DBMS PROJECT ON PLAYSTORE

Designed by

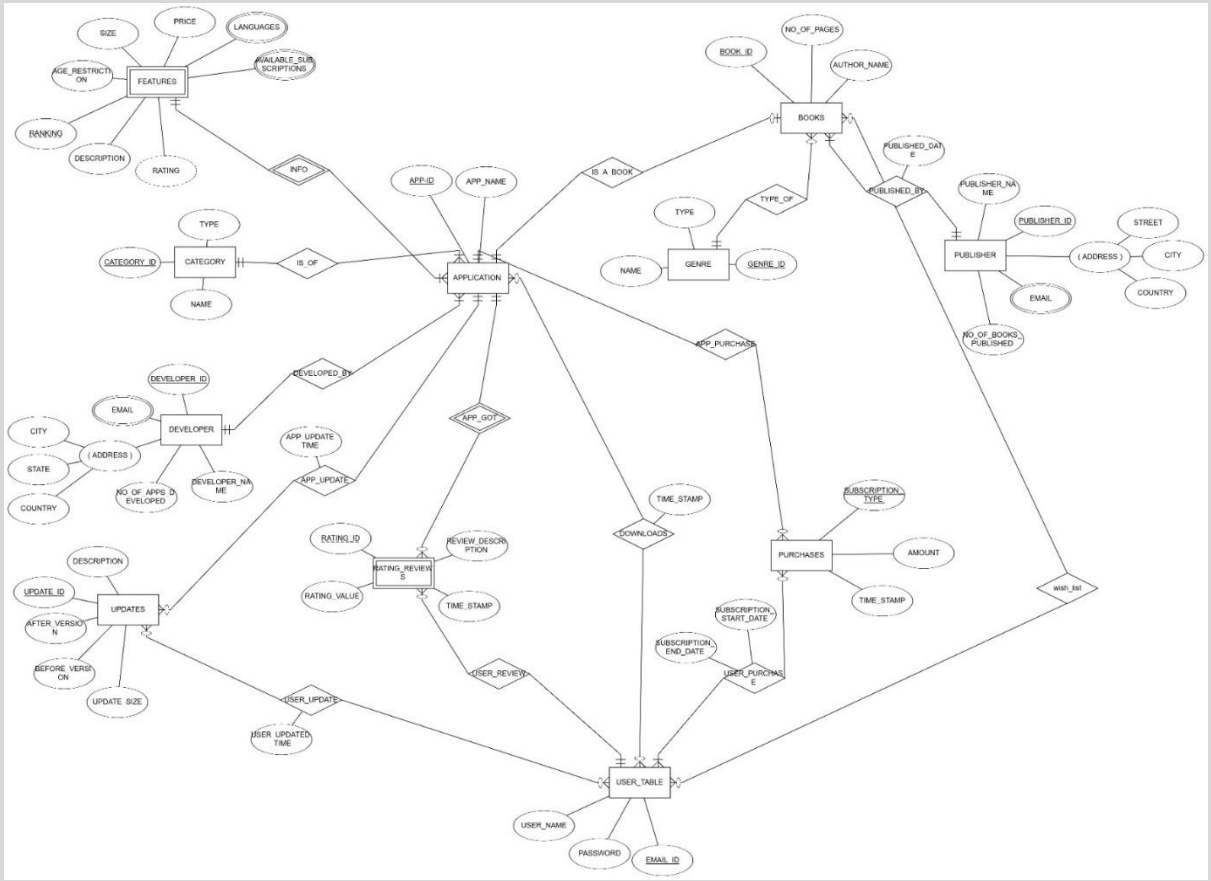
ABHI SAI MURAMSETTI, 21CSB0B37

SAI ASHISH REDDY,21CSB0B12

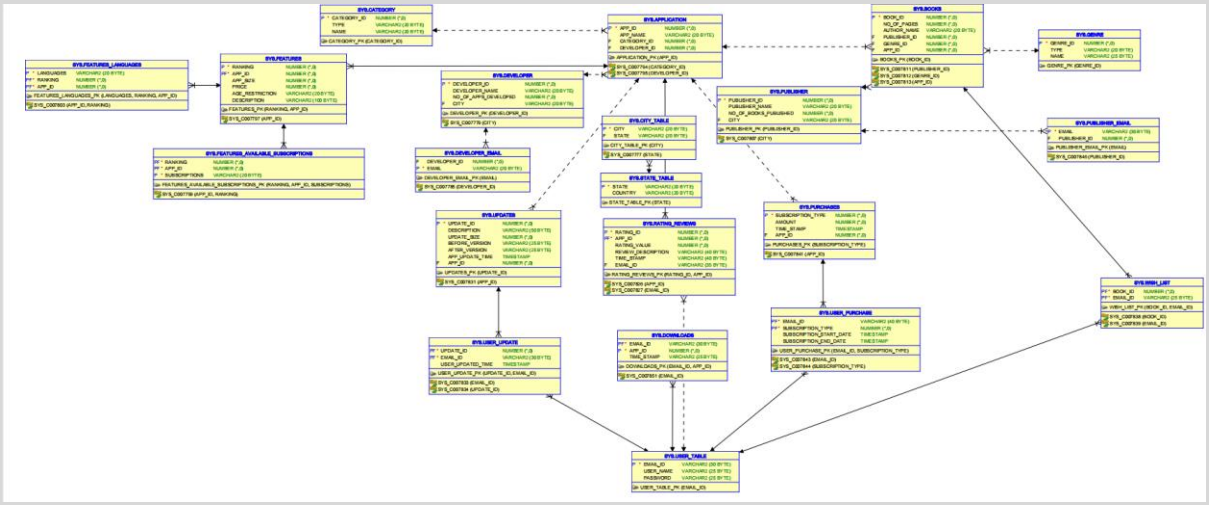
PROBLEM STATEMENT

The project is about managing database in Play Store. It covers collecting data on categories of applications and collecting ratings, size, reviews, number of downloads, description, price, offers of respective apps and userId, password, search history, updates, purchases, wishlist, etc for respective users. It also covers collecting data on books and audiobooks, details of the application creators, etc.

ER DIAGRAM



RELATION SCHEMA



ASSUMPTIONS

ASSUMPTIONS:

One developer can develop any number of games but one game can be developed by only one developer.

One publisher can develop any number of books but one book can be published by only one publisher.

Rating_id can be same for different applications but different for one particular application .

An app can have multiple updates and subscriptions but user can buy one or more subscriptions among them.

Any developer and publisher can have multiple email id

NORMALISATION

Entity: APPLICATION

Attributes:

- APP_ID
- APP_NAME
- CATEGORY_ID
- DEVELOPER_ID

Functional dependencies:

- APP_ID -> APP_NAME
- APP_ID -> CATEGORY_ID
- APP_ID -> DEVELOPER_ID

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: BOOKS

Attributes:

- BOOK_ID
- NO_OF_PAGES
- PUBLISHER_ID
- GENRE_ID

Functional dependencies:

- BOOK_ID -> NO_OF_PAGES
- BOOK_ID -> PUBLISHER_ID
- APP_ID -> GENRE_ID

1NF:

This relation doesn’t contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: PUBLISHER

Attributes:

- PUBLISHER_ID
- ADDRESS (COUNTRY, CITY, STATE)
- NO_OF_BOOKS_PUBLISHED

1NF:

This relation contains a composite attribute

(ADDRESS) and also a multi valued attribute email.

So,it doesn’t satisfy the 1NF condition.

Write the composite attribute as separate attributes.

And let’s make a separate table for email.

Attributes:

- PUBLISHER_ID
- COUNTRY
- CITY
- STATE
- NO_OF_BOOKS_PUBLISHED

Functional dependencies:

- PUBLISHER_ID -> COUNTRY
- PUBLISHER_ID -> CITY
- PUBLISHER_ID -> STATE
- PUBLISHER_ID -> NO_OF_BOOKS_PUBLISHED
- CITY -> COUNTRY
- STATE -> COUNTRY
- CITY-> STATE

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF.

