APP MANAGEMENT SYSTEM

Our Team :

Rishabh Jain - 002769493

Charan Patnaik - 002762551

Jared Videlefsky - 001966442

Felicia Sequeira - 002744669

CONTRIBUTION

We all contributed equally!

Rishabh Jain-25%
Charan Patnaik-25%
Jared Videlefsky-25%
Felicia Sequeira-25%

OVERVIEW

The Application Management System makes it easy for end users, developers, and businesses to collaborate and fulfill their particular responsibilities without difficulty as the optimal platform for managing applications. It is a reliable and effective database.

PROJECT OBJECTIVES

01

Store

The project's central storage area is where all of the apps will be kept. The store should be well-structured and arranged to give users a seamless and easy browsing experience. Based on application category, popularity, and ranking, the user ought to be able to locate whatever application they are looking for.

02

User

A user should be able to view the programs that are installed on their devices, their current versions, and their in-app subscriptions.

03

Developers

·The ability for developers to create apps, view the most recent metadata for those applications, and update versions of those applications is essential. App versions and the adverts linked to them should be simple for developers to track.

CONTENT

TABLES & USERS

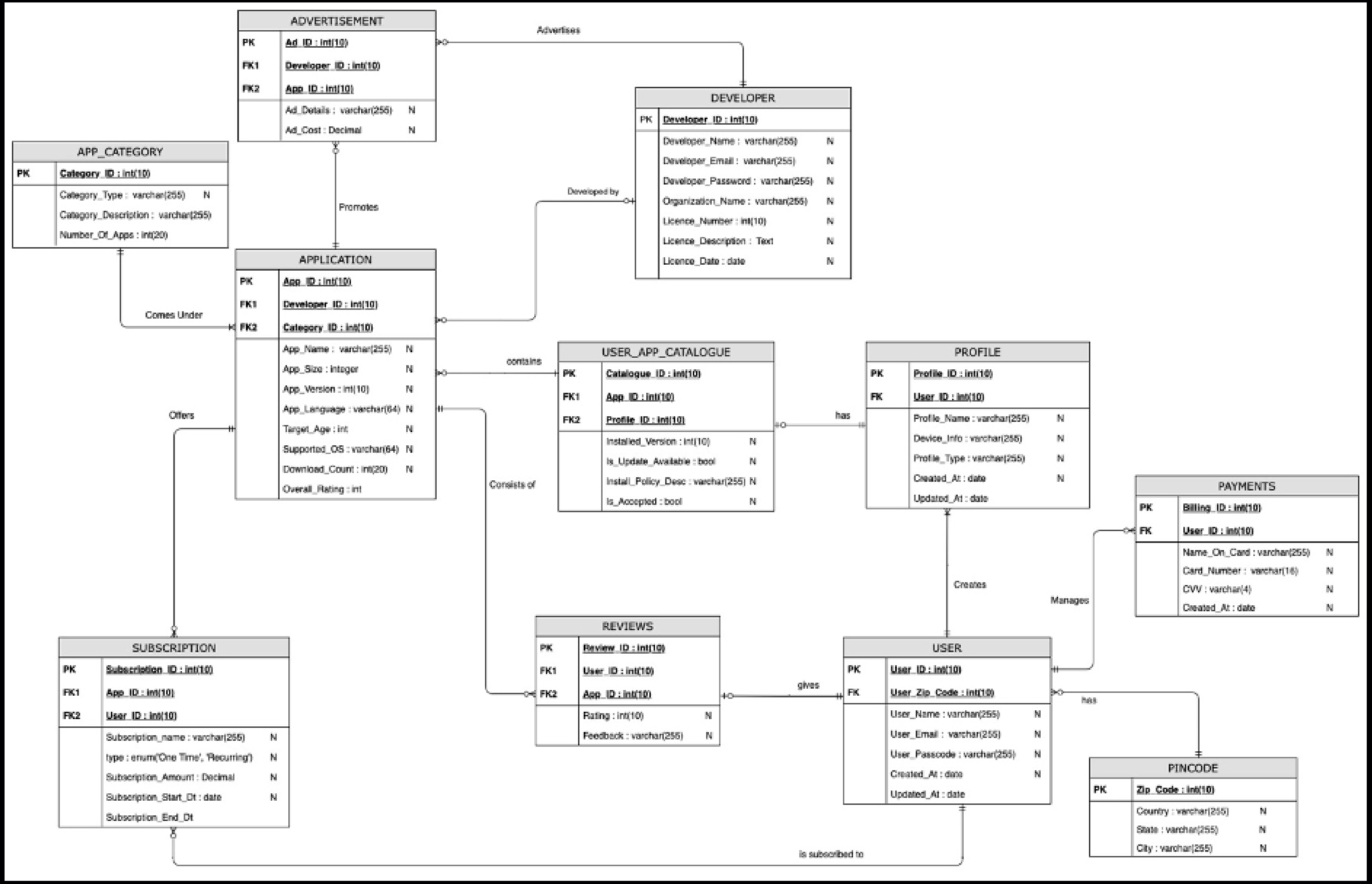
Tables

- USER_INFO
- PINCODE
- PAYMENTS
- DEVELOPER
- APP_CATEGORY
- APPLICATION
- PROFILE
- REVIEWS
- USER_APP_CATALOGUE
- ADVERTISEMENT
- SUBSCRIPTION

