Database Implementation

1. **Create Table with Check Constraints (1_Create_Table_with_Check_Constraints.sql):**
    - This script creates a database named marketplace.
    - It defines a STUDENT, ACCOUNT, and ADMIN tables.
    - There are check constraints on fields in all the three tables to ensure data integrity.
    - The relationships between the STUDENT, ACCOUNT, and ADMIN tables are established through foreign keys.
2. **Stored Procedures (2_Stored_Procedures.sql):**

    This script includes 3 stored procecures:

    - GetReviewsonItem: Retrieves reviews for a specific listing.
    - AddtoCart: Adds an item to the cart if it is available in the live listings.
    - GetSellerStats: Gets a sellers average rating and
3. **Table Views (3_Table_Views.sql):**

    This script has a total of 6 views:

    - reported_message_view: To display reported messages made by users across listings, for moderating user interactions and ensuring a safe platform environment.
    - saved_item_reviews_view: To show reviews on items that have been added to the cart or saved, which can help users in making informed decisions.
    - recently_posted_items_view: To list recently posted items on the marketplace, aiding users in quickly finding new listings.
    - items_by_seller_view: To count the number of items sold by each seller, useful for analyzing seller performance.
    - avg_item_price_by_category_view: To provide an average price of items within each category.
    - active_sellers_view: To identify active sellers on the platform, based on criteria like the number of listings, sales volume, or recent activity.
4. **DML Trigger (4_DML_Trigger.sql):**
    - This script defines a DML trigger named update_supervisor_trigger on the TRANSACTION table.
    - This trigger is designed to automatically assign a supervisor to each new transaction added to the TRANSACTION table.
5. **Computed Columns and User-Defined Functions (5_Computed_Columns_UDF.sql):**
    - It defines two functions, CalculateAge and CalculateDiscount for age calculation and discount percentage calculation, respectively.
6. **Data Encryption (6_Data_Encryption.sql):**

- The script deals with creating a master key and a certificate for data encryption.
- It involves creating symmetric keys and certificates for securing data in the **marketplace** database.

7. **Non-Clustered Indexes (7_Non_Clustered_Indexes.sql):**
    - Non-clustered indexes in a database serve to enhance query performance by providing an alternative structure for efficient data retrieval. This enables faster access to specific rows based on indexed columns, particularly beneficial for read-heavy workloads and queries.
    - This database contains 5 non-clustered indexes

Graphical User Interface

How to run: streamlit run .\app.py

1. **Database Connection:**
    - Establishes a connection to a SQL Server database using **pyodbc.connect**.
2. **Functions Defined:**
    - read_data(table): Reads data from a specified table and returns it as a Pandas DataFrame.
    - insert_student(): Inserts a new student record into the STUDENT table.
    - insert_item(): Inserts a new item record into the ITEM table, with validation for the item condition.
    - insert_listing(): Inserts a new listing record into the LISTING table.
    - update_data(): Updates a specified column in a table for a given primary key.
    - delete_data(): Deletes a record from a specified table based on the primary key.