3NF:

the above relation isC not in 3NF because there are some dependencies where a non-prime attributes determines another non-prime attribute,which means that there are some transitive dependencies.

They are:

- CITY -> COUNTRY
- STATE -> COUNTRY
- CITY -> STATE

To bring this table into 3NF we should do lossless decomposition. Decomposing the above table into 2 tables.

1) **PUBLISHER**-
   PUBLISHER_ID
   NO_OF_BOOKS_PUBLISHED
   COUNTRY
2) **CITY-**
   CITY
   COUNTRY

3). **STATE-**

STATE

COUNTRY

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: PUBLISHER_EMAIL

Attributes:

- EMAIL
- PUBLISHER_ID

Functional dependencies:

- EMAIL -> PUBLISHER_ID

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF.

Entity: CATEGORY

Attributes:

- CATEGORY_ID
- TYPE
- NAME

Functional dependencies:

- CATEGORY_ID -> TYPE
- CATEGORY_ID -> NAME

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: USER

Attributes:

- EMAIL_ID
- USER_NAME
- PASSWORD

Functional dependencies:

- EMAIL_ID -> USER_NAME
- EMAIL_ID -> PASSWORD

1NF:

This relation doesn't contain any multivalued attributes so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: GENRE

Attributes:

- GENRE_ID
- TYPE
- NAME

Functional dependencies:

- GENRE_ID -> TYPE
- GENRE_ID -> NAME

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: WISH_LIST

Attributes:

- EMAIL
- BOOK_ID

Functional dependencies:

As the both attributes together form a candidate key there are no functional dependencies expect the trivial ones.

So, the relation satisfies conditions of **1NF,2NF,3NF, BCNF.**

Entity: FEATURES

Attributes:

- RANKING
- APP_ID
- PRICE
- SIZE
- AGE_RESTRICTION
- DESCRIPTION
- LANGUAGES
- AVAILABLE SUBSCRIPTIONS

1NF:

This relation contains multivalued attributes(languages ,available_subscriptions) so we have to make separate tables for them.

Attributes:

- RANKING
- APP_ID
- PRICE
- SIZE
- AGE_RESTRICTION
- DESCRIPTION

Functional dependencies:

- RANKING APP_ID -> PRICE
- RANKING APP_ID -> SIZE
- RANKING APP_ID -> AGE_RESTRICTION
- RANKING APP_ID -> DESCRIPTION

2NF:

There is no partial dependency , so the relation is in 2NF.

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF.

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF.

Entity: FEATURES_LANGUAUGES

Attributes:

- RANKING
- APP_ID
- LANGUAGES

Functional dependencies:

As all the attributes together form a candidate key there are no functional dependencies expect the trivial ones.

So, the relation satisfies conditions of **1NF,2NF,3NF, BCNF.**

Entity: FEATURES_AVAILABLE_SUBSCRIPTIONS

Attributes:

- RANKING
- APP_ID
- AVAILABLE_SUBSCRIPTIONS

Functional dependencies:

As all the attributes together form a candidate key there are no functional dependencies expect the trivial ones.

So, the relation satisfies conditions of **1NF,2NF,3NF, BCNF.**

Entity: DEVELOPER

Attributes:

- DEVELOPER_ID
- ADDRESS (COUNTRY,STATE,CITY)
- EMAIL
- NO_OF_APPS_DEVELOPED

1NF:

This relation contains a composite attribute

(address) and also a multi valued attribute email.

so, it doesn't satisfy the 1nf condition.

write the composite attribute as separate attributes.

and let's make a separate table for email.

Attributes:

- DEVELOPER_ID
- COUNTRY
- CITY
- STATE
- NO_OF_BOOKS_DEVELOPED

Functional dependencies:

- DEVELOPE_ID -> COUNTRY
- DEVELOPER_ID -> CITY
- DEVELOPER_ID -> STATE
- DEVELOPER_ID -> NO_OF_BOOKS_PUBLISHED
- CITY -> COUNTRY
- STATE -> COUNTRY
- CITY-> STATE

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

the above relation is not in 3NF because there are some dependencies where a non-prime attributes determines another non-prime attribute ,which means that there are some transitive dependencies.

They are:

- CITY -> COUNTRY
- STATE -> COUNTRY
- CITY -> STATE

To bring this table into 3NF we should do lossless decomposition. Decomposing the above table into 2 tables.

1) **DEVELOPER-**
   DEVELOPER_ID
   NO_OF_BOOKS_DEVELOPED
   COUNTRY
2) **CITY-**
   CITY
   COUNTRY

3). **STATE-**

STATE

COUNTRY

CITY AND STATE TABLES ARE ALREADY CREATED DURING NORMALISATION OF PUBLISHER TABLE.JUST USE THEM.

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: DEVELOPER_EMAIL

Attributes:

- EMAIL
- DEVELOPER_ID

Functional dependencies:

- EMAIL -> DEVELOPER_ID

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF.

Entity: UPDATE

Attributes:

- UPDATE_ID
- DESCRIPTION
- UPDATE_SIZE
- AFTER_VERSION
- BEFORE_VERSION
- APP_ID
- APP_UPDATE_TIME

Functional dependencies:

- UPDATE_ID -> DESCRIPTION
- UPDATE_ID -> UPDATE_SIZE
- UPDATE_ID -> AFTER_VERSION
- UPDATE_ID -> BEFORE_VERSION
- UPDATE_ID -> APP_UPDATE_TIME
- UPDATE_ID -> APP_ID

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

Here our candidate key has only one attribute so there is no chance of having a partial dependency.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: USER_UPDATE

Attributes:

- UPDATE_ID
- EMAIL_ID
- USER_UPDATE_TIME

Functional dependencies:

- UPDATE_ID EMAIL_ID ->USER_UPDATE_TIME

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

There is no partial dependency , so the relation is in 2NF.

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: PURCHASES

Attributes:

- PAYMENT_ID
- APP_ID
- AMOUNT
- TIME_STAMP
- SUBSCRIPTION_START_DATE
- SUBSCRIPTION_END_DATE
- USER_UPDATE_TIME

Functional dependencies:

- PAYMENT_ID -> APP_ID
- PAYMENT_ID ->AMOUNT
- PAYMENT_ID ->TIME_STAMP
- PAYMENT_ID ->SUBSCRIPTION_START_DATE
- PAYMENT_ID ->SUBSCRIPTION_END_DATE

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

There is no partial dependency, so the relation is in 2NF.

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: USER_PURCHASE

Attributes:

- EMAIL_ID
- PAYMENT_ID

Functional dependencies:

As the both attributes together form a candidate key there are no functional dependencies expect the trivial ones.

So, the relation satisfies conditions of **1NF,2NF,3NF, BCNF.**

Entity: RATING REVIEWS

Attributes:

- RATING_ID
- APP_ID
- RATING_VALUE
- REVIEW_DESCRIPTION
- TIME_STAMP

Functional dependencies:

- RATING_ID APP_ID -> RATING_VALUE
- RATING_ID APP_ID -> REVIEW_DESCRIPTION
- RATING_ID APP_ID -> TIME_STAMP
- RATING_ID APP_ID -> EMAIL_ID

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

There is no partial dependency, so the relation is in 2NF.

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

Entity: DOWNLOADS

Attributes:

- EMAIL_ID
- APP_ID
- TIME_STAMP

Functional dependencies:

- EMAIL_ID APP_ID -> TIME_STAMP

1NF:

This relation doesn't contain any multivalued attributes or composite attribute so it is already in 1NF.

2NF:

There is no partial dependency in the above relation.

So, the relation is in 2NF

3NF:

In the above relation, all the dependencies are from candidate key to non-prime attributes.

So, the relation is in 3NF

BCNF:

Every dependency has only the super key determining the other attributes.

