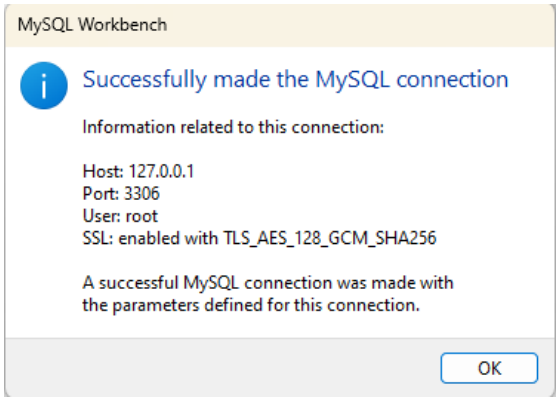
1. Open MySQL Workbench and create a New Connection by Click on + sign



2. Set up new connection to the EC2 instance by
    a. Input the IP addresses with port 22
    b. Input ec2-user for SSH username
    c. Browse location of SSH key file
    d. Leave mysql hostname, server, and root port as default value
    e. Click test connection to test if it is able to connect to MySQL Server in EC2



3. The successful connection should result as this:



Clean the Dataset

Both dataset were being inspected and cleaned by importing to jupyter notebook to remove the NaN values mainly because SQL cannot detect NaN for being Null. Moreover, both dataset got their duplicates removed to ensure that all data records are unique. I inspected the data and chose not to incorporate the last updated, current version, and android version because they contained the same values for the same application name. Only an increased number of reviews is evident and also last updated and version are not necessary for our purpose. The jupyter notebook and cleaned dataset are uploaded to the project Github link:
https://github.com/Protprommart/CSCIE59_SQLDatabase_Project
The python code for removing fields with Nulls is as following:

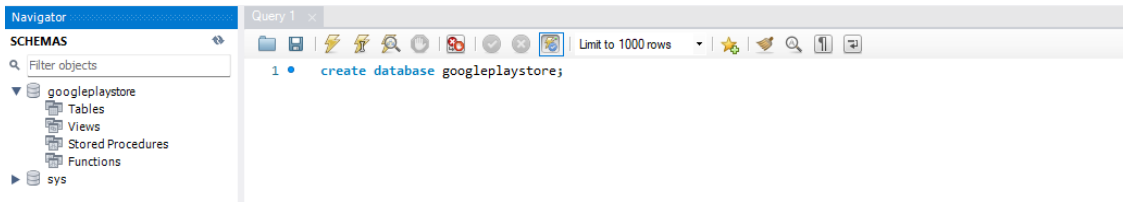
```
import pandas as pd
application_df = pd.read_csv("application.csv")
application_df.dropna(how='any',inplace=True)
```

```
application_df.drop_duplicates(inplace=True)
application_df.to_csv('cleaned_application.csv', index=False)

review_df = pd.read_csv("user_reviews.csv")
review_df.dropna(how='any',inplace=True)
review_df.drop_duplicates(inplace=True)
review_df.to_csv('cleaned_review.csv', index=False)
```

Import Google Play Store Dataset to MySQL

1. Create Database called googleplaystore



2. Choose to use the created database by use the command:

```
use googleplaystore;
```

3. Prepare two tables in the database before importing the csv files by the following commands:

```
CREATE TABLE application_data (
    app_name VARCHAR(100),
    category VARCHAR(50),
    rating FLOAT,
    reviews INT,
    size VARCHAR(50),
    installs VARCHAR(50),
    type ENUM('Free','Paid'),
    price DECIMAL(4,2),
    content_rating VARCHAR(50),
    genres VARCHAR(100),
    last_updated DATE,
    current_version VARCHAR(50),
    android_version VARCHAR(50)
);

CREATE TABLE user_reviews (
    app_name VARCHAR(100),
    Translated_Review VARCHAR(50),
    Sentiment ENUM('Positive','Negative','Neutral'),
    Sentiment_Polarity FLOAT,
```

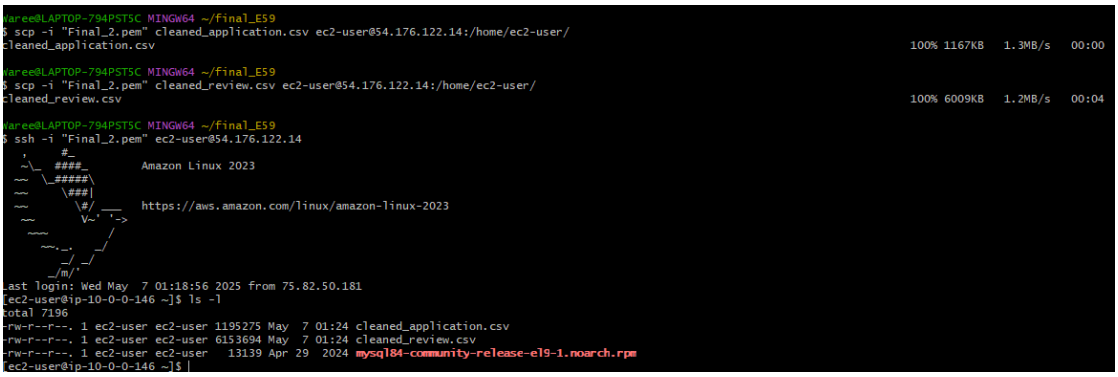
```
    Sentiment_Subjectivity FLOAT
);
```

4. Open command prompt or other equivalent. In the command prompt, use the following commands to move the csv files into EC2 instance

