Project Description:

Part 1: Views and Indexes

1. A view is a stored query that has been given a name and virtually saved in the database. Unlike ordinary tables in a relational database, a view is the dynamic result of one or more relational operations and it does not actually exist in the database but is produced upon request by a particular user, at the time of request.

   a. List the advantages of views over ordinary tables (base tables).

   b. Create the following three views (using the suggested names) for the wholesale database.

      i. **ProductPricing**: Join the Product and Supply table using the product_id and display the values for the following variables: product_id, vendor_id, name, production_cost, selling_price

      ii. **VendorBrand**: Create a cumulative sum of the selling price of the products and group them by their brand and vendor id, the View should contain the following variables: vendor_id, brand_id, Sum (selling_price).

      iii. **CostCategory**: Calculate the average selling price for the product and group them by their category. The View should contain the following variables: category_id, category_name, Avg(selling_price).

2. Databases usually contain large amounts of data and a DBMS is required to read data from disk whenever a query is executed. A database index is a data structure that improves the speed of data retrieval operations by enabling the DBMS to read only a subset of all data from disk when answering a query.

   a. Create an index for the **Vendor** table on the email attribute in the wholesale database and provide a point query for which the index is useful.

   b. Create an index for the **Supply** table on the selling_price attribute in the wholesale database and provide a range query for which the index is useful.

Part 2: Stored Procedure
1. Write a stored procedure for the wholesale database to output various statistics (mentioned below) for a Vendor. The procedure will be named `GetVendorStat`, will accept a vendor ID and generate the following statistics:
   - The Vendor name and category
   - The number of distinct brands this Vendor sells products for.
   - Using the `ProductPricing` view, for all the products sold by this Vendor, list the percentage profit (i.e. calculate the percentage difference between the `production_cost` and `selling_price`).
   - Using the `VendorBrand` view, list the name of the brand with the highest selling price for this Vendor.
   - Using the `CostCategory` view, list the average selling price for the Vendor's category.

2. Create a second stored procedure, `GetAllVendorsStats`, which will call `GetVendorStat` and generate the statistics for all the Vendors in the wholesale database.