So, the relation is in BCNF

RELATIONS AND INSERTIONS

```
CREATE TABLE DEVELOPER(
DEVELOPER_ID INT PRIMARY KEY,
DEVELOPER_NAME VARCHAR(20),
NO_OF_APPS_DEVELOPED INT,
CITY VARCHAR(20),
FOREIGN KEY (CITY) REFERENCES CITY_TABLE(CITY)
);
DESC DEVELOPER;
drop table developer;
```

```
INSERT INTO DEVELOPER  VALUES (1, 'John Smith', 5, 'San Francisco');
INSERT INTO DEVELOPER  VALUES  (2, 'Mary Johnson', 10, 'San Francisco');
INSERT INTO DEVELOPER  VALUES  (3, 'Alex Kim', 20, 'Seattle');
INSERT INTO DEVELOPER  VALUES (4, 'Emily Lee', 3, 'Los Angeles');
INSERT INTO DEVELOPER  VALUES (5, 'David Chen', 7, 'Houston');
INSERT INTO DEVELOPER  VALUES (6, 'Karen Wong', 15, 'Chicago');
INSERT INTO DEVELOPER  VALUES (7, 'abhi', 2000, 'Chicago');
CREATE TABLE CITY_TABLE(
CITY VARCHAR(20) PRIMARY KEY ,
STATE VARCHAR(20) ,
FOREIGN KEY (STATE) REFERENCES STATE_TABLE(STATE)
);
INSERT INTO CITY_TABLE VALUES ('San Francisco', 'California');
 INSERT INTO CITY_TABLE VALUES ('New York', 'New York');
 INSERT INTO CITY_TABLE VALUES ('Seattle', 'Washington');
 INSERT INTO CITY_TABLE VALUES ('Los Angeles', 'California');
 INSERT INTO CITY_TABLE VALUES ('Houston', 'Texas');
 INSERT INTO CITY_TABLE VALUES ('Chicago', 'Illinois');

drop table CITY_TABLE;

CREATE TABLE STATE_TABLE(
STATE VARCHAR(20) PRIMARY KEY,
COUNTRY VARCHAR(20)
);
DROP TABLE CITY_TABLE;
INSERT INTO STATE_TABLE  VALUES  ('California', 'United States');
INSERT INTO STATE_TABLE  VALUES  ('New York', 'United States');
INSERT INTO STATE_TABLE  VALUES  ('Washington', 'United States');
INSERT INTO STATE_TABLE  VALUES  ('Texas', 'United States');
INSERT INTO STATE_TABLE  VALUES  ('Illinois', 'United States');
INSERT INTO STATE_TABLE  VALUES  ('Florida', 'United States');



drop table STATE_TABLE;
```

```
CREATE TABLE DEVELOPER_EMAIL(
DEVELOPER_ID INT,
EMAIL varchar(20) PRIMARY KEY,
FOREIGN KEY (DEVELOPER_ID) REFERENCES DEVELOPER
);
drop table developer_email;
INSERT INTO DEVELOPER_EMAIL  VALUES (1, 'john@example.com');
INSERT INTO DEVELOPER_EMAIL  VALUES   (2, 'mary@example.com');
INSERT INTO DEVELOPER_EMAIL  VALUES  (3, 'alex@example.com');
INSERT INTO DEVELOPER_EMAIL  VALUES  (4, 'emily@example.com');
INSERT INTO DEVELOPER_EMAIL  VALUES  (5, 'david@example.com');
 INSERT INTO DEVELOPER_EMAIL  VALUES  (6, 'karen@example.com');
 INSERT INTO DEVELOPER_EMAIL  VALUES  (7, 'abhi@example.com');
select * from DEVELOPER_EMAIL;
```

```
CREATE TABLE CATEGORY(
CATEGORY_ID INT PRIMARY KEY,
TYPE VARCHAR(20),
NAME VARCHAR(20)
);

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (1, 'Game', 'Action');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (2, 'Game', 'Puzzle');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (3, 'Game', 'Sports');
```

```
INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (4, 'Game', 'Strategy');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (5, 'Game', 'Adventure');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (6, 'General', 'Social');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (7, 'General', 'Productivity');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (8, 'General', 'Entertainment');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (9, 'General', 'News');

INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (10, 'General', 'Education');
INSERT INTO CATEGORY (CATEGORY_ID, TYPE, NAME)
VALUES (11, 'BOOKS', 'BOOKS');
---------------------------------------
create table APPLICATION(
APP_ID INT PRIMARY KEY,
APP_NAME VARCHAR(20),
CATEGORY_ID INT,
DEVELOPER_ID INT,
FOREIGN KEY (CATEGORY_ID) REFERENCES CATEGORY,
FOREIGN KEY (DEVELOPER_ID) REFERENCES DEVELOPER
);
drop table APPLICATION;
INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (1, 'Angry Birds', 1, 1);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
```

```
VALUES (2, 'Candy Crush', 2, 2);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (3, 'FIFA', 3, 3);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (4, 'Civilization', 4, 4);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (5, 'Minecraft', 5, 5);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (6, 'Facebook', 6, 5);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (7, 'Instagram', 6, 6);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (8, 'Twitter', 6, 6);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (9, 'Microsoft Word', 7, 5);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (10, 'Netflix', 8, 5);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (11, 'YouTube', 8, 5);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (12, 'The New York Times', 9, 6);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (13, 'Duolingo', 10, 6);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
```

```
VALUES (14, 'Khan Academy', 10, 6);

INSERT INTO APPLICATION (APP_ID, APP_NAME, CATEGORY_ID, DEVELOPER_ID)
VALUES (15, 'Quizlet', 10, 6);
--books---
INSERT INTO APPLICATION VALUES (16, 'Gatsby', 11, 7);
INSERT INTO APPLICATION VALUES(17, 'Mockingbird', 11, 7);
INSERT INTO APPLICATION VALUES(18, 'Da Vinci',11, 7);
INSERT INTO APPLICATION VALUES(19, 'PridePrejudice', 11, 7);
INSERT INTO APPLICATION VALUES(20, 'Dune', 11, 7);
INSERT INTO APPLICATION VALUES(21, '1984', 11, 7);
INSERT INTO APPLICATION VALUES(22, 'GirlDragonTattoo', 11, 7);
INSERT INTO APPLICATION VALUES(23, 'Outlander', 11, 7);
INSERT INTO APPLICATION VALUES(24, 'Ender's ', 11, 7);
INSERT INTO APPLICATION VALUES(25, 'Catcher Rye', 11, 7);
select * from application;


CREATE TABLE FEATURES(
RANKING INT,
APP_ID INT,
APP_SIZE INT,
PRICE INT,
AGE_RESTRICTION VARCHAR(20),
DESCRIPTION VARCHAR(100),
FOREIGN KEY (APP_ID) REFERENCES APPLICATION(APP_ID),
PRIMARY KEY (RANKING,APP_ID)
);
INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (1, 1, 100, 0, '4+', 'The classic bird-flinging game that started it all.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (2, 2, 150, 0, '4+', 'Match candies to progress through levels.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (3, 3, 2000, 4.99, '4+', 'Play with your favorite soccer teams and compete in tournaments.');
```

```
INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (4, 4, 500, 19.99, '12+', 'Build and lead a civilization from ancient times to the modern era.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (5, 5, 300, 6.99, '7+', 'Explore and build in a blocky 3D world.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (6, 6, 300, 0, '12+', 'Connect with friends and family and discover new things.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (7, 7, 350, 0, '12+', 'Share photos and videos with your followers.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (8, 8, 400, 0, '17+', 'Join the conversation and get real-time updates on news and events.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (9, 9, 600, 149.99, '4+', 'Create and edit documents and collaborate with others.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (10, 10, 700, 9.99, '17+', 'Watch TV shows and movies on your device.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (11, 11, 750, 0, '17+', 'Discover, watch, and share videos.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (12, 12, 800, 0, '12+', 'Stay informed with breaking news and top stories.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (13, 13, 900, 0, '4+', 'Learn a new language for free.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (14, 14, 1000, 0, '4+', 'Get free lessons and practice exercises in math, science, and more.');

INSERT INTO FEATURES (RANKING, APP_ID, APP_SIZE, PRICE, AGE_RESTRICTION, DESCRIPTION)
VALUES (15, 15, 1000, 0, '4+', 'take quizzes');
```

```
CREATE TABLE FEATURES(
RANKING INT,
APP_ID INT,
APP_SIZE INT,
PRICE INT,
AGE_RESTRICTION VARCHAR(20),
DESCRIPTION VARCHAR(100),
FOREIGN KEY (APP_ID) REFERENCES APPLICATION(APP_ID),
PRIMARY KEY (RANKING,APP_ID)
);

-- Insert sample data into the "FEATURES" table

INSERT INTO FEATURES VALUES(1, 16, 25, 4.99, 'No age restriction', 'E-book version of The Great Gatsby');
INSERT INTO FEATURES VALUES(2, 17, 35, 5.99, 'No age restriction', 'Audiobook version of To Kill a Mockingbird');
INSERT INTO FEATURES VALUES(3, 18, 28, 3.99, 'No age restriction', 'E-book version of The Da Vinci Code');
INSERT INTO FEATURES VALUES(4, 19, 20, 2.99, 'No age restriction', 'E-book version of Pride and Prejudice');
INSERT INTO FEATURES VALUES(5, 20, 50, 6.99, 'No age restriction', 'Audiobook version of Dune');
INSERT INTO FEATURES VALUES(6, 21, 30, 4.99, 'No age restriction', 'E-book version of 1984');
INSERT INTO FEATURES VALUES(7, 22, 40, 5.99, 'No age restriction', 'Audiobook version of The Girl with the Dragon Tattoo');
INSERT INTO FEATURES VALUES(8, 23, 33, 4.99, 'No age restriction', 'E-book version of Outlander');
INSERT INTO FEATURES VALUES(9, 24, 55, 6.99, 'No age restriction', 'Audiobook version of Ender’s Game');
INSERT INTO FEATURES VALUES(10, 25, 22, 3.99, 'No age restriction', 'E-book version of The Catcher in the Rye');

select * from features;

CREATE TABLE FEATURES_AVAILABLE_SUBSCRIPTIONS(
RANKING INT,
APP_ID INT,
SUBSCRIPTIONS VARCHAR(20),
FOREIGN KEY (APP_ID,RANKING) REFERENCES FEATURES,
PRIMARY KEY(RANKING,APP_ID,SUBSCRIPTIONS)
);
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES      (1, 1, 'Angry Birds Plus');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES      (2, 2, 'Candy Crush Plus');
```

```
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (3, 3, 'FIFA VIP');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (6, 6, 'Facebook Premium');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (6, 6, 'Facebook Plus');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (7, 7, 'Instagram Pro');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (8, 8, 'Twitter Gold');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (8, 8, 'Twitter Plus');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (9, 9, 'Microsoft Word Pro');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (10, 10, 'Netflix Premium');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (10, 10, 'Netflix Plus');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (11, 11, 'YouTube Premium');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (12, 12, 'The New York Times Plus');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (13, 13, 'Duolingo Pro');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (15, 15, 'Quizlet Plus');
INSERT INTO FEATURES_AVAILABLE_SUBSCRIPTIONS VALUES (15, 15, 'Quizlet Pro');
CREATE TABLE FEATURES_LANGUAGES(
LANGUAGES VARCHAR(20),
RANKING INT,
APP_ID INT,
FOREIGN KEY (APP_ID,RANKING) REFERENCES FEATURES,
PRIMARY KEY (LANGUAGES,RANKING,APP_ID)
);
drop table FEATURES_LANGUAGES;
INSERT INTO FEATURES_LANGUAGES  VALUES ('English', 1, 1);
INSERT INTO FEATURES_LANGUAGES  VALUES('Spanish', 2, 2);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 3, 3);
INSERT INTO FEATURES_LANGUAGES  VALUES('French', 4, 4);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 5, 5);
INSERT INTO FEATURES_LANGUAGES  VALUES('Spanish', 6, 6);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 7, 7);
INSERT INTO FEATURES_LANGUAGES  VALUES('Chinese', 8, 8);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 9, 9);
INSERT INTO FEATURES_LANGUAGES  VALUES('German', 10, 10);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 11, 11);
INSERT INTO FEATURES_LANGUAGES  VALUES('Spanish', 12, 12);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 13, 13);
INSERT INTO FEATURES_LANGUAGES  VALUES('Spanish', 14, 14);
```

```
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 15, 15);
INSERT INTO FEATURES_LANGUAGES  VALUES('Japanese', 2, 2);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 10,10 );
INSERT INTO FEATURES_LANGUAGES  VALUES('Spanish', 15, 15);
INSERT INTO FEATURES_LANGUAGES  VALUES('English', 14, 14);
select * from FEATURES_LANGUAGES;
CREATE TABLE PUBLISHER(
PUBLISHER_ID INT PRIMARY KEY,
PUBLISHER_NAME VARCHAR(20),
NO_OF_BOOKS_PUBLISHED INT,
CITY VARCHAR(20),
FOREIGN KEY (CITY) REFERENCES CITY_TABLE(CITY)
);
CREATE TABLE CITY_TABLE(
CITY VARCHAR(20) PRIMARY KEY ,
STATE VARCHAR(20) ,
FOREIGN KEY (STATE) REFERENCES STATE_TABLE(STATE)
);
CREATE TABLE STATE_TABLE(
STATE VARCHAR(20) PRIMARY KEY,
COUNTRY VARCHAR(20)
);

INSERT INTO STATE_TABLE VALUES ('California', 'USA');
INSERT INTO STATE_TABLE VALUES ('Texas', 'USA');
INSERT INTO STATE_TABLE VALUES ('New York', 'USA');
INSERT INTO STATE_TABLE VALUES ('Maharashtra', 'India');