Users

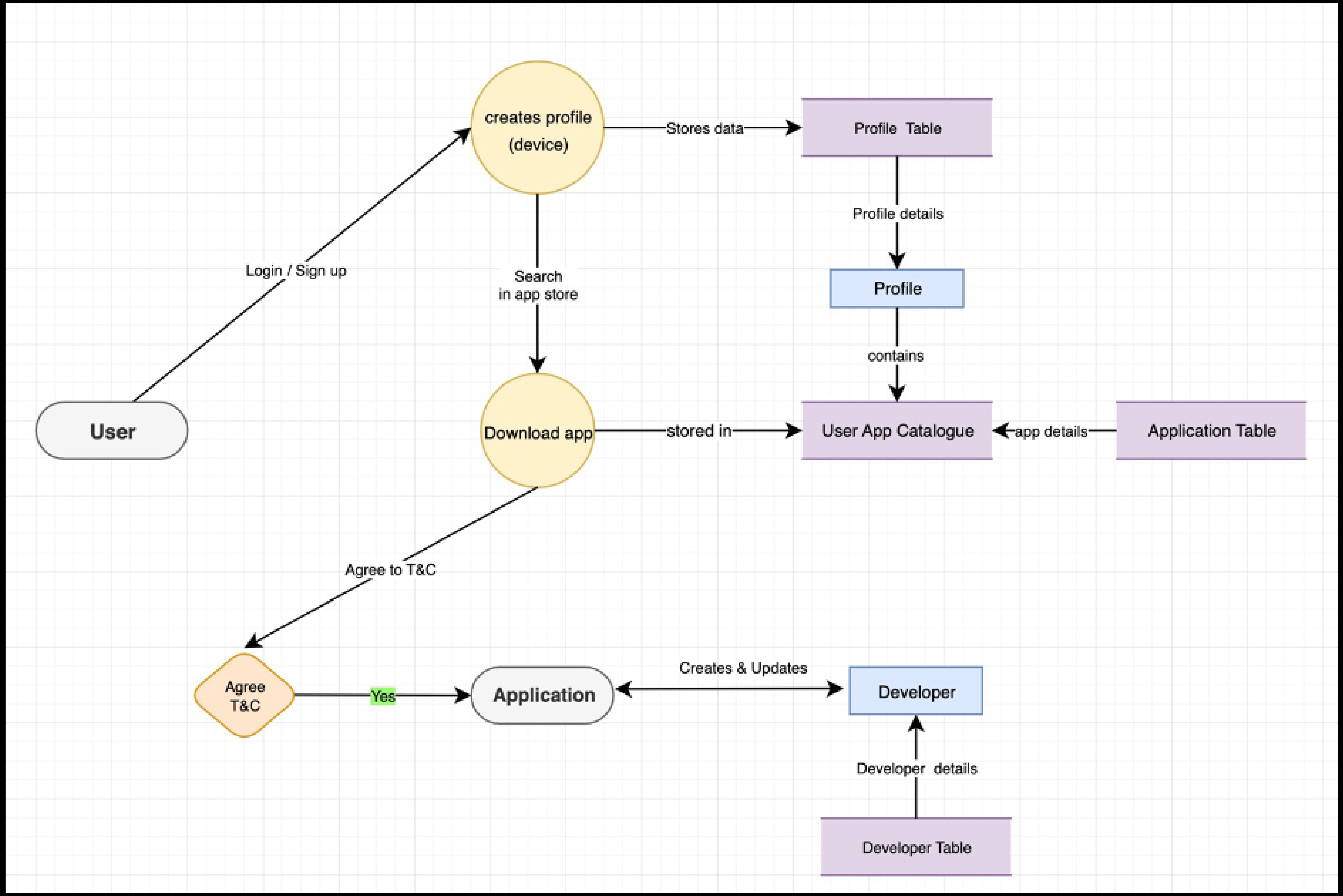
- DB_ADMIN
- DEVELOPER_MANAGER
- USER

ER - DIAGRAM


ADVERTISEMENT
PK Ad_ID : int(10)
FK1 Developer_ID : int(10)
FK2 App_ID : int(10)
Ad_Details : varchar(255) N
Ad_Cost : Decimal N
Advertises
DEVELOPER
PK Developer_ID : int(10)
Developer_Name : varchar(255) N
Developer_Email : varchar(255) N
Developer_Password : varchar(255) N
Organization_Name : varchar(255) N
Licence_Number : int(10) N
Licence_Description : Text N
Licence_Date : date N
Developed by
Promotes
APP_CATEGORY
PK Category_ID : int(10)
Category_Type : varchar(255) N
Category_Description : varchar(255)
Number_Of_Apps : int(20)
Comes Under
APPLICATION
PK App_ID : int(10)
FK1 Developer_ID : int(10)
FK2 Category_ID : int(10)
App_Name : varchar(255) N
App_Size : integer N
App_Version : int(10) N
App_Language : varchar(64) N
Target_Age : int N
Supported_OS : varchar(64) N
Download_Count : int(20) N
Overall_Rating : int
contains
USER_APP_CATALOGUE
PK Catalogue_ID : int(10)
FK1 App_ID : int(10)
FK2 Profile_ID : int(10)
Installed_Version : int(10) N
Is_Update_Available : bool N
Install_Policy_Desc : varchar(255) N
Is_Accepted : bool N
has
PROFILE
PK Profile_ID : int(10)
FK User_ID : int(10)
Profile_Name : varchar(255) N
Device_Info : varchar(255) N
Profile_Type : varchar(255) N
Created_At : date N
Updated_At : date
Offers
Consists of
Creates
Manages
PAYMENTS
PK Billing_ID : int(10)
FK User_ID : int(10)
Name_On_Card : varchar(255) N
Card_Number : varchar(16) N
CVV : varchar(4) N
Created_At : date N
SUBSCRIPTION
PK Subscription_ID : int(10)
FK1 App_ID : int(10)
FK2 User_ID : int(10)
Subscription_name : varchar(255) N
type : enum('One Time', 'Recurring') N
Subscription_Amount : Decimal N
Subscription_Start_Dt : date N
Subscription_End_Dt
REVIEWS
PK Review_ID : int(10)
FK1 User_ID : int(10)
FK2 App_ID : int(10)
Rating : int(10) N
Feedback : varchar(255) N
gives
USER
PK User_ID : int(10)
FK User_Zip_Code : int(10)
User_Name : varchar(255) N
User_Email : varchar(255) N
User_Passcode : varchar(255) N
Created_At : date N
Updated_At : date
has
PINCODE
PK Zip_Code : int(10)
Country : varchar(255) N
State : varchar(255) N
City : varchar(255) N
is subscribed to

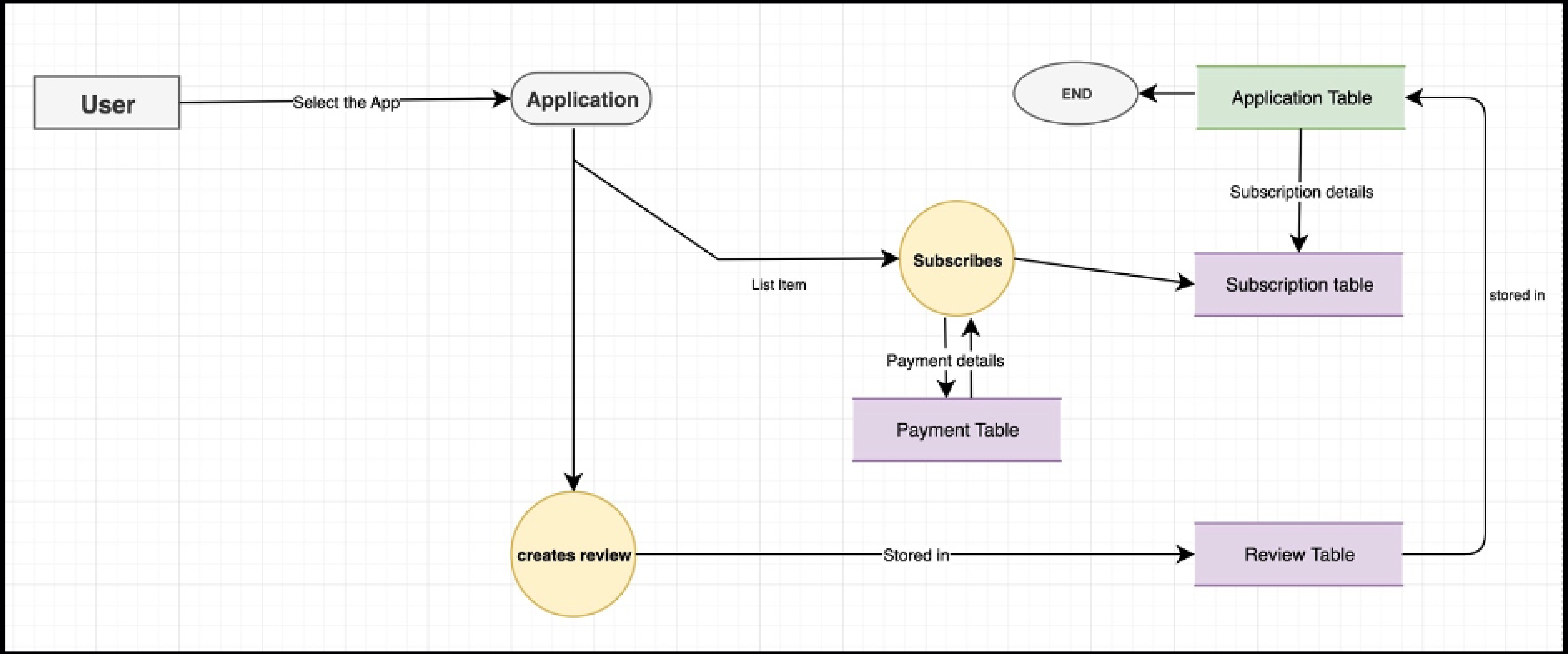
DATA FLOW DIAGRAM -1

User Login & App install Module.