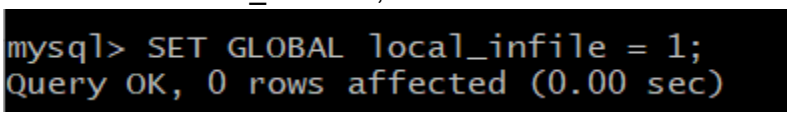
Locate to file path where csv files are located:
cd final_E59

Copy both files to the home directory of the EC2 insurance using scp command:
scp -i "Final_2.pem" cleaned_application.csv ec2-user@54.176.122.14:/home/ec2-user/
scp -i "Final_2.pem" cleaned_review.csv ec2-user@54.176.122.14:/home/ec2-user/

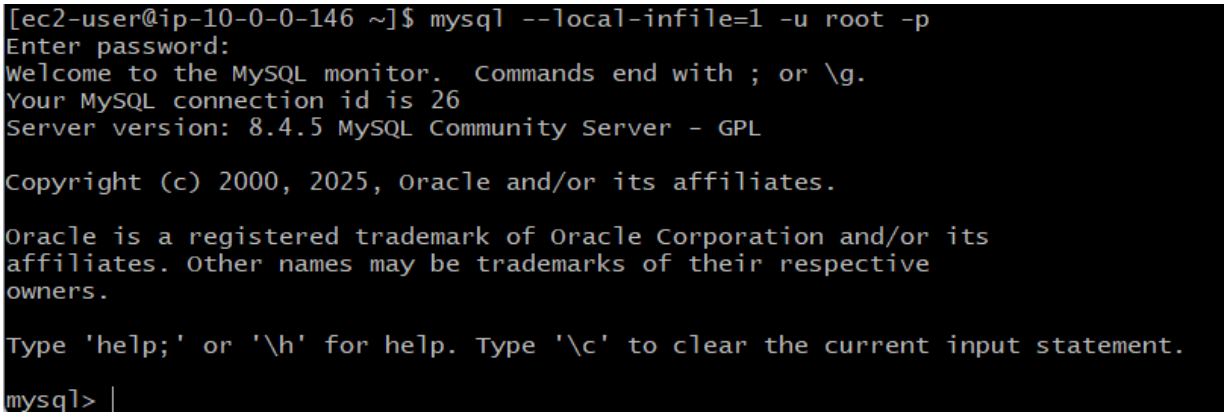
Log into the instance through SSH:
ssh -i "Final_2.pem" ec2-user@54.176.122.14



log into mysql and enable file privilege for MySQL
SET GLOBAL local_infile = 1;



Exit out from MySQL, then log in again with local infile enabled:
mysql --local-infile=1 -u root -p



Inside MySQL, load the files to the created table with the following commands:

```
Use googleplaystore;

LOAD DATA LOCAL INFILE '/home/ec2-user/cleaned_application.csv'
INTO TABLE application_data
FIELDS TERMINATED BY ','
ENCLOSED BY ""
LINES TERMINATED BY '\n'
IGNORE 1 ROWS;

LOAD DATA LOCAL INFILE '/home/ec2-user/cleaned_review.csv'
INTO TABLE user_reviews
FIELDS TERMINATED BY ','
ENCLOSED BY ""
LINES TERMINATED BY '\n'
IGNORE 1 ROWS;
```

```
mysql> Use googleplaystore;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> LOAD DATA LOCAL INFILE '/home/ec2-user/cleaned_application.csv'
    -> INTO TABLE application_data
    -> FIELDS TERMINATED BY ','
    -> ENCLOSED BY '"'
    -> LINES TERMINATED BY '\n'
    -> IGNORE 1 ROWS;
Query OK, 8886 rows affected, 9502 warnings (0.22 sec)
Records: 8886  Deleted: 0  Skipped: 0  Warnings: 9502

mysql> LOAD DATA LOCAL INFILE '/home/ec2-user/cleaned_review.csv'
    -> INTO TABLE user_reviews
    -> FIELDS TERMINATED BY ','
    -> ENCLOSED BY '"'
    -> LINES TERMINATED BY '\n'
    -> IGNORE 1 ROWS;
Query OK, 29692 rows affected, 18833 warnings (0.38 sec)
Records: 29692  Deleted: 0  Skipped: 0  Warnings: 18833
```

5. Check the loaded tables with the following commands:

SELECT * FROM googleplaystore.application_data;
SELECT * FROM googlestore.user_reviews;

```
1 •  SELECT * FROM googleplaystore.application_data;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

app_name	category	rating	reviews	size	installs	type	price	content_rating	genres	last_updated	current_version	android_version
Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0.00	Everyone	Art & Design	0000-00-00	1.0.0	4.0.3 and up
Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0.00	Everyone	Art & Design;Pretend Play	0000-00-00	2.0.0	4.0.3 and up
U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0.00	Everyone	Art & Design	0000-00-00	1.2.4	4.0.3 and up
Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0.00	Teen	Art & Design	0000-00-00	Varies with device	4.2 and up
Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0.00	Everyone	Art & Design;Creativity	0000-00-00	1.1	4.4 and up
Paper flowers instructions	ART_AND_DESIGN	4.4	167	5.6M	50,000+	Free	0.00	Everyone	Art & Design	0000-00-00	1.0	2.3 and up
Smoke Effect Photo Maker - Smoke Editor	ART_AND_DESIGN	3.8	178	19M	50,000+	Free	0.00	Everyone	Art & Design	0000-00-00	1.1	4.0.3 and up
Infinite Painter	ART_AND_DESIGN	4.1	36815	29M	1,000,000+	Free	0.00	Everyone	Art & Design	0000-00-00	6.1.61.1	4.2 and up