INSERT INTO STATE_TABLE VALUES ('São Paulo', 'Brazil');
INSERT INTO STATE_TABLE VALUES ('Ontario', 'Canada');

INSERT INTO CITY_TABLE VALUES ('Los Angeles', 'California');
INSERT INTO CITY_TABLE VALUES ('Houston', 'Texas');
INSERT INTO CITY_TABLE VALUES ('New York City', 'New York');
INSERT INTO CITY_TABLE VALUES ('Mumbai', 'Maharashtra');
INSERT INTO CITY_TABLE VALUES ('São Paulo', 'São Paulo');
INSERT INTO CITY_TABLE VALUES ('Toronto', 'Ontario');
```

```
select * from publisher;
drop table publisher;
INSERT INTO PUBLISHER VALUES (1, 'Penguin Books', 500, 'New York City');
INSERT INTO PUBLISHER VALUES (2, 'HarperCollins', 700, 'Houston');
INSERT INTO PUBLISHER VALUES (3, 'Random House', 800, 'New York City');
INSERT INTO PUBLISHER VALUES (4, 'Hachette Livre', 600, 'Mumbai');
INSERT INTO PUBLISHER VALUES (5, 'Macmillan Publishers', 400, 'Houston');
INSERT INTO PUBLISHER VALUES (6, 'SimonSchuster', 450, 'New York City');

CREATE TABLE PUBLISHER_EMAIL(
EMAIL VARCHAR(30) PRIMARY KEY,
PUBLISHER_ID INT,
FOREIGN KEY (PUBLISHER_ID) REFERENCES PUBLISHER
);
drop table publisher_email;
INSERT INTO PUBLISHER_EMAIL VALUES ('info@penguinbooks.com', 1);
INSERT INTO PUBLISHER_EMAIL VALUES ('info2@penguinbooks.com', 1);
INSERT INTO PUBLISHER_EMAIL VALUES ('contact@harpercollins.com', 2);
INSERT INTO PUBLISHER_EMAIL VALUES ('info@randomhouse.com', 3);
INSERT INTO PUBLISHER_EMAIL VALUES ('info2@randomhouse.com', 3);
INSERT INTO PUBLISHER_EMAIL VALUES ('contact@hachette-livre.com', 4);
INSERT INTO PUBLISHER_EMAIL VALUES ('info@macmillanpublishers.com', 5);
INSERT INTO PUBLISHER_EMAIL VALUES ('contact@simonandschuster.com', 6);
CREATE TABLE GENRE(
GENRE_ID INT PRIMARY KEY,
TYPE VARCHAR(20),
NAME VARCHAR(25)
);
drop table genre;
CREATE TABLE BOOKS(
BOOK_ID INT PRIMARY KEY,
NO_OF_PAGES INT,
AUTHOR_NAME VARCHAR(20),
PUBLISHER_ID INT,
GENRE_ID INT,
APP_ID INT,
```

```
FOREIGN KEY (PUBLISHER_ID) REFERENCES PUBLISHER,
FOREIGN KEY (GENRE_ID) REFERENCES GENRE,
FOREIGN KEY (APP_ID) REFERENCES APPLICATION
);

INSERT INTO GENRE VALUES(1, 'E-book', 'Fiction');
INSERT INTO GENRE VALUES(2, 'E-book', 'Non-fiction');
INSERT INTO GENRE VALUES(3, 'Audio-book', 'Fiction');
INSERT INTO GENRE VALUES(4, 'Audio-book', 'Non-fiction');

INSERT INTO BOOKS VALUES(1, 300, 'J.K. Rowling', 1, 1, 16);
INSERT INTO BOOKS VALUES(2, 250, 'Stephen King', 2, 1, 17);
INSERT INTO BOOKS VALUES(3, 400, 'Dan Brown', 3, 2, 18);
INSERT INTO BOOKS VALUES(4, 350, 'Jane Austen', 1, 1, 19);
INSERT INTO BOOKS VALUES(5, 600, 'Frank Herbert', 4, 1, 20);
INSERT INTO BOOKS VALUES(6, 200, 'George Orwell', 5, 2, 21);
INSERT INTO BOOKS VALUES(7, 450, 'Stieg Larsson', 6, 3, 22);
INSERT INTO BOOKS VALUES(8, 550, 'Diana Gabaldon', 2, 3, 23);
INSERT INTO BOOKS VALUES(9, 400, 'Orson Scott Card', 6, 1, 24);
INSERT INTO BOOKS VALUES(10, 300, 'J.D. Salinger', 1, 4, 25);



----------------------------------------
CREATE TABLE USER_TABLE(
EMAIL_ID VARCHAR(30) PRIMARY KEY,
USER_NAME VARCHAR(25),
PASSWORD VARCHAR(25)
);
drop table user_table;
INSERT INTO USER_TABLE VALUES ('john.doe@gmail.com', 'John Doe', 'password123');
INSERT INTO USER_TABLE VALUES ('jane.smith@yahoo.com', 'Jane Smith', 'abc123');
INSERT INTO USER_TABLE VALUES ('susan.wilson@hotmail.com', 'Susan Wilson', 'ilovecats');
INSERT INTO USER_TABLE VALUES ('peter.jackson@gmail.com', 'Peter Jackson', 'lotr123');
INSERT INTO USER_TABLE VALUES ('mary.johnson@gmail.com', 'Mary Johnson', 'password456');
INSERT INTO USER_TABLE VALUES ('bob.dylan@yahoo.com', 'Bob Dylan', 'blowinthewind');
INSERT INTO USER_TABLE VALUES ('lisa.anderson@gmail.com', 'Lisa Anderson', 'letitbe');
```

```
INSERT INTO USER_TABLE VALUES ('michael.nguyen@hotmail.com', 'Michael Nguyen', 'coffee123');
INSERT INTO USER_TABLE VALUES ('emily.roberts@yahoo.com', 'Emily Roberts', 'happylife');
INSERT INTO USER_TABLE VALUES ('jason.miller@gmail.com', 'Jason Miller', 'qwerty123');
CREATE TABLE DOWNLOADS(
EMAIL_ID varchar(30),
APP_ID INT,
TIME_STAMP VARCHAR(25),
FOREIGN KEY (EMAIL_ID) REFERENCES USER_TABLE,
FOREIGN KEY (APP_ID) REFERENCES APPLICATION,
PRIMARY KEY (EMAIL_ID,APP_ID)
);

drop table downloads;
select* from downloads;
INSERT INTO DOWNLOADS VALUES('john.doe@gmail.com', 1, '2023-05-13 12:30:00');
INSERT INTO DOWNLOADS VALUES('jane.smith@yahoo.com', 1, '2023-05-13 13:45:00');
INSERT INTO DOWNLOADS VALUES('susan.wilson@hotmail.com', 2, '2023-05-13 10:15:00');
INSERT INTO DOWNLOADS VALUES('peter.jackson@gmail.com', 2, '2023-05-13 11:20:00');
INSERT INTO DOWNLOADS VALUES('mary.johnson@gmail.com', 3, '2023-05-13 14:00:00');
INSERT INTO DOWNLOADS VALUES('bob.dylan@yahoo.com', 3, '2023-05-13 15:30:00');
INSERT INTO DOWNLOADS VALUES('lisa.anderson@gmail.com', 4, '2023-05-13 16:45:00');
INSERT INTO DOWNLOADS VALUES('michael.nguyen@hotmail.com', 5, '2023-05-13 17:00:00');
INSERT INTO DOWNLOADS VALUES('emily.roberts@yahoo.com', 5, '2023-05-13 18:15:00');
INSERT INTO DOWNLOADS VALUES('jason.miller@gmail.com', 6, '2023-05-13 19:30:00');
INSERT INTO DOWNLOADS VALUES('jane.smith@yahoo.com', 6, '2023-05-13 20:45:00');
INSERT INTO DOWNLOADS VALUES('susan.wilson@hotmail.com', 7, '2023-05-13 21:15:00');
INSERT INTO DOWNLOADS VALUES('peter.jackson@gmail.com', 8, '2023-05-13 22:20:00');
INSERT INTO DOWNLOADS VALUES('mary.johnson@gmail.com', 8, '2023-05-13 23:00:00');
INSERT INTO DOWNLOADS VALUES('bob.dylan@yahoo.com', 10, '2023-05-13 23:30:00');
select * from downloads;
CREATE TABLE RATING_REVIEWS(
RATING_ID INT,
APP_ID INT,
RATING_VALUE INT,
REVIEW_DESCRIPTION VARCHAR(40),
TIME_STAMP VARCHAR(40),
```

```
EMAIL_ID  VARCHAR(35),

FOREIGN KEY(APP_ID) REFERENCES APPLICATION,

FOREIGN KEY(EMAIL_ID) REFERENCES USER_TABLE,

PRIMARY KEY(RATING_ID,APP_ID)

);

drop table RATING_REVIEWS;

INSERT INTO RATING_REVIEWS VALUES (1, 1, 4, 'Fun game!', '2023-05-13 13:00:00', 'john.doe@gmail.com');

INSERT INTO RATING_REVIEWS VALUES (2, 1, 3, 'Too many ads', '2023-05-13 14:00:00',
'jane.smith@yahoo.com');

INSERT INTO RATING_REVIEWS VALUES (1, 2, 5, 'Addictive!', '2023-05-13 15:00:00',
'susan.wilson@hotmail.com');

INSERT INTO RATING_REVIEWS VALUES (2, 2, 2, 'Boring after a while', '2023-05-13 16:00:00',
'peter.jackson@gmail.com');

INSERT INTO RATING_REVIEWS VALUES (1, 3, 4, 'Great soccer game', '2023-05-13 17:00:00',