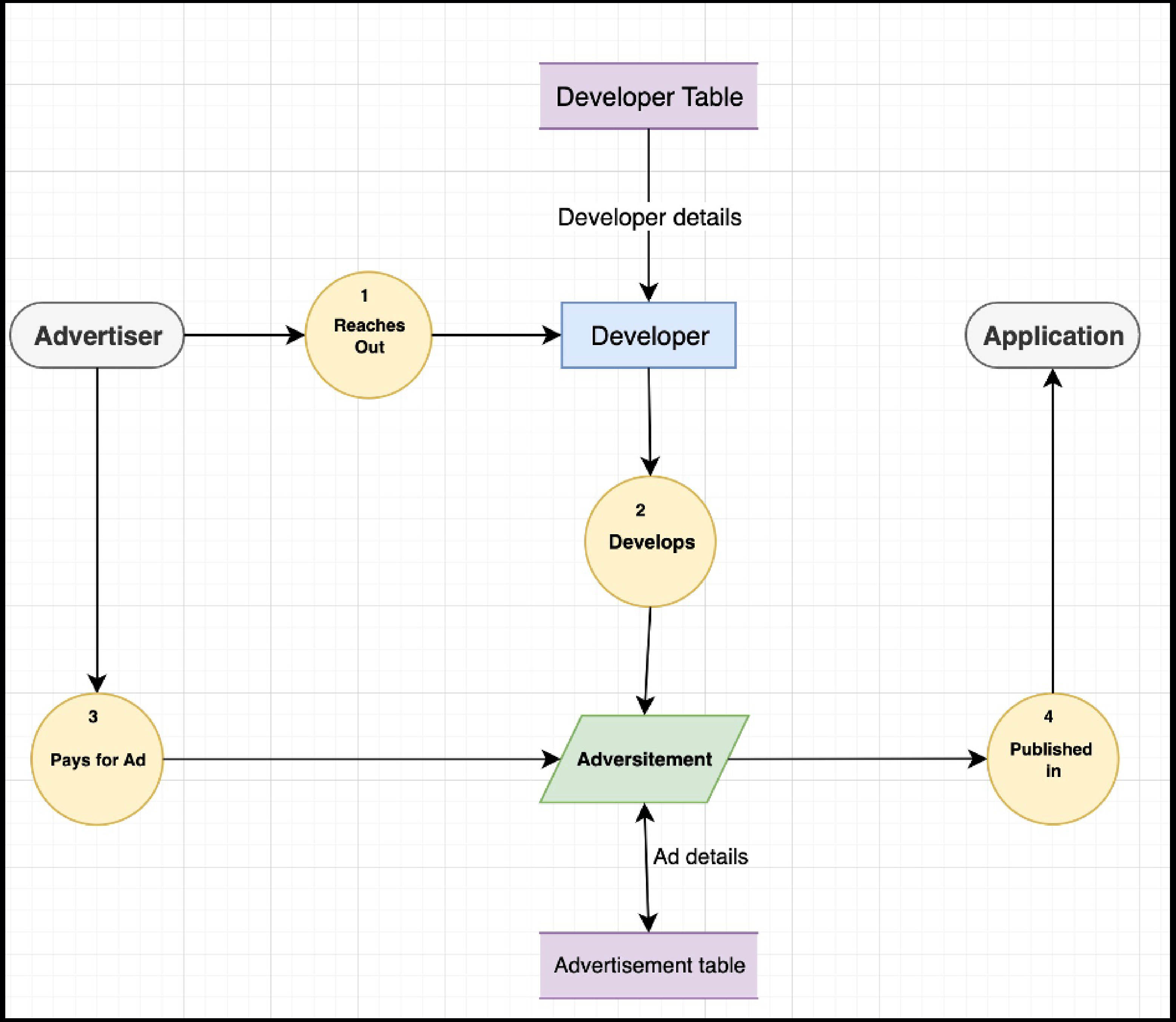
DATA FLOW DIAGRAM -2

User Subscription & Review Module



DATA FLOW DIAGRAM -3

Advertisement Module



SECURITY RULES

DB_ADMIN
Complete Access -> All Tables

- The DB_ADMIN has complete access to all the tables, He can perform everything on every table

DEVELOPER_MANAGER
SELECT -> REVIEWS
APP_CATEGORY

SELECT, INSERT, UPDATE -> APPLICATION
ADVERTISEMENT
SUBSCRIPTION

- The DEVELOPER_MANAGER can select data from the REVIEWS and the APP_CATEGORY table and can perform Inserts and Updates on the APPLICATION, ADVERTISEMENT and SUBSCRIPTIONS table.

USER
SELECT -> USER_APP_CATALOGUE
SUBSCRIPTION

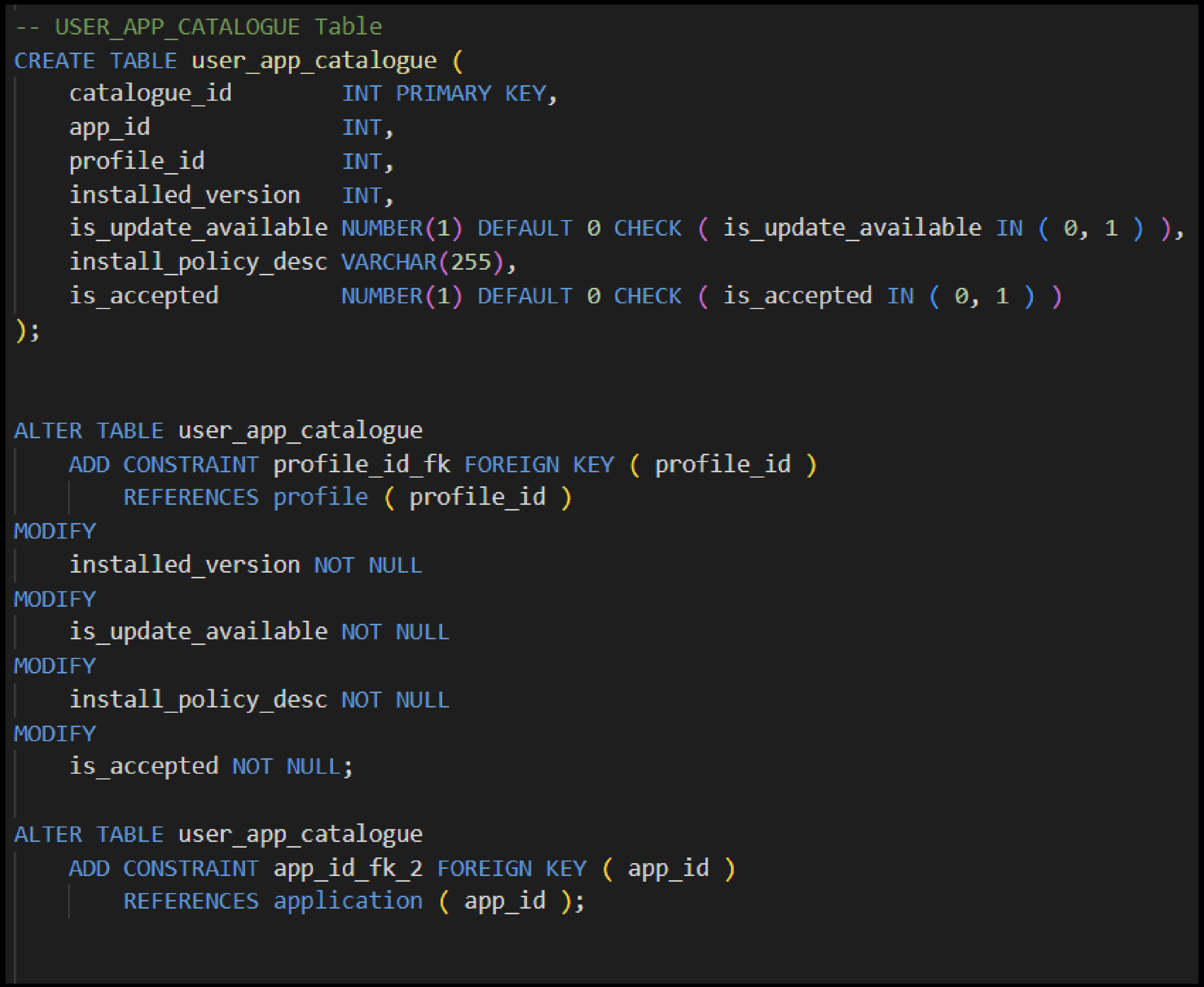
SELECT, INSERT, UPDATE -> USER
PAYMENTS
PINCODE
PROFILE
REVIEWS

- The USER can select from the USER_APP_CATALOGUE and can perform inserts and updates on USER, PAYMENTS, PINCODE, PROFILE and REVIEWS.

