The objective of this project is to:

- Design and implement a relational database schema for an app store.
- Simulate real-world use cases like trending apps, user activity, app ratings, and category-wise distribution.
- Write meaningful and optimized SQL queries to replicate actual store functionalities.

Database Schema Overview

The project includes the following tables or entities:

ENTITES :

USERS

DEVELOPER

APP

APP_VERSION

PERMISSION

APP_PERMISSIN

CATEGORY

APP_CATEGORY

REVIEW

DOWNLOAD

WISHLIST

ASSUMPTIONS :

- One app may belong to multiple categories .
- A developer can publish multiple apps.
- An app can request multiple permissions.
- Each download record includes a timestamp and network type.
- Review ratings are between 0.0 and 5.0.
- Each app version can have only one developer.

What is Normalization?

Normalization is a database design technique used to reduce data redundancy and improve data integrity by organizing data into multiple related tables. The goal is to eliminate update anomalies and ensure efficient data organization.

All your tables are well-structured with respect to **keys and dependencies**, and satisfy the

Boyce-Codd Normal Form.

In the **AppHub** database design, **data redundancy is minimized** through proper normalization and the use of **associative tables** (like APP_CATEGORY, APP_PERMISSION, and WISHLIST) to manage many-to-many relationships. By ensuring all tables follow **BCNF**, we eliminate unnecessary data duplication. As a result, the schema avoids **insertion**, **update**, and **deletion anomalies**, preserving both **data integrity** and **consistency** across the system.

How to Use :

- Clone git repository link is https://github.com/Chandaka-Preetesh/DATABASE_PROJECT.git .
- Use Oracle SQL (or compatible SQL DBMS).
- Run the schema creation scripts to build tables.
- Insert sample data as provided.
- Try Executing SQL queries .