```
1 •  SELECT * FROM googleplaystore.user_reviews;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

app_name	Translated_Review	Sentiment	Sentiment_Polarity	Sentiment_Subjectivity
10 Best Foods for You	I like eat delicious food. That's I'm cooking food	Positive	1	0.533333
10 Best Foods for You	This help eating healthy exercise regular basis	Positive	0.25	0.288462
10 Best Foods for You	Works great especially going grocery store	Positive	0.4	0.875
10 Best Foods for You	Best idea us	Positive	1	0.3
10 Best Foods for You	Best way	Positive	1	0.3
10 Best Foods for You	Amazing	Positive	0.6	0.9
10 Best Foods for You	Looking forward app,	Neutral	0	0
10 Best Foods for You	It helpful site ! It help foods get !	Neutral	0	0

Created Normalized Tables

1. Create new tables with constraints and primary key with the following commands:

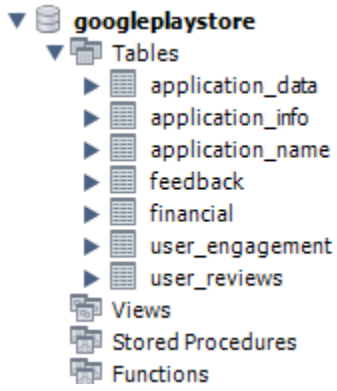
```
CREATE TABLE application_name (
   app_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
   app_name VARCHAR(100) NOT NULL,
   PRIMARY KEY (app_id),
   CONSTRAINT UNIQUE (app_name)
);

CREATE TABLE user_engagement (
   engage_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
   app_id INT UNSIGNED NOT NULL,
   installs DOUBLE,
   reviews_number INT,
   review_average DECIMAL(2,1),
   PRIMARY KEY (engage_id),
   FOREIGN KEY (app_id) REFERENCES application_name(app_id) ON DELETE
CASCADE
);

CREATE TABLE financial (
   finance_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
   app_id INT UNSIGNED NOT NULL,
   type ENUM('Free','Paid'),
   price DECIMAL(10,2),
   PRIMARY KEY (finance_id),
   FOREIGN KEY (app_id) REFERENCES application_name(app_id) ON DELETE
CASCADE
);
```

```
CREATE TABLE application_info (
    info_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
    app_id INT UNSIGNED NOT NULL,
    size VARCHAR(50),
    category VARCHAR(50),
    genres VARCHAR(100),
    content_rating VARCHAR(50),
    PRIMARY KEY (info_id),
    FOREIGN KEY (app_id) REFERENCES application_name(app_id) ON DELETE
CASCADE
);

CREATE TABLE feedback (
    feedback_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
    app_id INT UNSIGNED NOT NULL,
    translated_review VARCHAR(200),
    sentiment ENUM('Positive','Negative','Neutral'),
    sentiment_polarity FLOAT,
    sentiment_subjectivity FLOAT,
    PRIMARY KEY (feedback_id),
    FOREIGN KEY (app_id) REFERENCES application_name(app_id) ON DELETE
CASCADE
);
```



2. Load information onto the tables

Load application names into application_data table:

```
INSERT INTO application_name (app_name)
SELECT DISTINCT app_name
FROM application_data;
```

```
1 •  SELECT * FROM googleplaystore.application_name order by app_id asc;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

app_id	app_name
1	Photo Editor & Candy Camera & Grid & ScrapBook
2	Coloring book moana
3	U Launcher Lite – FREE Live Cool Themes, Hide ...
4	Sketch - Draw & Paint
5	Pixel Draw - Number Art Coloring Book
6	Paper flowers instructions
7	Smoke Effect Photo Maker - Smoke Editor
8	Infinite Painter
9	Garden Coloring Book
10	Kids Paint Free - Drawing Fun
11	Text on Photo - Fonteee
12	Name Art Photo Editor - Focus n Filters
13	Tattoo Name On My Photo Editor
14	Mandala Coloring Book
15	3D Color Pixel by Number - Sandbox Art Coloring
16	Photo Designer - Write your name with shapes
17	350 Diy Room Decor Ideas
18	FlipaClip - Cartoon animation
19	ibis Paint X
20	Logo Maker - Small Business

Load data into user_engagement table:

```
INSERT INTO user_engagement (app_id, reviews_number, review_average, installs)
SELECT application_name.app_id, application_data.reviews, application_data.rating,
CAST(REPLACE(REPLACE(application_data.installs, '+', ''), ',', '') AS DOUBLE)
FROM application_name
INNER JOIN application_data
ON application_name.app_name = application_data.app_name;
```

```
1 •  SELECT * FROM googleplaystore.user_engagement LIMIT 10;
```

Result Grid | Filter Rows: | Edit: | Export/Impor

engage_id	app_id	installs	reviews_number	review_average
1	1	10,000+	159	4.1
2	2	500,000+	967	3.9
3	3	5,000,000+	87510	4.7
4	4	50,000,000+	215644	4.5
5	5	100,000+	967	4.3
6	6	50,000+	167	4.4
7	7	50,000+	178	3.8
8	8	1,000,000+	36815	4.1
9	9	1,000,000+	13791	4.4
10	10	10,000+	121	4.7
NULL	NULL	NULL	NULL	NULL

Load data into financial table:

```
INSERT INTO financial (app_id, type, price)
SELECT application_name.app_id, application_data.type,
CAST(REPLACE(application_data.price, '$', '') AS DECIMAL(10,2))
FROM application_name
INNER JOIN application_data
ON application_name.app_name = application_data.app_name;
```