TABLES CREATION

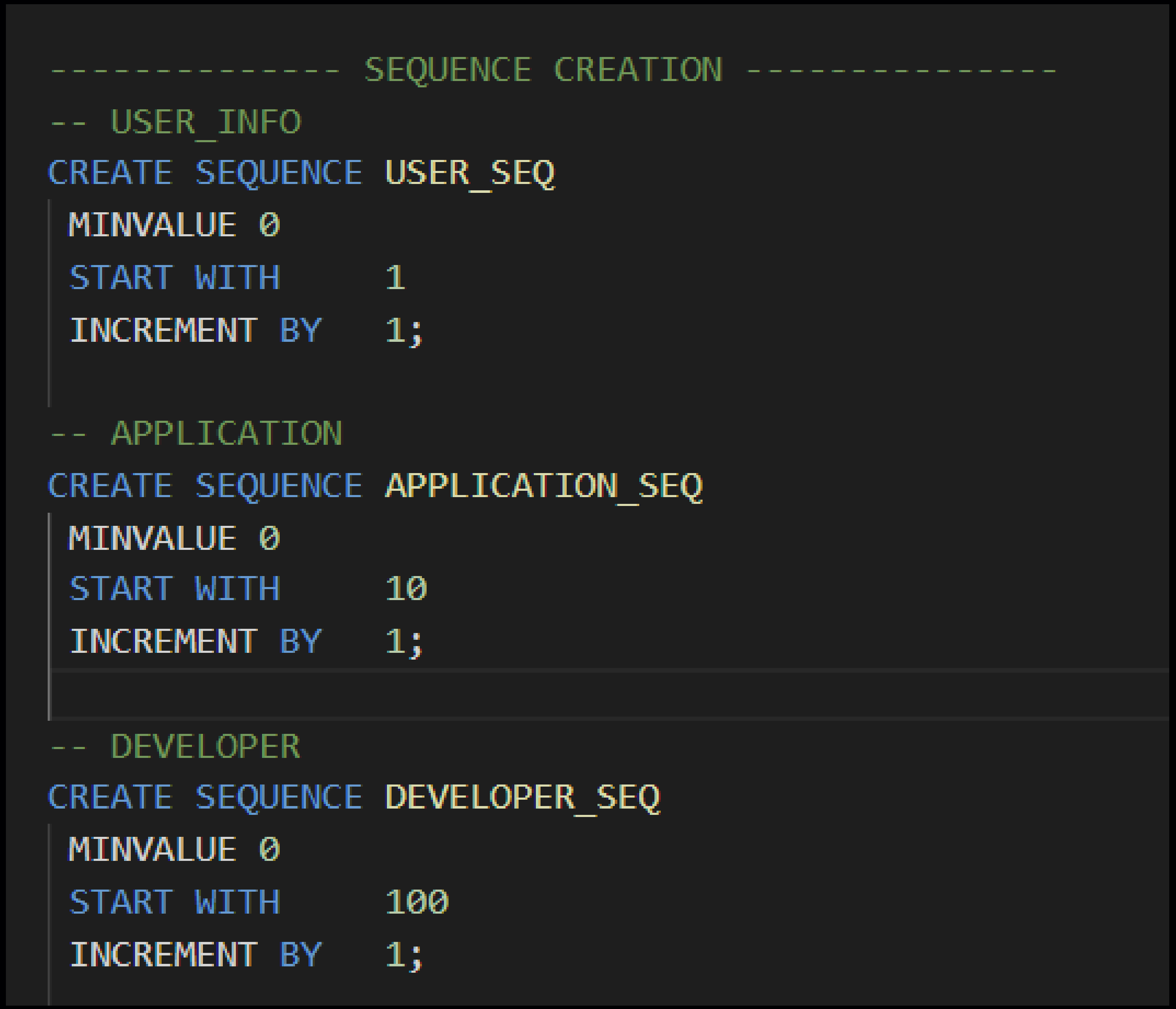
Table List:

1. Application
2. Advertisement
3. App_Category
4. License
5. Developer
6. User_App_Catalogue
7. User
8. Subscription
9. Install_Policy
10. Profile
11. Payments
12. Reviews



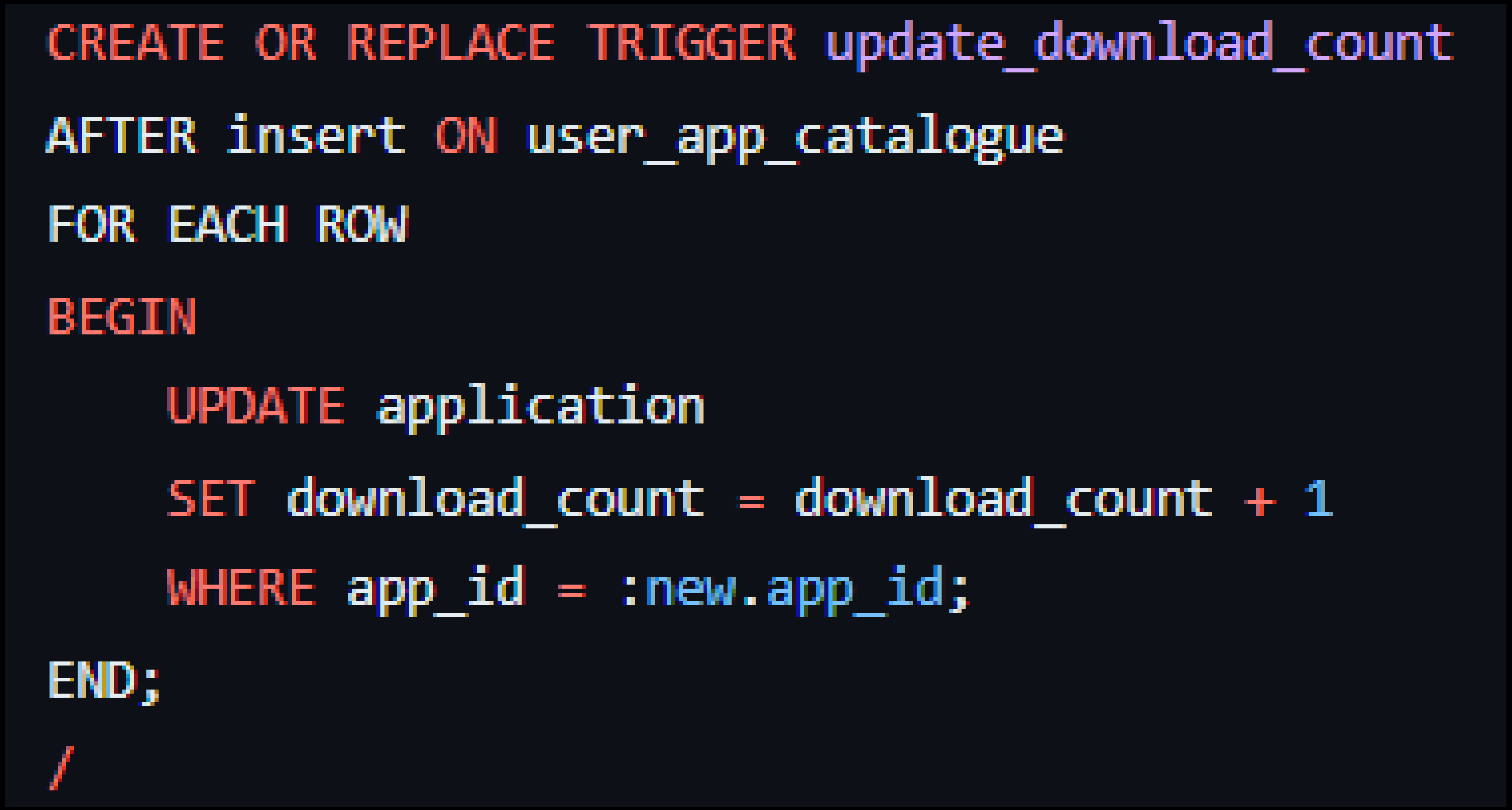
```
-- USER_APP_CATALOGUE Table
CREATE TABLE user_app_catalogue (
    catalogue_id        INT PRIMARY KEY,
    app_id              INT,
    profile_id          INT,
    installed_version   INT,
    is_update_available NUMBER(1) DEFAULT 0 CHECK ( is_update_available IN ( 0, 1 ) ),
    install_policy_desc VARCHAR(255),
    is_accepted         NUMBER(1) DEFAULT 0 CHECK ( is_accepted IN ( 0, 1 ) )
);


ALTER TABLE user_app_catalogue
    ADD CONSTRAINT profile_id_fk FOREIGN KEY ( profile_id )
        REFERENCES profile ( profile_id )
MODIFY
    installed_version NOT NULL
MODIFY
    is_update_available NOT NULL
MODIFY
    install_policy_desc NOT NULL
MODIFY
    is_accepted NOT NULL;

ALTER TABLE user_app_catalogue
    ADD CONSTRAINT app_id_fk_2 FOREIGN KEY ( app_id )
        REFERENCES application ( app_id );
```

