Assignment #3 Database Building a Custom Data Model

The class selects a business to model:

- Run the business
- Compute Revenue
- Compute Costs
- Track History

Turn in the relationship diagram from DB Visualizer as a PDF / High resolution JPG and all of the create statements. Make sure to add the necessary Primary and Foreign Keys.

Solution:-

Here in this assignment, as a first step we look for all the tables that are independent and do not require any foreign keys, like store, customer, product. But product is mostly associated with a brand, thus a brand name needs to be added with a product, now it is not an independent table. When we investigate further, then we find that if multiple phone numbers are present for the same customer, then table needs to be divided and foreign key comes into picture. It mainly depends on the data we want to store in the table, or in other words we can say that, more the details, the more the complexity of the data model.

To make the things more complex, type of phone like home, office/work, cell/mobile, fax can be considered. The complications goes on if a customer have multiple phones of the same type, say two home phone numbers. Another question arises here is that whether the phone type can be repeated in the table, the answer is that it can be repeated but there is no real reason to do that.

Some other questions like can the name of a person repeat, the answer is yes since there is a unique customer ID that acts as a primary key. But making person name as a primary key is a big mistake, since names often repeat.

Now lets investigate product table further. A Product can have multiple categories and multiple sub categories. If we store product and categories/subcategories in the same table, then many null values will be present, which violates 3rd normal form.

We choose a **Retail Store as our business model**, where store sells various products under different categories and subcategories to customers. Description of each table is as under:-

a) The `Store` table represents stores with details such as storeName, streetAddress, city, state, zip, and storeNumber.
b) The `Product` table represents products with details such as ProductName, IDbrand (foreign key referencing Brand), ManufacturerPartNumber, SKU, cost, price, and ICategory (foreign key referencing Category).
c) The `Customer` table represents customers with details such as Firstname and LastName.
d) The `CustomerPhone` table represents a many-to-many relationship between customers and phone types (stored in `RefPhone`). It uses composite primary key (IDPhoneType, IDCustomer) to manage the relationship.
e) The `RefPhone` table stores phone types with details such as PhoneType.
f) The `Brand` table represents brands with details such as BrandName.
g) The `Category` table represents categories with details such as CategoryName.
h) The `SubCategory` table represents subcategories with details such as SubCategoryName and IDCategory (foreign key referencing Category).
i) The `Stock` table represents the stock of products in stores, with details such as IDstore (foreign key referencing Store), IDproduct (foreign key referencing Product), and itemscount.
j) The ‘Transactions’ table keeps record of all the items sold where we can calculate get revenue

Following are the SQL statements that were used to create table.

Create Store table

```
CREATE TABLE Store (
    IDstore INT PRIMARY KEY,
    storeName VARCHAR(255) NOT NULL,
    streetAddress VARCHAR(255),
    city VARCHAR(255),
    state VARCHAR(255),
    zip VARCHAR(10),
    storeNumber INT
);
```

Create Product table

```
CREATE TABLE Product (
    IDproduct INT PRIMARY KEY,
    ProductName VARCHAR(255) NOT NULL,
    IDbrand INT,
    ManufacturerPartNumber VARCHAR(255),
    SKU VARCHAR(255),
    cost DECIMAL(10, 2),
    price DECIMAL(10, 2),
    ICategory INT,
    FOREIGN KEY (IDbrand) REFERENCES Brand(IDBrand),
    FOREIGN KEY (ICategory) REFERENCES Category(IDCategory)
);
```

Create Customer table

```
CREATE TABLE Customer (
    IDcustomer INT PRIMARY KEY,
    Firstname VARCHAR(255) NOT NULL,
    LastName VARCHAR(255) NOT NULL
);
```

Create CustomerPhone table

```
CREATE TABLE CustomerPhone (
    IDPhoneType INT,
    IDCustomer INT,
    PRIMARY KEY (IDPhoneType, IDCustomer),
    FOREIGN KEY (IDPhoneType) REFERENCES RefPhone(IDPhoneType),
    FOREIGN KEY (IDCustomer) REFERENCES Customer(IDcustomer)
);
```

Create RefPhone table

```
CREATE TABLE RefPhone (
    IDPhoneType INT PRIMARY KEY,
    PhoneType VARCHAR(255) NOT NULL
);
```

Create Brand table

```
CREATE TABLE Brand (
    IDBrand INT PRIMARY KEY,
    BrandName VARCHAR(255) NOT NULL
);
```

Create Category table

```
CREATE TABLE Category (
    IDCategory INT PRIMARY KEY,
    CategoryName VARCHAR(255) NOT NULL
);
```

Create SubCategory table

```
CREATE TABLE SubCategory (
    IdSubcat INT PRIMARY KEY,
    IDCategory INT,
    SubCategoryName VARCHAR(255) NOT NULL,
    FOREIGN KEY (IDCategory) REFERENCES Category(IDCategory)
);
```