'mary.johnson@gmail.com');

INSERT INTO RATING_REVIEWS VALUES (2, 3, 1, 'Worst game ever', '2023-05-13 18:00:00',
'bob.dylan@yahoo.com');

INSERT INTO RATING_REVIEWS VALUES (1, 4, 5, 'Love this strategy game', '2023-05-13 19:00:00',
'lisa.anderson@gmail.com');

INSERT INTO RATING_REVIEWS VALUES (2, 4, 4, 'Great graphics', '2023-05-13 20:00:00',
'michael.nguyen@hotmail.com');

INSERT INTO RATING_REVIEWS VALUES (1, 5, 3, 'Not my thing', '2023-05-13 21:00:00',
'emily.roberts@yahoo.com');

INSERT INTO RATING_REVIEWS VALUES (2, 5, 5, 'Amazing!', '2023-05-13 22:00:00',
'jason.miller@gmail.com');


CREATE TABLE UPDATES(

UPDATE_ID INT PRIMARY KEY,

DESCRIPTION VARCHAR(50),

UPDATE_SIZE INT,

BEFORE_VERSION VARCHAR(25),

AFTER_VERSION VARCHAR(25),

APP_UPDATE_TIME TIMESTAMP,

APP_ID INT,

FOREIGN KEY (APP_ID) REFERENCES APPLICATION

);

drop table updates;

INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (1, 'Bug fixes and performance improvements', 10, '1.2.3', '1.2.4',TIMESTAMP '2023-2-14 09:00:00', 1);
```

```
INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (2, 'New levels and challenges', 50, '2.5.1', '2.6.0', TIMESTAMP'2023-05-14 10:00:00', 2);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (3, 'New team kits and player updates', 20, '1.1.2', '1.2.0', TIMESTAMP'2023-05-14 11:00:00', 3);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (4, 'New civilizations and wonders', 80, '4.2.1', '4.3.0',TIMESTAMP '2023-05-14 12:00:00', 4);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (5, 'New mobs and biomes', 60, '1.1.1', '1.2.0',TIMESTAMP '2023-05-14 13:00:00', 5);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (6, 'Bug fixes and performance improvements', 15, '10.5.3', '10.5.4',TIMESTAMP '2023-05-14 14:00:00',
6);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (7, 'New filters and effects', 30, '2.3.1', '2.4.0', TIMESTAMP'2023-05-14 15:00:00', 7);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (8, 'New features and bug fixes', 40, '6.1.1', '6.2.0',TIMESTAMP '2023-05-14 16:00:00', 8);


INSERT INTO UPDATES (UPDATE_ID, DESCRIPTION, UPDATE_SIZE, BEFORE_VERSION,
AFTER_VERSION, APP_UPDATE_TIME, APP_ID)

VALUES (9, 'New templates and design options', 25, '3.0.0', '3.0.1',TIMESTAMP '2023-05-14 17:00:00', 9);


select * from UPDATES;

CREATE TABLE USER_UPDATE(

UPDATE_ID INT,

EMAIL_ID varchar(30),

USER_UPDATED_TIME TIMESTAMP,

PRIMARY KEY(UPDATE_ID,EMAIL_ID),

FOREIGN KEY(EMAIL_ID) REFERENCES USER_TABLE,
```

```
FOREIGN KEY(UPDATE_ID) REFERENCES UPDATES
);
INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (1, 'john.doe@gmail.com', TIMESTAMP'2023-05-14 09:30:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (2, 'jane.smith@yahoo.com', TIMESTAMP'2023-05-14 10:15:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (3, 'susan.wilson@hotmail.com', TIMESTAMP'2023-05-14 11:45:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (4, 'peter.jackson@gmail.com',TIMESTAMP '2023-05-14 12:30:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (5, 'mary.johnson@gmail.com',TIMESTAMP '2023-05-14 13:15:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (6, 'bob.dylan@yahoo.com', TIMESTAMP'2023-05-14 14:00:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (7, 'lisa.anderson@gmail.com', TIMESTAMP'2023-05-14 15:45:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (8, 'michael.nguyen@hotmail.com',TIMESTAMP '2023-05-14 16:30:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (9, 'emily.roberts@yahoo.com', TIMESTAMP'2023-05-14 17:15:00');

INSERT INTO USER_UPDATE (UPDATE_ID, EMAIL_ID, USER_UPDATED_TIME)
VALUES (1, 'jason.miller@gmail.com',TIMESTAMP '2023-05-14 18:00:00');



CREATE TABLE WISH_LIST(
BOOK_ID INT,
EMAIL_ID VARCHAR(25),
```

```
FOREIGN KEY(BOOK_ID) REFERENCES BOOKS,
FOREIGN KEY(EMAIL_ID) REFERENCES USER_TABLE,
PRIMARY KEY(BOOK_ID,EMAIL_ID)
);
INSERT INTO WISH_LIST (BOOK_ID, EMAIL_ID)
VALUES (1, 'john.doe@gmail.com');

INSERT INTO WISH_LIST (BOOK_ID, EMAIL_ID)
VALUES (2, 'jane.smith@yahoo.com');

INSERT INTO WISH_LIST (BOOK_ID, EMAIL_ID)
VALUES (3, 'susan.wilson@hotmail.com');

CREATE TABLE PURCHASES(
SUBSCRIPTION_TYPE INT PRIMARY KEY,
AMOUNT INT,
TIME_STAMP TIMESTAMP,
APP_ID INT,
FOREIGN KEY(APP_ID) REFERENCES APPLICATION
);
INSERT INTO PURCHASES (SUBSCRIPTION_TYPE, AMOUNT, TIME_STAMP, APP_ID)
VALUES (1, 10,TIMESTAMP '2023-05-14 14:30:00', 1);

INSERT INTO PURCHASES (SUBSCRIPTION_TYPE, AMOUNT, TIME_STAMP, APP_ID)
VALUES (2, 50, TIMESTAMP'2023-05-13 12:00:00', 2);

INSERT INTO PURCHASES (SUBSCRIPTION_TYPE, AMOUNT, TIME_STAMP, APP_ID)
VALUES (3, 20,TIMESTAMP '2023-05-12 16:45:00', 3);

INSERT INTO PURCHASES (SUBSCRIPTION_TYPE, AMOUNT, TIME_STAMP, APP_ID)
VALUES (4, 150, TIMESTAMP'2023-05-11 10:15:00', 6);

INSERT INTO PURCHASES (SUBSCRIPTION_TYPE, AMOUNT, TIME_STAMP, APP_ID)
VALUES (5, 80,TIMESTAMP '2023-05-10 09:30:00', 7);
CREATE TABLE USER_PURCHASE(
EMAIL_ID VARCHAR(40) ,
```

```
SUBSCRIPTION_TYPE INT,

SUBSCRIPTION_START_DATE TIMESTAMP,

SUBSCRIPTION_END_DATE TIMESTAMP,

FOREIGN KEY(EMAIL_ID) REFERENCES USER_TABLE,

FOREIGN KEY(SUBSCRIPTION_TYPE) REFERENCES PURCHASES,

PRIMARY KEY(EMAIL_ID,SUBSCRIPTION_TYPE)

);

INSERT INTO USER_PURCHASE (EMAIL_ID, SUBSCRIPTION_TYPE, SUBSCRIPTION_START_DATE,
SUBSCRIPTION_END_DATE)

VALUES ('john.doe@gmail.com', 1, TIMESTAMP'2023-05-14 14:30:00',TIMESTAMP '2024-05-14 14:30:00');


INSERT INTO USER_PURCHASE (EMAIL_ID, SUBSCRIPTION_TYPE, SUBSCRIPTION_START_DATE,
SUBSCRIPTION_END_DATE)

VALUES ('jane.smith@yahoo.com', 2, TIMESTAMP'2023-05-13 12:00:00', TIMESTAMP'2024-05-13 12:00:00');


INSERT INTO USER_PURCHASE (EMAIL_ID, SUBSCRIPTION_TYPE, SUBSCRIPTION_START_DATE,
SUBSCRIPTION_END_DATE)

VALUES ('susan.wilson@hotmail.com', 3, TIMESTAMP'2023-05-12 16:45:00',TIMESTAMP '2024-05-12
16:45:00');


INSERT INTO USER_PURCHASE (EMAIL_ID, SUBSCRIPTION_TYPE, SUBSCRIPTION_START_DATE,
SUBSCRIPTION_END_DATE)

VALUES ('peter.jackson@gmail.com', 4,TIMESTAMP '2023-05-11 10:15:00',TIMESTAMP '2024-05-11
10:15:00');


INSERT INTO USER_PURCHASE (EMAIL_ID, SUBSCRIPTION_TYPE, SUBSCRIPTION_START_DATE,
SUBSCRIPTION_END_DATE)

VALUES ('mary.johnson@gmail.com', 5,TIMESTAMP '2023-05-10 09:30:00',TIMESTAMP '2024-05-10
09:30:00');4
```