SEQUENCE CREATION



```
-------------- SEQUENCE CREATION ---------------
-- USER_INFO
CREATE SEQUENCE USER_SEQ
 MINVALUE 0
 START WITH     1
 INCREMENT BY   1;

-- APPLICATION
CREATE SEQUENCE APPLICATION_SEQ
 MINVALUE 0
 START WITH     10
 INCREMENT BY   1;

-- DEVELOPER
CREATE SEQUENCE DEVELOPER_SEQ
 MINVALUE 0
 START WITH     100
 INCREMENT BY   1;
```

TRIGGERS



```
CREATE OR REPLACE TRIGGER update_download_count
AFTER insert ON user_app_catalogue
FOR EACH ROW
BEGIN
    UPDATE application
    SET download_count = download_count + 1
    WHERE app_id = :new.app_id;
END;
/
```

STORED PROCEDURES

ADMIN_PACKAGE

- PROCEDURE delete_advertisement
- PROCEDURE publish_ad
- PROCEDURE get_advertisements_by_app_id
- PROCEDURE update_advertisement
- PROCEDURE add_app_category
- PROCEDURE update_category_description
- PROCEDURE add_new_pincode
- PROCEDURE sign_up_developer
- PROCEDURE update_developer

STORED PROCEDURES

DEVELOPER_PACKAGE

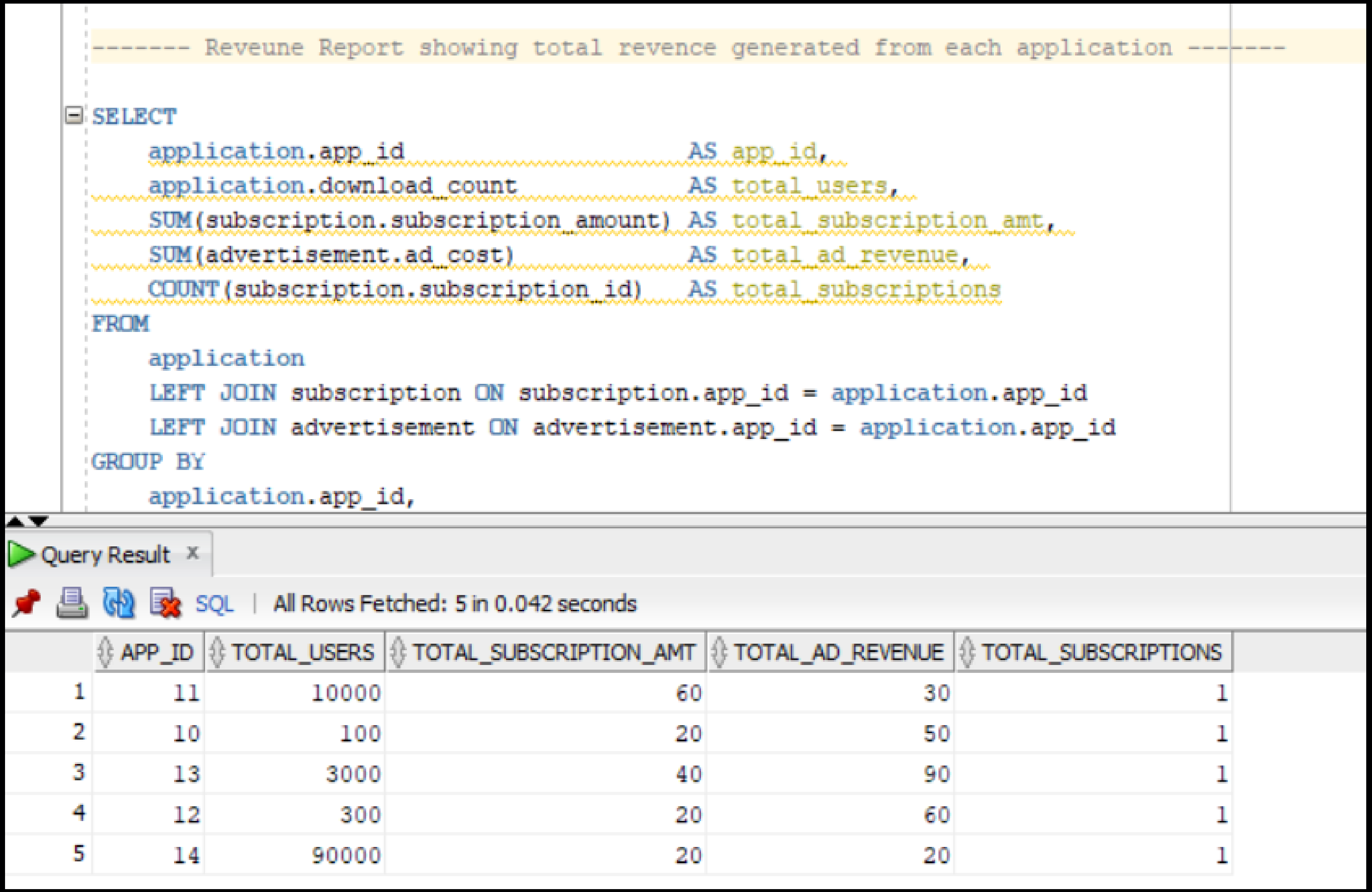
- PROCEDURE publish_application
- PROCEDURE get_app_categories
- PROCEDURE get_advertisements_by_app_id

STORED PROCEDURES

user_manager_pkg

- PROCEDURE select_user_info
- PROCEDURE insert_user_info_pkg
- PROCEDURE create_profile_pkg
- PROCEDURE update_user_info
- PROCEDURE insert_user_app_catalog_pkg
- PROCEDURE get_apps_for_profile
- PROCEDURE post_review
- PROCEDURE update_review
- PROCEDURE add_billing_info
- PROCEDURE get_user_payments
- PROCEDURE buy_subscription
- PROCEDURE get_subscriptions