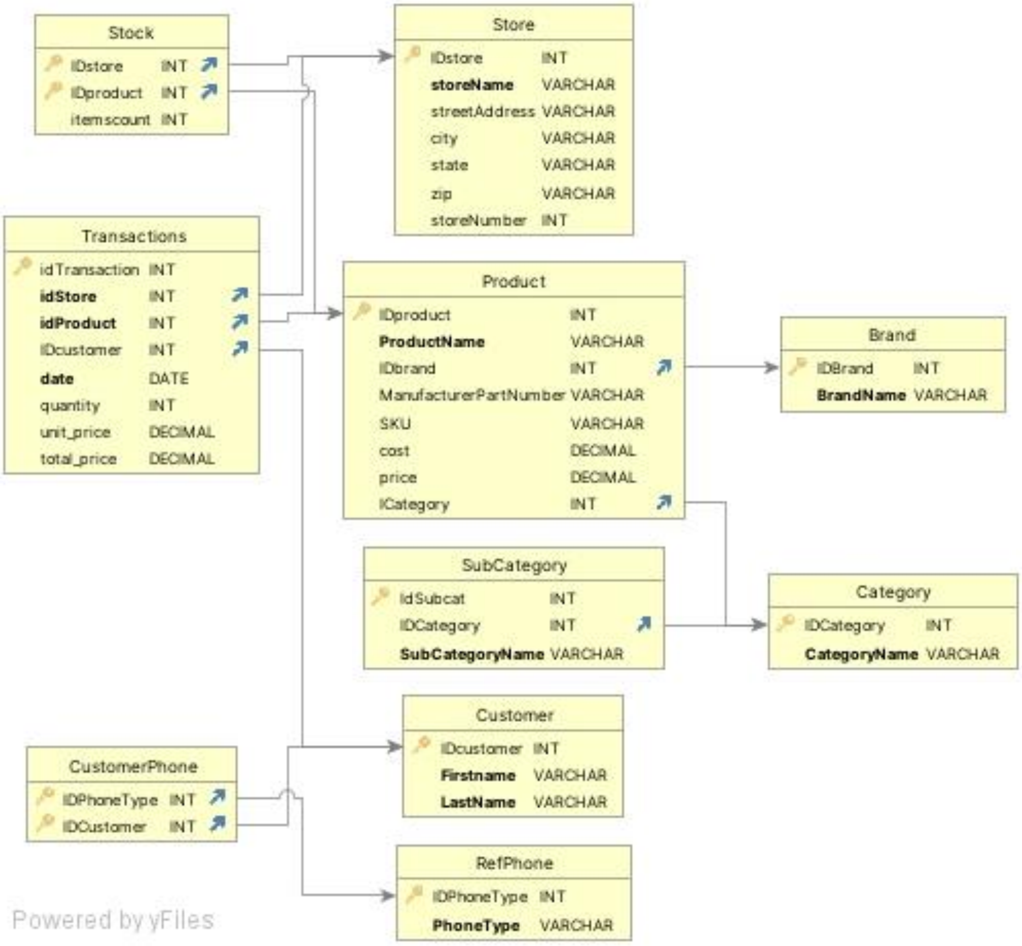
Create Stock table

```
CREATE TABLE Stock (
    IDstore INT,
    IDproduct INT,
    itemscount INT,
    PRIMARY KEY (IDstore, IDproduct),
    FOREIGN KEY (IDstore) REFERENCES Store(IDstore),
    FOREIGN KEY (IDproduct) REFERENCES Product(IDproduct)
);
```

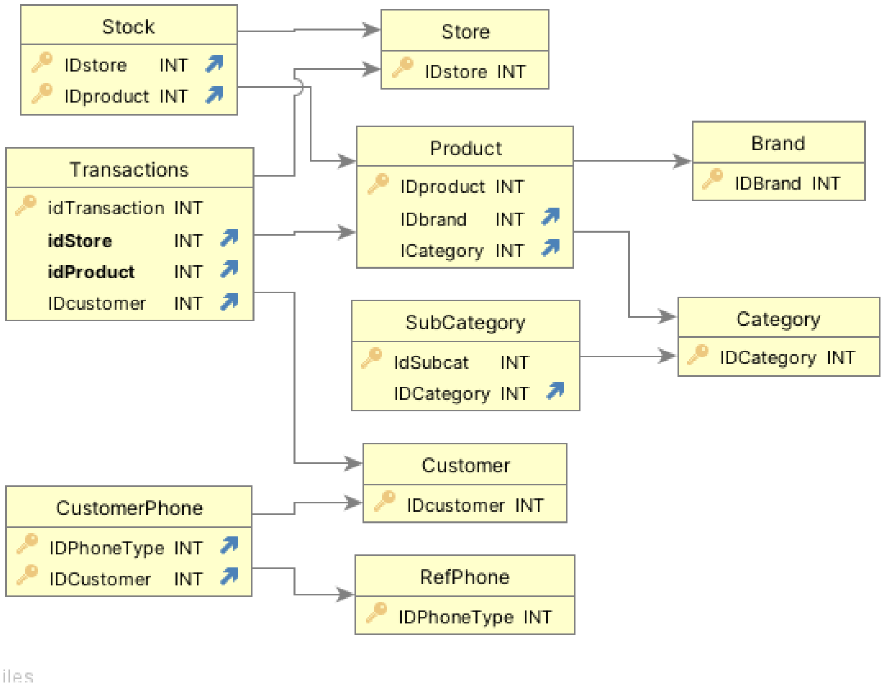
Create Transactions table

```
CREATE TABLE Transactions (
  idTransaction INT PRIMARY KEY,
  idStore INT NOT NULL,
  idProduct INT NOT NULL,
  IDcustomer INT,
  date DATE NOT NULL,
  quantity INT,
  unit_price DECIMAL(10,2),
  total_price DECIMAL(10,2) AS (quantity * unit_price),
  FOREIGN KEY (idStore) REFERENCES Store(IDstore),
  FOREIGN KEY (idProduct) REFERENCES Product(IDproduct),
  FOREIGN KEY (IDcustomer) REFERENCES Customer(IDcustomer)
);
```