```
1 •  SELECT * FROM googleplaystore.financial;
```

Result Grid | Filter Rows: | Edit:

finance_id	app_id	type	price
220	220	Free	0.00
221	221	Free	0.00
222	222	Paid	4.99
223	223	Paid	4.99
224	224	Free	0.00
225	225	Free	0.00
226	226	Free	0.00
227	227	Free	0.00
228	228	Free	0.00
229	229	Free	0.00
230	230	Free	0.00
231	231	Free	0.00
232	232	Free	0.00

Load data into application info table :

INSERT INTO application_info (app_id, size, category, genres, content_rating)
SELECT application_name.app_id, application_data.size, application_data.category, application_data.genres, application_data.content_rating
FROM application_name
INNER JOIN application_data
ON application_name.app_name = application_data.app_name;

```
1 •   SELECT * FROM googleplaystore.application_info LIMIT 10;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

info_id	app_id	size	category	genres	content_rating
1	1	19M	ART_AND_DESIGN	Art & Design	Everyone
2	2	14M	ART_AND_DESIGN	Art & Design;Pretend Play	Everyone
3	3	8.7M	ART_AND_DESIGN	Art & Design	Everyone
4	4	25M	ART_AND_DESIGN	Art & Design	Teen
5	5	2.8M	ART_AND_DESIGN	Art & Design;Creativity	Everyone
6	6	5.6M	ART_AND_DESIGN	Art & Design	Everyone
7	7	19M	ART_AND_DESIGN	Art & Design	Everyone
8	8	29M	ART_AND_DESIGN	Art & Design	Everyone
9	9	33M	ART_AND_DESIGN	Art & Design	Everyone
10	10	3.1M	ART_AND_DESIGN	Art & Design;Creativity	Everyone
NULL	NULL	NULL	NULL	NULL	NULL

Load data into feedback table:

INSERT INTO feedback (app_id, translated_review, sentiment, sentiment_polarity, sentiment_subjectivity)
SELECT application_name.app_id, user_reviews.Translated_Review, user_reviews.Sentiment, user_reviews.Sentiment_Polarity, user_reviews.Sentiment_Subjectivity
FROM application_name
INNER JOIN user_reviews
ON application_name.app_name = user_reviews.app_name;

```
1 •   SELECT * FROM googleplaystore.feedback LIMIT 10;
```

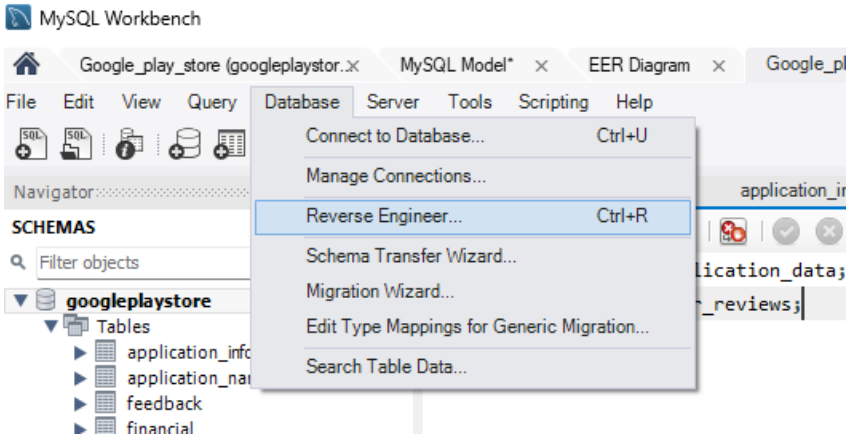
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch r

feedback_id	app_id	translated_review	sentiment	sentiment_polarity	sentiment_subjectivity
1	1079	I like eat delicious food. That's I'm cooking food	Positive	1	0.533333
2	1079	This help eating healthy exercise regular basis	Positive	0.25	0.288462
3	1079	Works great especially going grocery store	Positive	0.4	0.875
4	1079	Best idea us	Positive	1	0.3
5	1079	Best way	Positive	1	0.3
6	1079	Amazing	Positive	0.6	0.9
7	1079	Looking forward app,	Neutral	0	0
8	1079	It helpful site ! It help foods get !	Neutral	0	0
9	1079	good you.	Positive	0.7	0.6
10	1079	Useful information The amount spelling errors q...	Positive	0.2	0.1
NULL	NULL	NULL	NULL	NULL	NULL

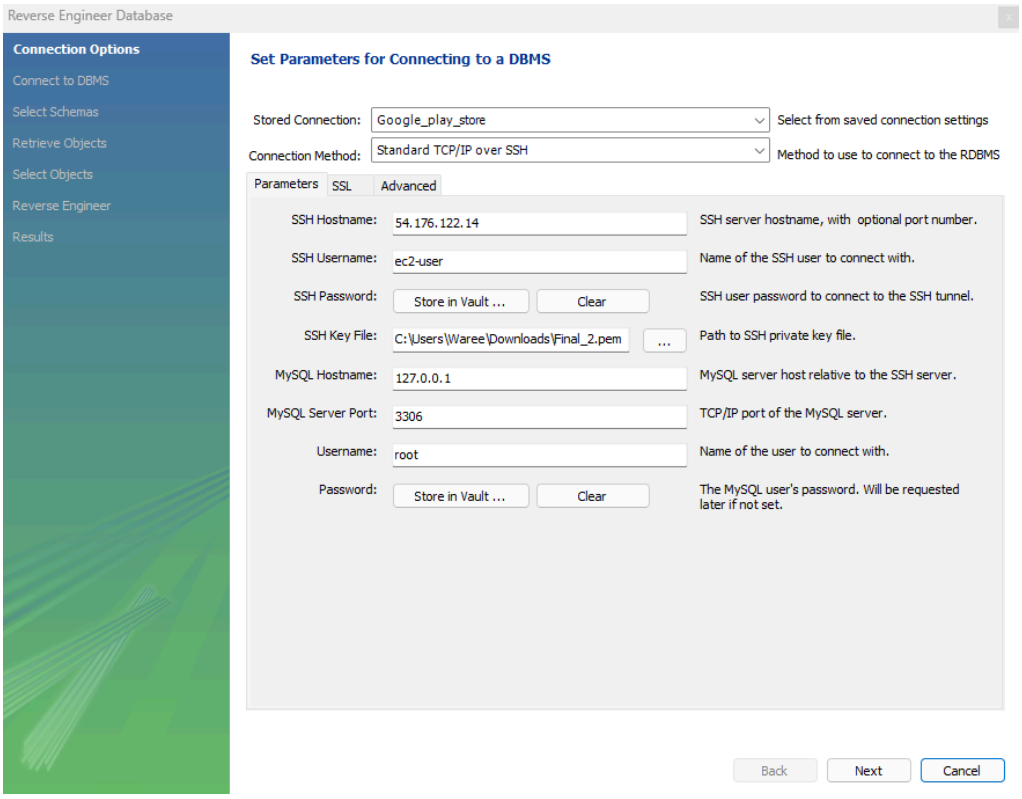
3. Delete application_data and user_reviews tables with the following command:

```
DROP TABLE application_data;
DROP TABLE user_reviews;
```