REPORTS



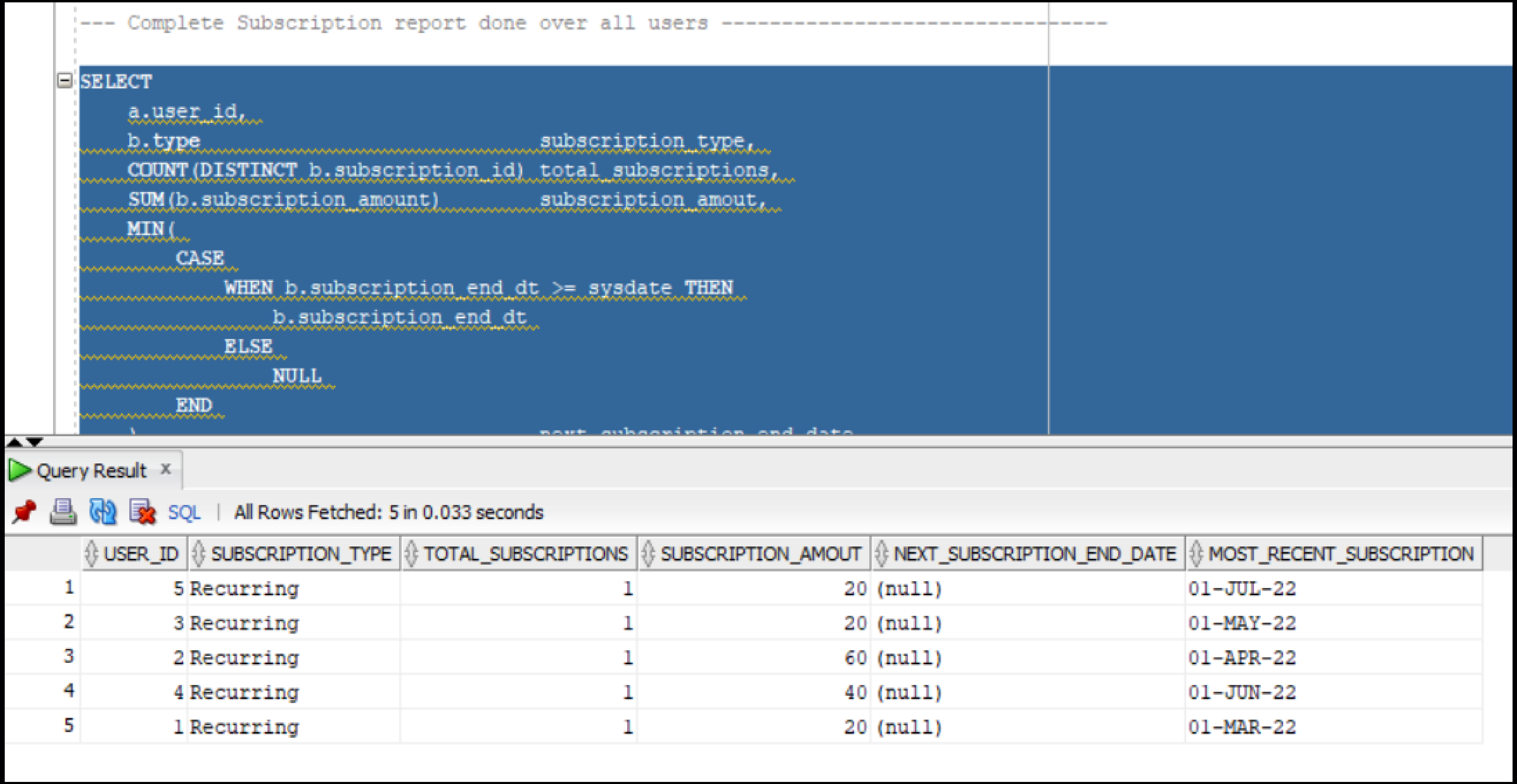
```
------- Reveune Report showing total revence generated from each application -------

SELECT
    application.app_id                        AS app_id,
    application.download_count                AS total_users,
    SUM(subscription.subscription_amount) AS total_subscription_amt,
    SUM(advertisement.ad_cost)                AS total_ad_revenue,
    COUNT(subscription.subscription_id)     AS total_subscriptions
FROM
    application
    LEFT JOIN subscription ON subscription.app_id = application.app_id
    LEFT JOIN advertisement ON advertisement.app_id = application.app_id
GROUP BY
    application.app_id,
```

Query Result ×

SQL | All Rows Fetched: 5 in 0.042 seconds

	APP_ID	TOTAL_USERS	TOTAL_SUBSCRIPTION_AMT	TOTAL_AD_REVENUE	TOTAL_SUBSCRIPTIONS
1	11	10000	60	30	1
2	10	100	20	50	1
3	13	3000	40	90	1
4	12	300	20	60	1
5	14	90000	20	20	1

REPORTS



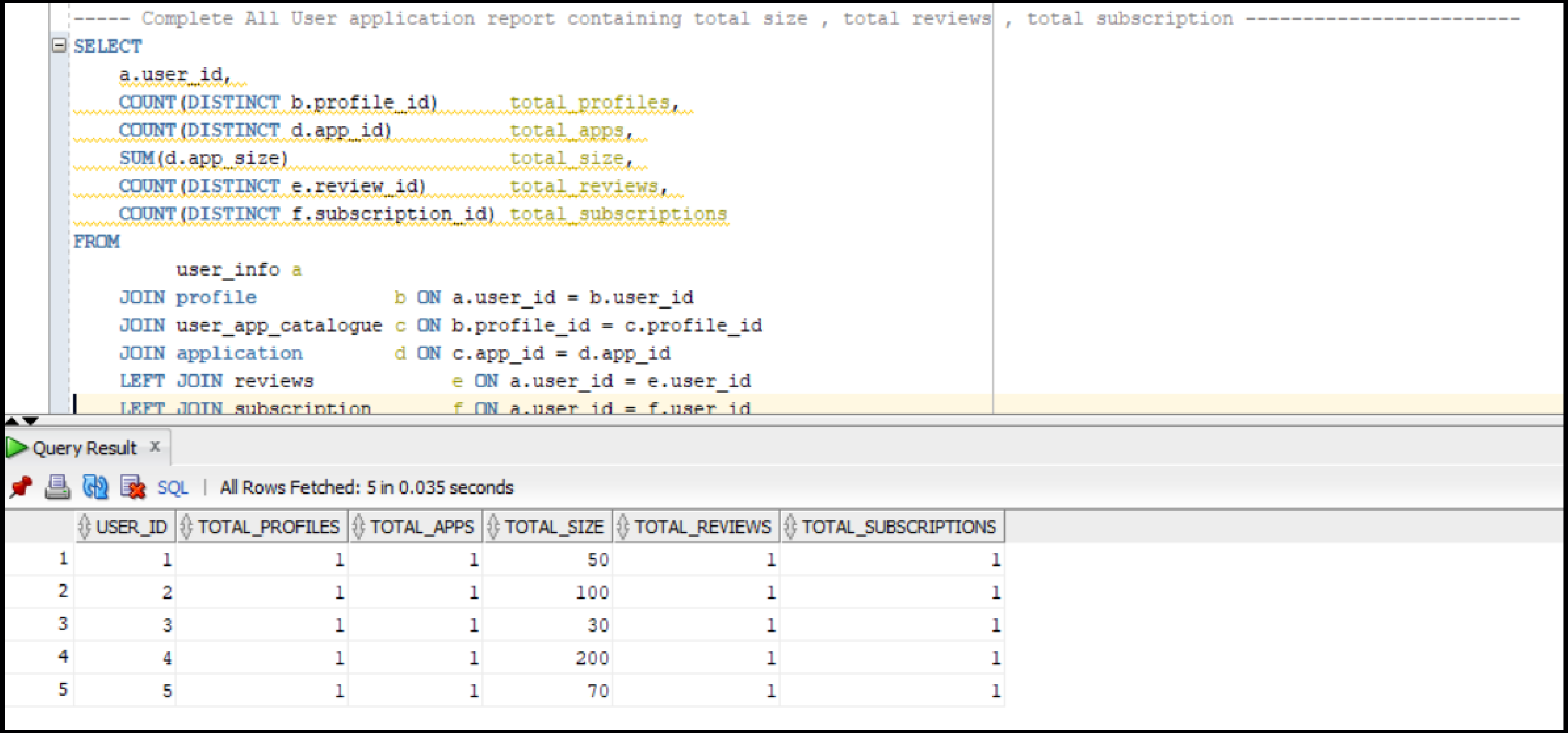
```
--- Complete Subscription report done over all users ---------------------------------

SELECT
    a.user_id,
    b.type                                  subscription_type,
    COUNT(DISTINCT b.subscription_id)  total_subscriptions,
    SUM(b.subscription_amount)             subscription_amout,
    MIN(
        CASE
            WHEN b.subscription_end_dt >= sysdate THEN
                b.subscription_end_dt
            ELSE
                NULL
        END
    )                                       next_subscription_end_date
```