Business Model Retail Store References of Tables

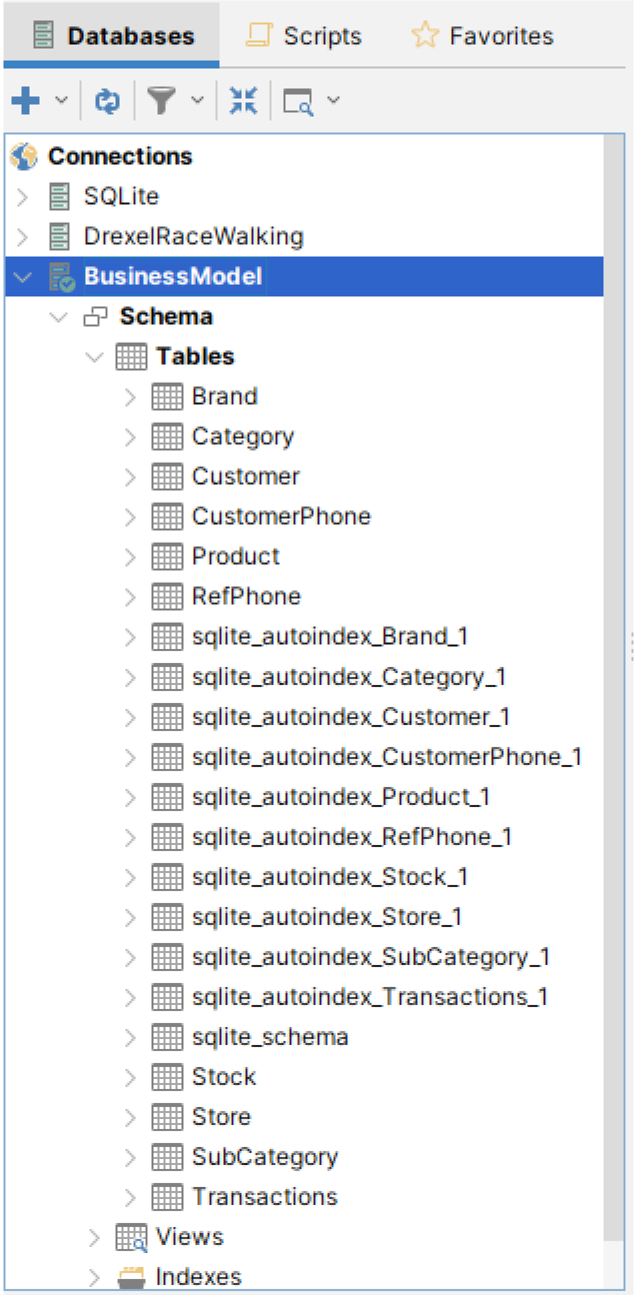


Primary key References



Database file :- BusinessModel.db

Connection and Tables



- **Connections**
  - SQLite
  - DrexelRaceWalking
  - **BusinessModel**
    - **Schema**
      - **Tables**
        - Brand
        - Category
        - Customer
        - CustomerPhone
        - Product
        - RefPhone
        - sqlite_autoindex_Brand_1
        - sqlite_autoindex_Category_1
        - sqlite_autoindex_Customer_1
        - sqlite_autoindex_CustomerPhone_1
        - sqlite_autoindex_Product_1
        - sqlite_autoindex_RefPhone_1
        - sqlite_autoindex_Stock_1
        - sqlite_autoindex_Store_1
        - sqlite_autoindex_SubCategory_1
        - sqlite_autoindex_Transactions_1
        - sqlite_schema
        - Stock
        - Store
        - SubCategory
        - Transactions
      - Views
      - Indexes