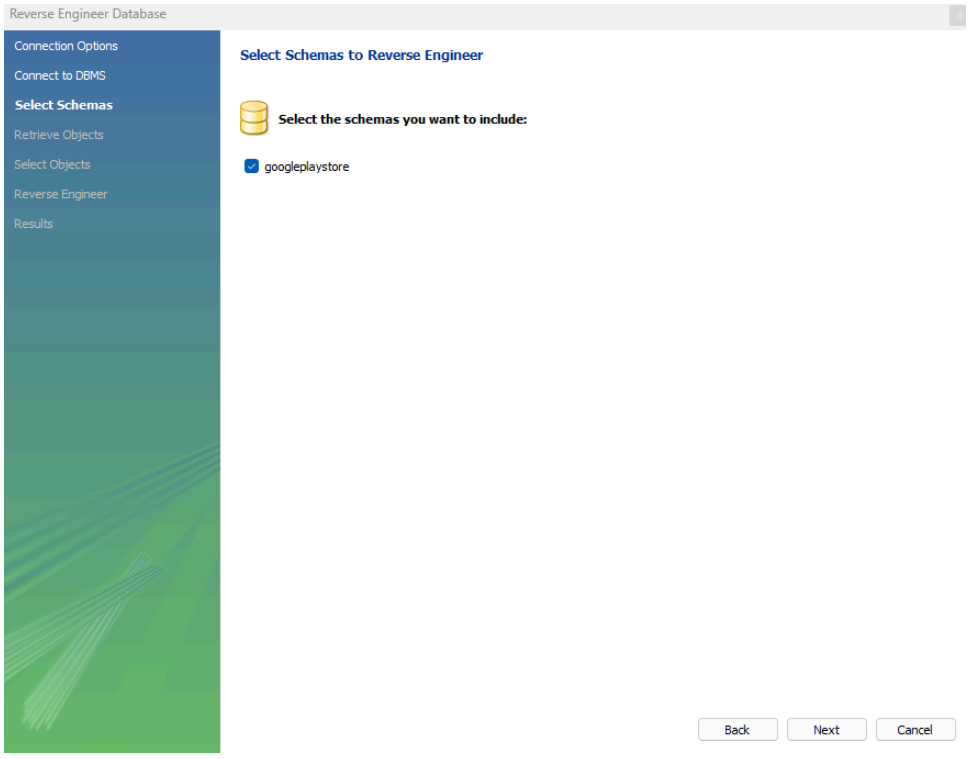
4. View ERD of created tables in the MySQL workbench
    a. Click on the Database tab and choose reverse engineer



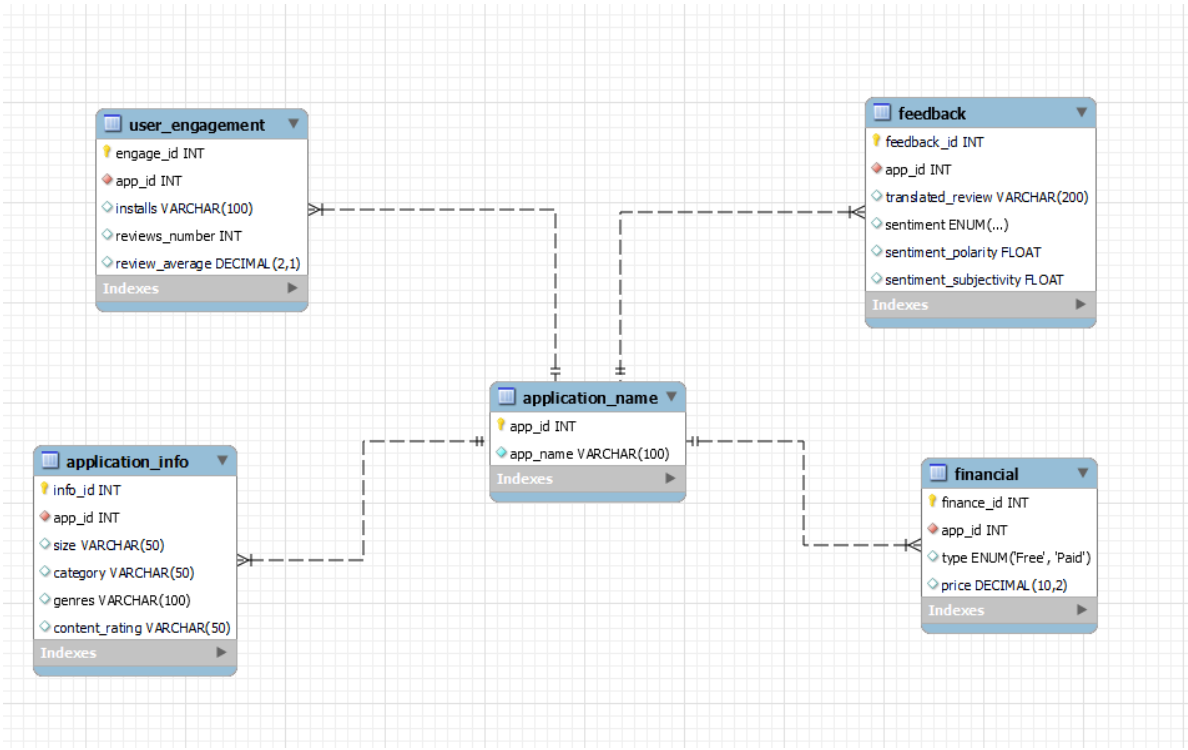
    b. Choose the Google_play_store EC2 instance method standard TCP/IP over SSH