Query Result ×

SQL | All Rows Fetched: 5 in 0.033 seconds

	USER_ID	SUBSCRIPTION_TYPE	TOTAL_SUBSCRIPTIONS	SUBSCRIPTION_AMOUT	NEXT_SUBSCRIPTION_END_DATE	MOST_RECENT_SUBSCRIPTION
1	5	Recurring	1	20	(null)	01-JUL-22
2	3	Recurring	1	20	(null)	01-MAY-22
3	2	Recurring	1	60	(null)	01-APR-22
4	4	Recurring	1	40	(null)	01-JUN-22
5	1	Recurring	1	20	(null)	01-MAR-22

REPORTS



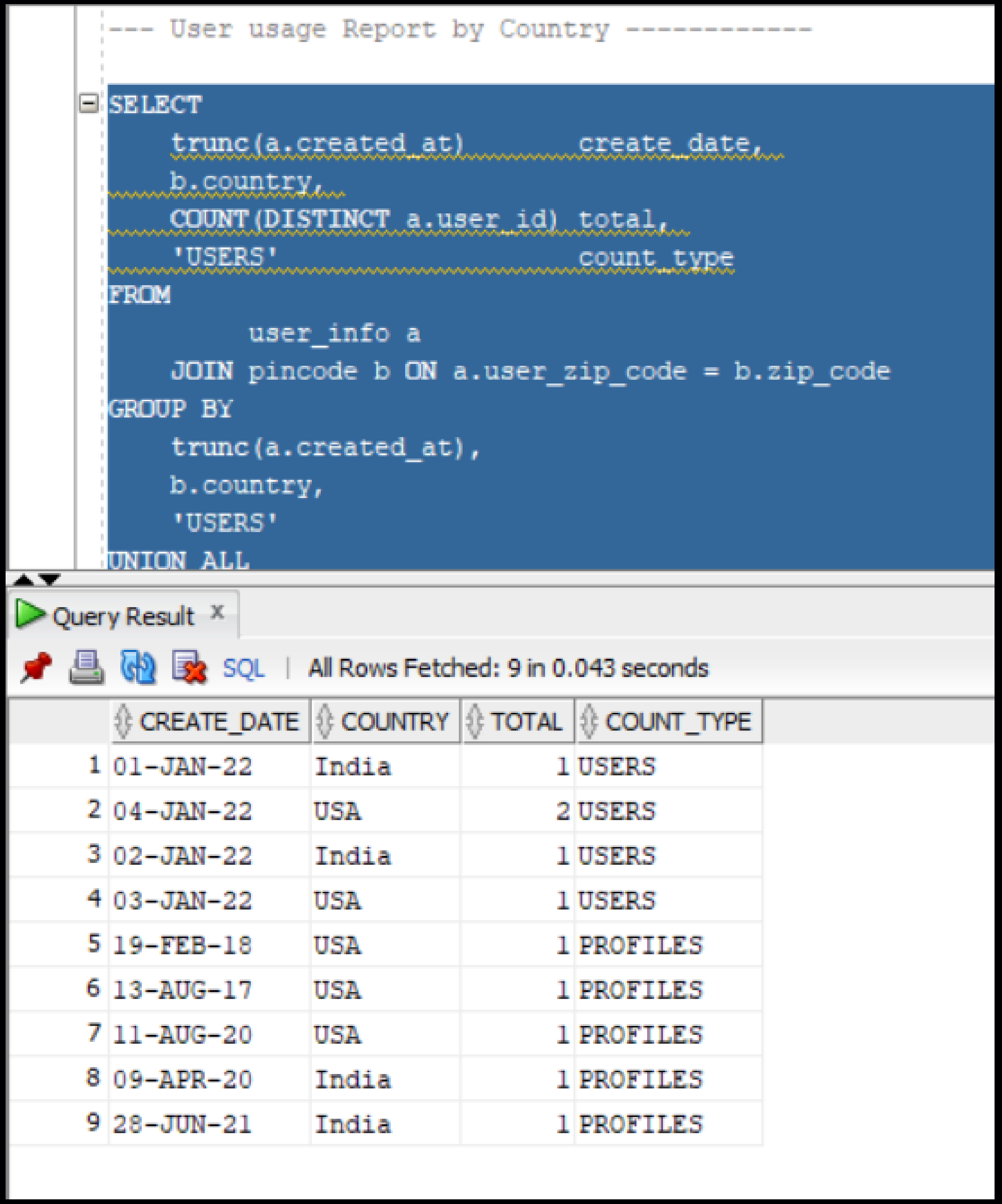
```
----- Complete All User application report containing total size , total reviews , total subscription -----------------------
SELECT
    a.user_id,
    COUNT(DISTINCT b.profile_id)      total_profiles,
    COUNT(DISTINCT d.app_id)          total_apps,
    SUM(d.app_size)                   total_size,
    COUNT(DISTINCT e.review_id)       total_reviews,
    COUNT(DISTINCT f.subscription_id) total_subscriptions
FROM
         user_info a
    JOIN profile              b ON a.user_id = b.user_id
    JOIN user_app_catalogue c ON b.profile_id = c.profile_id
    JOIN application          d ON c.app_id = d.app_id
    LEFT JOIN reviews              e ON a.user_id = e.user_id
    LEFT JOIN subscription         f ON a.user_id = f.user_id
```

Query Result

All Rows Fetched: 5 in 0.035 seconds

	USER_ID	TOTAL_PROFILES	TOTAL_APPS	TOTAL_SIZE	TOTAL_REVIEWS	TOTAL_SUBSCRIPTIONS
1	1	1	1	50	1	1
2	2	1	1	100	1	1
3	3	1	1	30	1	1
4	4	1	1	200	1	1
5	5	1	1	70	1	1

REPORTS



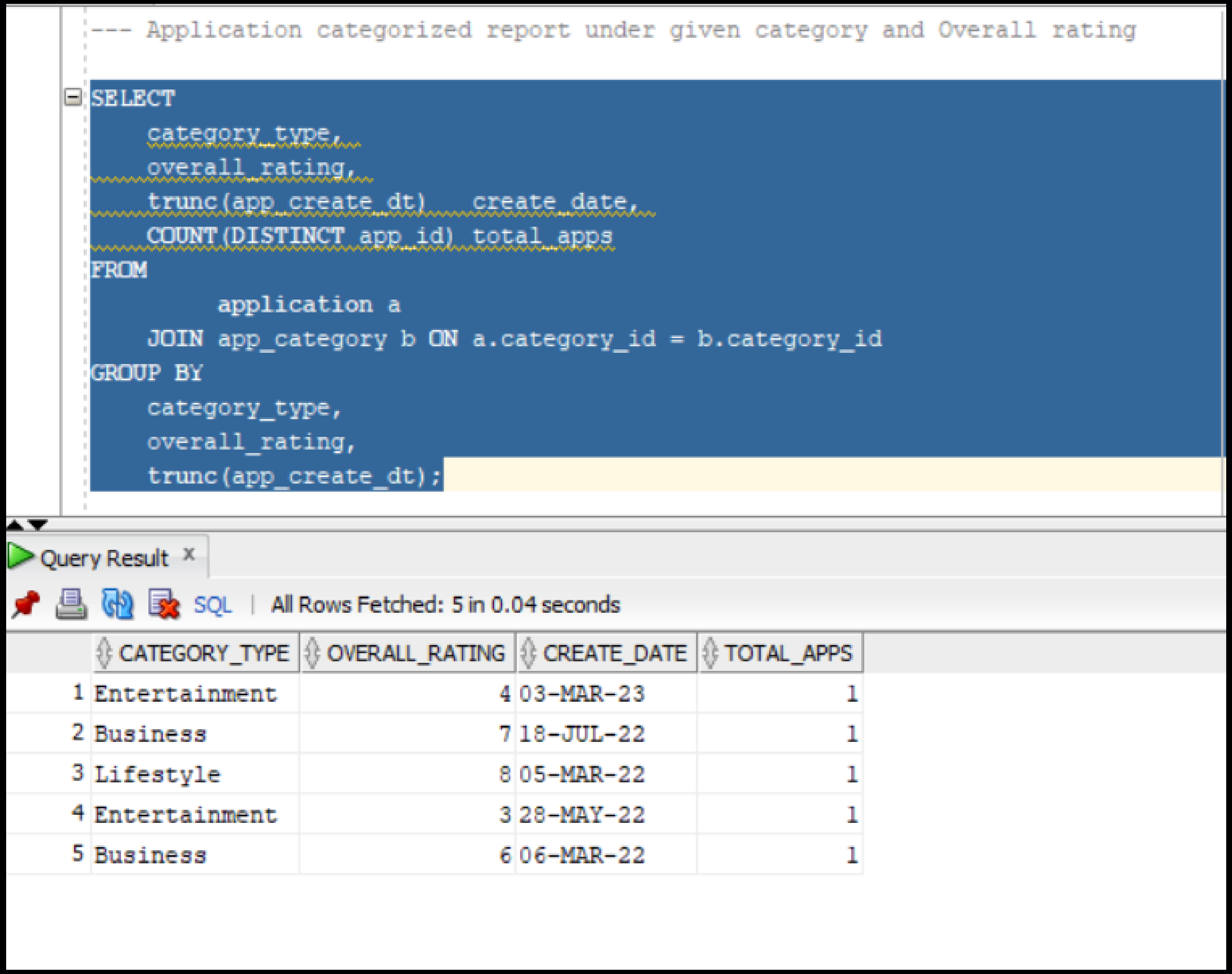
```
--- User usage Report by Country ------------

SELECT
    trunc(a.created_at)        create_date,
    b.country,
    COUNT(DISTINCT a.user_id) total,
    'USERS'                    count_type
FROM
         user_info a
    JOIN pincode b ON a.user_zip_code = b.zip_code
GROUP BY
    trunc(a.created_at),
    b.country,
    'USERS'
UNION ALL
```