Tables:

	DEVELOPER_ID	DEVELOPER_NAME	NO_OF_APPS_DEVELOPED	CITY
1	1	John Smith	5	San Francisco
2	2	Mary Johnson	10	San Francisco
3	3	Alex Kim	20	Seattle
4	4	Emily Lee	3	Los Angeles
5	5	David Chen	7	Houston
6	6	Karen Wong	15	Chicago
7	7	abhi	2000	Chicago

	DEVELOPER_ID	EMAIL
1	1	john@example.com
2	2	mary@example.com
3	3	alex@example.com
4	4	emily@example.com
5	5	david@example.com
6	6	karen@example.com
7	7	abhi@example.com

	EMAIL	PUBLISHER_ID
1	info@penguinbooks.com	1
2	info2@penguinbooks.com	1
3	contact@harpercollins.com	2
4	info@randomhouse.com	3
5	info2@randomhouse.com	3
6	contact@hachette-livre.com	4
7	info@macmillanpublishers.com	5
8	contact@simonandschuster.com	6

	PUBLISHER_ID	PUBLISHER_NAME	NO_OF_BOOKS_PUBLISHED	CITY
1	1	Penguin Books	500	New York City
2	2	HarperCollins	700	Houston
3	3	Random House	800	New York City
4	4	Hachette Livre	600	Mumbai
5	5	Macmillan Publishers	400	Houston
6	6	SimonSchuster	450	New York City

	APP_ID	APP_NAME	CATEGORY_ID	DEVELOPER_ID
1	1	Angry Birds	1	1
2	2	Candy Crush	2	2
3	3	FIFA	3	3
4	4	Civilization	4	4
5	5	Minecraft	5	5
6	6	Facebook	6	5
7	7	Instagram	6	6
8	8	Twitter	6	6
9	9	Microsoft Word	7	5
10	10	Netflix	8	5
11	11	YouTube	8	5
12	12	The New York Times	9	6
13	13	Duolingo	10	6
14	14	Khan Academy	10	6
15	15	Quizlet	10	6
16	16	Gatsby	11	7
17	17	Mockingbird	11	7
18	18	Da Vinci	11	7
19	19	PridePrejudice	11	7
20	20	Dune	11	7
21	21	1984	11	7
22	22	GirlDragonTattoo	11	7
23	23	Outlander	11	7
24	24	Ender's	11	7
25	25	Catcher Rye	11	7

	EMAIL_ID	USER_NAME	PASSWORD
1	john.doe@gmail.com	John Doe	password123
2	jane.smith@yahoo.com	Jane Smith	abc123
3	susan.wilson@hotmail.com	Susan Wilson	ilovecats
4	peter.jackson@gmail.com	Peter Jackson	lotr123
5	mary.johnson@gmail.com	Mary Johnson	password456
6	bob.dylan@yahoo.com	Bob Dylan	blowinthewind
7	lisa.anderson@gmail.com	Lisa Anderson	letitbe
8	michael.nguyen@hotmail.com	Michael Nguyen	coffee123
9	emily.roberts@yahoo.com	Emily Roberts	happylife
10	jason.miller@gmail.com	Jason Miller	qwerty123

	UPDATE_ID	DESCRIPTION	UPDATE_SIZE	BEFORE_VERSION	AFTER_VERSION	APP_UPDATE_TIME	APP_ID
1	1	Bug fixes and performance improvements	10	1.2.3	1.2.4	14-02-23 9:00:00.000000000 AM	1
2	2	New levels and challenges	50	2.5.1	2.6.0	14-05-23 10:00:00.000000000 AM	2
3	3	New team kits and player updates	20	1.1.2	1.2.0	14-05-23 11:00:00.000000000 AM	3
4	4	New civilizations and wonders	80	4.2.1	4.3.0	14-05-23 12:00:00.000000000 PM	4
5	5	New mobs and biomes	60	1.1.1	1.2.0	14-05-23 1:00:00.000000000 PM	5
6	6	Bug fixes and performance improvements	15	10.5.3	10.5.4	14-05-23 2:00:00.000000000 PM	6
7	7	New filters and effects	30	2.3.1	2.4.0	14-05-23 3:00:00.000000000 PM	7
8	8	New features and bug fixes	40	6.1.1	6.2.0	14-05-23 4:00:00.000000000 PM	8
9	9	New templates and design options	25	3.0.0	3.0.1	14-05-23 5:00:00.000000000 PM	9

	UPDATE_ID	EMAIL_ID	USER_UPDATED_TIME
1	1	john.doe@gmail.com	14-05-23 9:30:00.000000000 AM
2	2	jane.smith@yahoo.com	14-05-23 10:15:00.000000000 AM
3	3	susan.wilson@hotmail.com	14-05-23 11:45:00.000000000 AM
4	4	peter.jackson@gmail.com	14-05-23 12:30:00.000000000 PM
5	5	mary.johnson@gmail.com	14-05-23 1:15:00.000000000 PM
6	6	bob.dylan@yahoo.com	14-05-23 2:00:00.000000000 PM
7	7	lisa.anderson@gmail.com	14-05-23 3:45:00.000000000 PM
8	8	michael.nguyen@hotmail.com	14-05-23 4:30:00.000000000 PM
9	9	emily.roberts@yahoo.com	14-05-23 5:15:00.000000000 PM
10	1	jason.miller@gmail.com	14-05-23 6:00:00.000000000 PM

	RATING_ID	APP_ID	RATING_VALUE	REVIEW_DESCRIPTION	TIME_STAMP	EMAIL_ID
1	1	1	4	Fun game!	2023-05-13 13:00:00	john.doe@gmail.com
2	2	1	3	Too many ads	2023-05-13 14:00:00	jane.smith@yahoo.com
3	1	2	5	Addictive!	2023-05-13 15:00:00	susan.wilson@hotmail.com
4	2	2	2	Boring after a while	2023-05-13 16:00:00	peter.jackson@gmail.com
5	1	3	4	Great soccer game	2023-05-13 17:00:00	mary.johnson@gmail.com
6	2	3	1	Worst game ever	2023-05-13 18:00:00	bob.dylan@yahoo.com
7	1	4	5	Love this strategy game	2023-05-13 19:00:00	lisa.anderson@gmail.com
8	2	4	4	Great graphics	2023-05-13 20:00:00	michael.nguyen@hotmail.com
9	1	5	3	Not my thing	2023-05-13 21:00:00	emily.roberts@yahoo.com
10	2	5	5	Amazing!	2023-05-13 22:00:00	jason.miller@gmail.com

	SUBSCRIPTION_TYPE	AMOUNT	TIME_STAMP	APP_ID
1	1	10	14-05-23 2:30:00.000000000 PM	1
2	2	50	13-05-23 12:00:00.000000000 PM	2
3	3	20	12-05-23 4:45:00.000000000 PM	3
4	4	150	11-05-23 10:15:00.000000000 AM	6
5	5	80	10-05-23 9:30:00.000000000 AM	7

	EMAIL_ID	SUBSCRIPTION_TYPE	SUBSCRIPTION_START_DATE	SUBSCRIPTION_END_DATE
1	john.doe@gmail.com	1	14-05-23 2:30:00.000000000 PM	14-05-24 2:30:00.000000000 PM
2	jane.smith@yahoo.com	2	13-05-23 12:00:00.000000000 PM	13-05-24 12:00:00.000000000 PM
3	susan.wilson@hotmail.com	3	12-05-23 4:45:00.000000000 PM	12-05-24 4:45:00.000000000 PM
4	peter.jackson@gmail.com	4	11-05-23 10:15:00.000000000 AM	11-05-24 10:15:00.000000000 AM
5	mary.johnson@gmail.com	5	10-05-23 9:30:00.000000000 AM	10-05-24 9:30:00.000000000 AM