c. Choose googleplaystore database and continue click Next



d. Check the ERD of all our tables

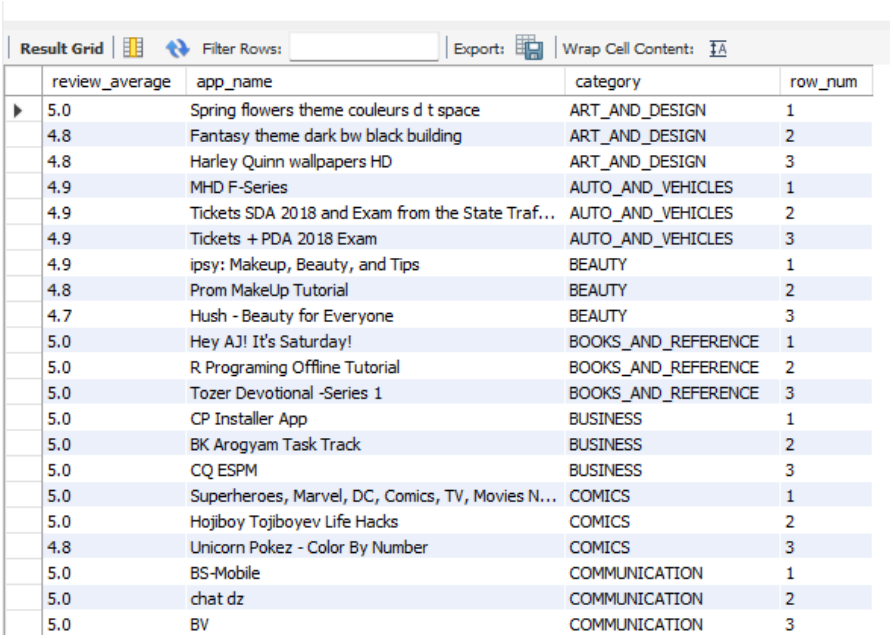


Queries to Demonstrate Purpose of Data

User-engagement

What are the top three applications for most review average of each category?

```
SELECT *
FROM (
SELECT u.review_average, a.app_name, i.category,
        ROW_NUMBER() OVER(PARTITION BY category ORDER BY
u.review_average DESC ) AS row_num
FROM user_engagement AS u
INNER JOIN application_name AS a ON u.app_id = a.app_id
INNER JOIN application_info AS i ON u.app_id = i.app_id ) AS ranked_apps WHERE
row_num <= 3;
```



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

review_average	app_name	category	row_num
5.0	Spring flowers theme couleurs d t space	ART_AND_DESIGN	1
4.8	Fantasy theme dark bw black building	ART_AND_DESIGN	2
4.8	Harley Quinn wallpapers HD	ART_AND_DESIGN	3
4.9	MHD F-Series	AUTO_AND_VEHICLES	1
4.9	Tickets SDA 2018 and Exam from the State Traf...	AUTO_AND_VEHICLES	2
4.9	Tickets + PDA 2018 Exam	AUTO_AND_VEHICLES	3
4.9	ipsy: Makeup, Beauty, and Tips	BEAUTY	1
4.8	Prom MakeUp Tutorial	BEAUTY	2
4.7	Hush - Beauty for Everyone	BEAUTY	3
5.0	Hey AJ! It's Saturday!	BOOKS_AND_REFERENCE	1
5.0	R Programing Offline Tutorial	BOOKS_AND_REFERENCE	2
5.0	Tozer Devotional -Series 1	BOOKS_AND_REFERENCE	3
5.0	CP Installer App	BUSINESS	1
5.0	BK Arogyam Task Track	BUSINESS	2
5.0	CQ ESPM	BUSINESS	3
5.0	Superheroes, Marvel, DC, Comics, TV, Movies N...	COMICS	1
5.0	Hojiboy Tojiboyev Life Hacks	COMICS	2
4.8	Unicorn Pokez - Color By Number	COMICS	3
5.0	BS-Mobile	COMMUNICATION	1
5.0	chat dz	COMMUNICATION	2
5.0	BV	COMMUNICATION	3