Query Result ×

SQL | All Rows Fetched: 9 in 0.043 seconds

	CREATE_DATE	COUNTRY	TOTAL	COUNT_TYPE
1	01-JAN-22	India	1	USERS
2	04-JAN-22	USA	2	USERS
3	02-JAN-22	India	1	USERS
4	03-JAN-22	USA	1	USERS
5	19-FEB-18	USA	1	PROFILES
6	13-AUG-17	USA	1	PROFILES
7	11-AUG-20	USA	1	PROFILES
8	09-APR-20	India	1	PROFILES
9	28-JUN-21	India	1	PROFILES

REPORTS



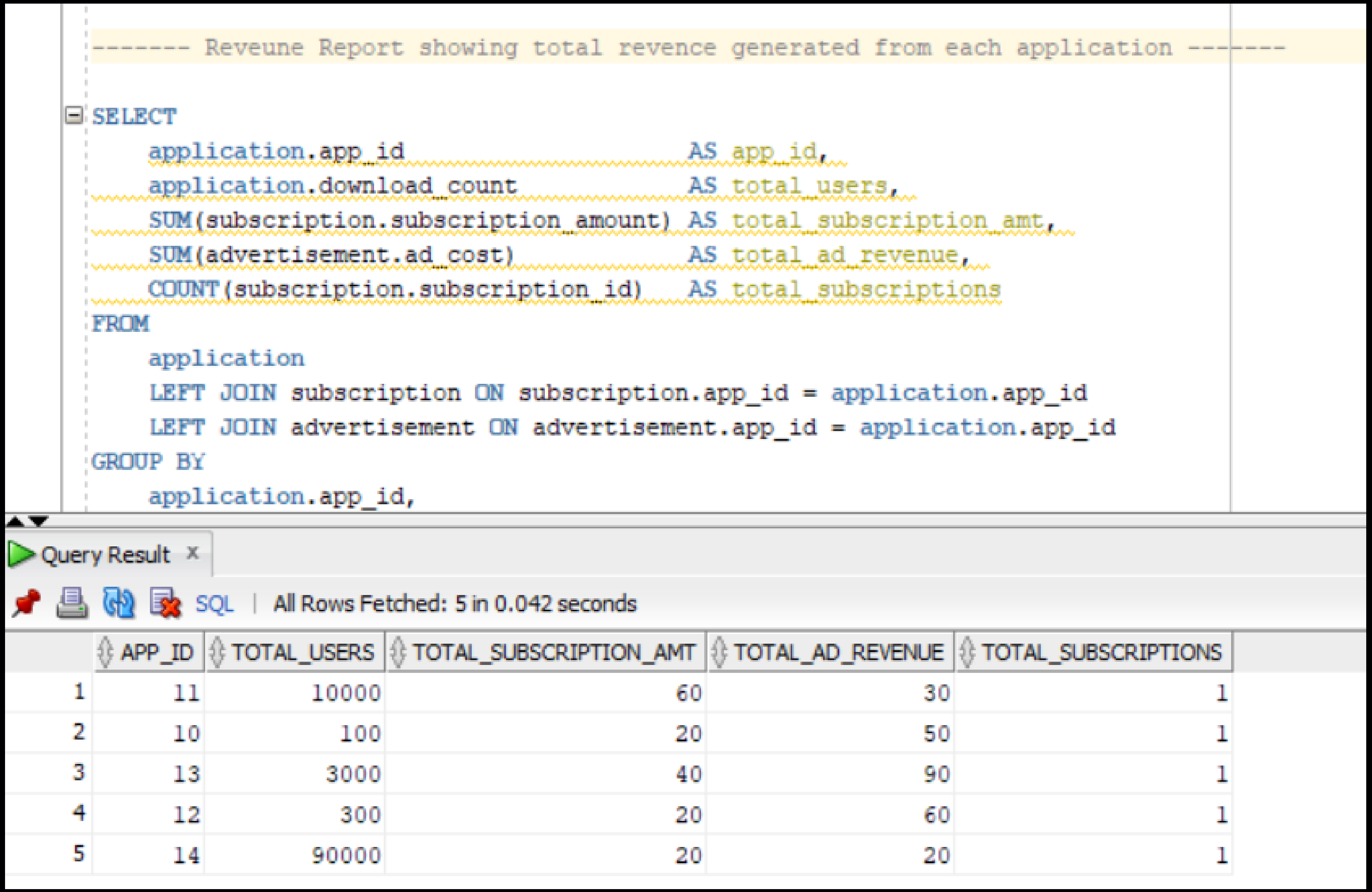
```
--- Application categorized report under given category and Overall rating

SELECT
    category_type,
    overall_rating,
    trunc(app_create_dt)   create_date,
    COUNT(DISTINCT app_id) total_apps
FROM
         application a
    JOIN app_category b ON a.category_id = b.category_id
GROUP BY
    category_type,
    overall_rating,
    trunc(app_create_dt);
```

Query Result

SQL | All Rows Fetched: 5 in 0.04 seconds

	CATEGORY_TYPE	OVERALL_RATING	CREATE_DATE	TOTAL_APPS
1	Entertainment	4	03-MAR-23	1
2	Business	7	18-JUL-22	1
3	Lifestyle	8	05-MAR-22	1
4	Entertainment	3	28-MAY-22	1
5	Business	6	06-MAR-22	1

REPORTS



```
------- Reveune Report showing total revence generated from each application -------

SELECT
    application.app_id                        AS app_id,
    application.download_count                AS total_users,
    SUM(subscription.subscription_amount) AS total_subscription_amt,
    SUM(advertisement.ad_cost)                AS total_ad_revenue,
    COUNT(subscription.subscription_id)    AS total_subscriptions
FROM
    application
    LEFT JOIN subscription ON subscription.app_id = application.app_id
    LEFT JOIN advertisement ON advertisement.app_id = application.app_id
GROUP BY
    application.app_id,
```

Query Result

SQL | All Rows Fetched: 5 in 0.042 seconds

	APP_ID	TOTAL_USERS	TOTAL_SUBSCRIPTION_AMT	TOTAL_AD_REVENUE	TOTAL_SUBSCRIPTIONS
1	11	10000	60	30	1
2	10	100	20	50	1
3	13	3000	40	90	1
4	12	300	20	60	1
5	14	90000	20	20	1