	BOOK_ID	NO_OF_PAGES	AUTHOR_NAME	PUBLISHER_ID	GENRE_ID	APP_ID
1	1	300	J.K. Rowling	1	1	16
2	2	250	Stephen King	2	1	17
3	3	400	Dan Brown	3	2	18
4	4	350	Jane Austen	1	1	19
5	5	600	Frank Herbert	4	1	20
6	6	200	George Orwell	5	2	21
7	7	450	Stieg Larsson	6	3	22
8	8	550	Diana Gabaldon	2	3	23
9	9	400	Orson Scott Card	6	1	24
10	10	300	J.D. Salinger	1	4	25

	GENRE_ID	TYPE	NAME
1	1	E-book	Fiction
2	2	E-book	Non-fiction
3	3	Audio-book	Fiction
4	4	Audio-book	Non-fiction

	BOOK_ID	EMAIL_ID
1	1	john.doe@gmail.com
2	2	jane.smith@yahoo.com
3	3	susan.wilson@hotmail.com

	EMAIL_ID	APP_ID	TIME_STAMP
1	john.doe@gmail.com	1	2023-05-13 12:30:00
2	jane.smith@yahoo.com	1	2023-05-13 13:45:00
3	susan.wilson@hotmail.com	2	2023-05-13 10:15:00
4	peter.jackson@gmail.com	2	2023-05-13 11:20:00
5	mary.johnson@gmail.com	3	2023-05-13 14:00:00
6	bob.dylan@yahoo.com	3	2023-05-13 15:30:00
7	lisa.anderson@gmail.com	4	2023-05-13 16:45:00
8	michael.nguyen@hotmail.com	5	2023-05-13 17:00:00
9	emily.roberts@yahoo.com	5	2023-05-13 18:15:00
10	jason.miller@gmail.com	6	2023-05-13 19:30:00
11	jane.smith@yahoo.com	6	2023-05-13 20:45:00
12	susan.wilson@hotmail.com	7	2023-05-13 21:15:00
13	peter.jackson@gmail.com	8	2023-05-13 22:20:00
14	mary.johnson@gmail.com	8	2023-05-13 23:00:00
15	bob.dylan@yahoo.com	10	2023-05-13 23:30:00

	RANKING	APP_ID	APP_SIZE	PRICE	AGE_RESTRICTION	DESCRIPTION
1	1	1	100	0	4+	The classic bird-flinging game that started it all.
2	2	2	150	0	4+	Match candies to progress through levels.
3	3	3	2000	5	4+	Play with your favorite soccer teams and compete in tournaments.
4	4	4	500	20	12+	Build and lead a civilization from ancient times to the modern era.
5	5	5	300	7	7+	Explore and build in a blocky 3D world.
6	6	6	300	0	12+	Connect with friends and family and discover new things.
7	7	7	350	0	12+	Share photos and videos with your followers.
8	8	8	400	0	17+	Join the conversation and get real-time updates on news and events.
9	9	9	600	150	4+	Create and edit documents and collaborate with others.
10	10	10	700	10	17+	Watch TV shows and movies on your device.
11	11	11	750	0	17+	Discover, watch, and share videos.
12	12	12	800	0	12+	Stay informed with breaking news and top stories.
13	13	13	900	0	4+	Learn a new language for free.
14	14	14	1000	0	4+	Get free lessons and practice exercises in math, science, and more.
15	15	15	1000	0	4+	take quizzes
16	1	16	25	5	No age restriction	E-book version of The Great Gatsby
17	2	17	35	6	No age restriction	Audiobook version of To Kill a Mockingbird
18	3	18	28	4	No age restriction	E-book version of The Da Vinci Code
19	4	19	20	3	No age restriction	E-book version of Pride and Prejudice
20	5	20	50	7	No age restriction	Audiobook version of Dune
21	6	21	30	5	No age restriction	E-book version of 1984
22	7	22	40	6	No age restriction	Audiobook version of The Girl with the Dragon Tattoo
23	8	23	33	5	No age restriction	E-book version of Outlander
24	9	24	55	7	No age restriction	Audiobook version of Ender’s Game
25	10	25	22	4	No age restriction	E-book version of The Catcher in the Rye

	RANKING	APP_ID	SUBSCRIPTIONS
1	1	1	Angry Birds Plus
2	2	2	Candy Crush Plus
3	3	3	FIFA VIP
4	6	6	Facebook Plus
5	6	6	Facebook Premium
6	7	7	Instagram Pro
7	8	8	Twitter Gold
8	8	8	Twitter Plus
9	9	9	Microsoft Word Pro
10	10	10	Netflix Plus
11	10	10	Netflix Premium
12	11	11	YouTube Premium
13	13	13	Duolingo Pro
14	15	15	Quizlet Plus
15	15	15	Quizlet Pro

	LANGUAGES	RANKING	APP_ID
1	Chinese	8	8
2	English	1	1
3	English	3	3
4	English	5	5
5	English	7	7
6	English	9	9
7	English	10	10
8	English	11	11
9	English	13	13
10	English	14	14
11	English	15	15
12	French	4	4
13	German	10	10
14	Japanese	2	2
15	Spanish	2	2
16	Spanish	6	6
17	Spanish	12	12
18	Spanish	14	14
19	Spanish	15	15

	CATEGORY_ID	TYPE	NAME
1	1	Game	Action
2	2	Game	Puzzle
3	3	Game	Sports
4	4	Game	Strategy
5	5	Game	Adventure
6	6	General	Social
7	7	General	Productivity
8	8	General	Entertainment
9	9	General	News
10	10	General	Education
11	11	BOOKS	BOOKS

	CITY	STATE
1	San Francisco	California
2	New York	New York
3	Seattle	Washington
4	Los Angeles	California
5	Houston	Texas
6	Chicago	Illinois
7	New York City	New York
8	Mumbai	Maharashtra
9	São Paulo	São Paulo
10	Toronto	Ontario

	STATE	COUNTRY
1	California	United States
2	New York	United States
3	Washington	United States
4	Texas	United States
5	Illinois	United States
6	Florida	United States
7	Maharashtra	India
8	São Paulo	Brazil
9	Ontario	Canada

QUERIES:

Queries:

1.Display the book name, author name,

No of pages which are of ‘audio-book’ format.

Code:

```
select app_name as book_name,author_name,no_of_pages
from application,
(select app_id,author_name,no_of_pages from books
where genre_id in
(select genre_id from genre where type='Audio-book'))t
where application.app_id=t.app_id;
```

	BOOK_NAME	AUTHOR_NAME	NO_OF_PAGES
1	GirlDragonTattoo	Stieg Larsson	450
2	Outlander	Diana Gabaldon	550
3	Catcher Rye	J.D. Salinger	300

2.Display all the users and their downloaded apps.

```
select user_name,app_name as downloaded_apps from
application,
(select user_name,app_id  from user_table,downloads
where user_table.email_id=downloads.email_id)t
where t.app_id=application.app_id;
```

	USER_NAME	DOWNLOADED_APPS
1	John Doe	Angry Birds
2	Jane Smith	Angry Birds
3	Jane Smith	Facebook
4	Susan Wilson	Candy Crush
5	Susan Wilson	Instagram
6	Peter Jackson	Candy Crush
7	Peter Jackson	Twitter
8	Mary Johnson	FIFA
9	Mary Johnson	Twitter
10	Bob Dylan	FIFA
11	Bob Dylan	Netflix
12	Lisa Anderson	Civilization
13	Michael Nguyen	Minecraft
14	Emily Roberts	Minecraft
15	Jason Miller	Facebook

3.Display the details of app(id,name,rating_value,developer) which have received atleast one 5 rating.

```
select f.app_id,f.app_name,developer.developer_id,developer_name from developer, (select application.app_id,app_name,developer_id from application, (select distinct app_id from rating_reviews where rating_value = 5)t where t.app_id=application.app_id)f where f.developer_id=developer.developer_id;
```

	APP_ID	APP_NAME	DEVELOPER_ID	DEVELOPER_NAME
1	2	Candy Crush	2	Mary Johnson
2	4	Civilization	4	Emily Lee
3	5	Minecraft	5	David Chen