What are the top three applications for the most amount of review number of each category?

```
SELECT * FROM (
SELECT distinct(app_name), max_reviews, category,
ROW_NUMBER() OVER(PARTITION BY category, app_name ORDER BY
max_reviews DESC) AS name_rank,
ROW_NUMBER() OVER(PARTITION BY category ORDER BY max_reviews DESC )
AS row_num
FROM (
SELECT a.app_name, a.app_id, MAX(u.reviews_number) AS max_reviews
FROM user_engagement AS u
INNER JOIN application_name AS a ON u.app_id = a.app_id
GROUP BY a.app_id, a.app_name) AS ranked_apps
INNER JOIN application_info AS i ON ranked_apps.app_id = i.app_id ) AS
ranked_apps2 WHERE name_rank = 1 AND row_num <= 3;
```

app_name	max_reviews	category	name_rank	row_num
Weather & Clock Widget for Android	2371543	WEATHER	1	1
Textgram - write on photos	295237	ART_AND_DESIGN	1	2
Tickets + PDA 2018 Exam	197136	AUTO_AND_VEHICLES	1	2
ipsy: Makeup, Beauty, and Tips	49790	BEAUTY	1	2
LINE WEBTOON - Free Comics	1013944	COMICS	1	2
WhatsApp Messenger	69119316	COMMUNICATION	1	2
Zoosk Dating App: Meet Singles	516917	DATING	1	2
StubHub - Tickets to Sports, Concerts & Events	26089	EVENTS	1	2
Clash Royale	23136735	FAMILY	1	2
Bank of Brazil	1336246	FINANCE	1	2
Tastely	611136	FOOD_AND_DRINK	1	2
Clash of Clans	44893888	GAME	1	2
Calorie Counter - MyFitnessPal	1873523	HEALTH_AND_FITNESS	1	2
Muslim Pro - Prayer Times, Azan, Quran & Qibla	1133393	LIFESTYLE	1	2
Waze - GPS, Maps, Traffic Alerts & Live Navi...	7232629	MAPS_AND_NAVIGATION	1	2
My Calendar - Period Tracker	156410	MEDICAL	1	2
Twitter	11667403	NEWS_AND_MAGAZINES	1	2
Feed Baby - Baby Tracker	76795	PARENTING	1	2
CM Launcher 3D - Theme, Wallpapers, Efficient	6702776	PERSONALIZATION	1	2
Dream League Soccer 2018	9883806	SPORTS	1	2
Security Master - Antivirus, VPN, AppLock, B...	24900999	TOOLS	1	2
AccuWeather: Daily Forecast & Live Weather...	2053404	WEATHER	1	2
ibis Paint X	224399	ART_AND_DESIGN	1	3
AutoScout24 - used car finder	186648	AUTO_AND_VEHICLES	1	3
Best Hairstyles step by step	45452	BEAUTY	1	3
Bible	2440695	BOOKS_AND_REFERENCE	1	3

What are the top installs for each category?

```
SELECT * FROM(
SELECT a.app_name, i.category, u.installs,
DENSE_RANK() OVER(PARTITION BY i.category ORDER BY u.installs DESC)  AS
installs_rank
FROM user_engagement AS u
INNER JOIN application_name AS a ON u.app_id = a.app_id
INNER JOIN application_info AS i ON u.app_id = i.app_id) AS rank_install WHERE
installs_rank = 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

app_name	category	installs	installs_rank
Sketch - Draw & Paint	ART_AND_DESIGN	50000000	1
AutoScout24 - used car finder	AUTO_AND_VEHICLES	10000000	1
Android Auto - Maps, Media, Messaging & Voice	AUTO_AND_VEHICLES	10000000	1
Beauty Camera - Selfie Camera	BEAUTY	10000000	1
Google Play Books	BOOKS_AND_REFERENCE	1000000000	1
File Commander - File Manager/Explorer	BUSINESS	100000000	1
OfficeSuite : Free Office + PDF Editor	BUSINESS	100000000	1
OfficeSuite : Free Office + PDF Editor	BUSINESS	100000000	1
OfficeSuite : Free Office + PDF Editor	BUSINESS	100000000	1
OfficeSuite : Free Office + PDF Editor	BUSINESS	100000000	1
LINE WEBTOON - Free Comics	COMICS	10000000	1
LINE WEBTOON - Free Comics	COMICS	10000000	1
LINE WEBTOON - Free Comics	COMICS	10000000	1
LINE WEBTOON - Free Comics	COMICS	10000000	1
Hangouts	COMMUNICATION	1000000000	1
Messenger – Text and Video Chat for Free	COMMUNICATION	1000000000	1
Messenger – Text and Video Chat for Free	COMMUNICATION	1000000000	1
Google Chrome: Fast & Secure	COMMUNICATION	1000000000	1
Hangouts	COMMUNICATION	1000000000	1
Hangouts	COMMUNICATION	1000000000	1
Hangouts	COMMUNICATION	1000000000	1
Hangouts	COMMUNICATION	1000000000	1
Hangouts	COMMUNICATION	1000000000	1
Skype - free IM & video calls	COMMUNICATION	1000000000	1
Gmail	COMMUNICATION	1000000000	1
Hangouts	COMMUNICATION	1000000000	1

Financial

How many of the applications are paid versus free?

```
SELECT COUNT(app_id), type FROM financial GROUP BY type;
```

Result Grid | Filter Rows:

COUNT(app_id)	type
8275	Free
611	Paid

What is the installed number of the application with the highest price of the category?

```
SELECT * FROM(
SELECT a.app_name, i.category, f.price, u.installs,
        ROW_NUMBER() OVER(PARTITION BY i.category ORDER BY f.price DESC )
AS max_price
FROM financial AS f
INNER JOIN application_name AS a ON f.app_id = a.app_id
INNER JOIN application_info AS i ON f.app_id = i.app_id
INNER JOIN user_engagement AS u ON f.app_id = u.app_id
) AS high_price WHERE max_price = 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

app_name	category	price	installs	max_price
X Launcher Prime: With OS Style Theme & No Ads	ART_AND_DESIGN	1.99	1000	1
Ulysse Speedometer Pro	AUTO_AND_VEHICLES	1.99	50000	1
Female Daily	BEAUTY	0.00	100000	1
Officiel du SCRABBLE LAROUSSE	BOOKS_AND_REFERENCE	4.60	5000	1
I am rich	BUSINESS	399.99	10000	1
Manga-FR - Anime Vostfr	COMICS	0.00	10000	1
AG Contacts, Premium edition	COMMUNICATION	4.99	500	1
AMBW Dating App: Asian Men Black Women Int...	DATING	7.99	100	1
Fuzzy Numbers: Pre-K Number Foundation	EDUCATION	5.99	1000	1
My Talking Pet	ENTERTAINMENT	4.99	100000	1
EF Events	EVENTS	0.00	100	1
most expensive app (H)	FAMILY	399.99	100	1
I am rich	FINANCE	399.99	10000	1
Paprika Recipe Manager	FOOD_AND_DRINK	4.99	50000	1
The World Ends With You	GAME	17.99	10000	1
Fast Tract Diet	HEALTH_AND_FITNESS	7.99	1000	1
tinyCam Monitor FREE	HOUSE_AND_HOME	0.00	10000...	1
Cardboard	LIBRARIES_AND_DEMO	0.00	10000...	1
I'm Rich - Trump Edition	LIFESTYLE	400.00	10000	1
BackCountry Navigator TOPO GPS PRO	MAPS_AND_NAVIGATION	11.99	100000	1
Vargo Anesthesia Mega App	MEDICAL	79.99	1000	1
Égalité et Réconciliation	NEWS_AND_MAGAZINES	2.99	500	1
Baby Connect (activity log)	PARENTING	4.99	50000	1
BIG Launcher	PERSONALIZATION	9.99	10000	1
NewTek NDI	PHOTOGRAPHY	19.99	1000	1
ACCDB MDB DB Manager Pro - Editor for MS Acc...	PRODUCTIVITY	8.99	500	1

Feedback

Which top ten applications have the positive sentiment polarity from the users?

```
SELECT a.app_name, max(fb.sentiment_polarity) AS max_sentiment
FROM feedback AS fb
INNER JOIN application_name AS a ON fb.app_id = a.app_id
GROUP BY a.app_name
ORDER BY max_sentiment DESC LIMIT 10;
```

app_name	max_sentiment
10 Best Foods for You	1
11st	1
3D Live Neon Weed Launcher	1
1LINE – One Line with One Touch	1
2018Emoji Keyboard 😍 Emoticons Lite -sticker&gif	1
365Scores - Live Scores	1
2Date Dating App, Love and matching	1
2GIS: directory & navigator	1
2RedBeans	1
2ndLine - Second Phone Number	1

How much sentiment is positive vs negative from the users for each application?

```
SELECT a.app_name,
SUM(CASE WHEN fb.sentiment = 'Positive' THEN 1 ELSE 0 END) AS Positive,
SUM(CASE WHEN fb.sentiment = 'Negative' THEN 1 ELSE 0 END) AS Negative
FROM feedback AS fb
INNER JOIN application_name AS a ON fb.app_id = a.app_id
GROUP BY a.app_name;
```

app_name	Positive	Negative
10 Best Foods for You	79	5
11st	23	7
1800 Contacts - Lens Store	32	3
1LINE – One Line with One Touch	27	8
2018Emoji Keyboard 😂 Emoticons Lite -sticker&gif	22	1
21-Day Meditation Experience	35	5
2Date Dating App, Love and matching	26	7
2GIS: directory & navigator	23	6
2RedBeans	28	2
2ndLine - Second Phone Number	17	7
30 Day Fitness Challenge - Workout at Home	27	2
365Scores - Live Scores	5	0
3D Live Neon Weed Launcher	2	0
4 in a Row	17	3
4K Wallpapers and Ultra HD Backgrounds	7	2
7 Cups: Anxiety & Stress Chat	7	1
7 Day Food Journal Challenge	9	0
7 Minute Workout	10	1
7 Weeks - Habit & Goal Tracker	10	4
8 Ball Pool	48	46
850 Sports News Digest	38	1
8fit Workouts & Meal Planner	82	11
95Live -SG#1 Live Streaming App	16	10
A Call From Santa Claus!	20	14
A Word A Day	3	0
A&E - Watch Full Episodes of TV Shows	20	3

Discussion

In this project, I have successfully created the normalised data based on the two-table dataset that contained the information of the applications on the Google Play Store platform. The purpose is to create a database that a data analyst can work with to create a dashboard that includes financial, user-engagement, and feedback aspects. What I wish I could further improve on this project is I wish to find the column that includes the date for when data is inserted and the versions of the applications. By including a date column, we can investigate more into trends of finance, user engagement, and feedback over time. What prevents me from including the date is that the last updated column of the raw dataset contains all the same date, even though the review numbers increase. In the future, I would like to improve by spending more time on initial data exploration and data selection before beginning the project. I was also attempting to import a MySQL database into the Power BI desktop, but there were too many complications and not enough time to create a sample dashboard for me. Overall, I created the normalised database as I intended to design in the ERD, and it is usable in the real world for querying for data analysis.

References

"How Google Play Works." *Play.google*, 2025, play.google/howplayworks/. Accessed 5 May 2025.

"IPhone vs. Android User & Revenue Statistics (2025)." *Backlinko*, 13 Mar. 2024, backlinko.com/iphone-vs-android-statistics. Accessed 5 May 2025.