

SQL Server Assignment

Yaduttam Ganeriwala

Personnel No: 740636

Phone No. +91 8584979334

Creating the Schema

```
-- create schemas
```

```
CREATE SCHEMA production;
```

```
go
```

```
CREATE SCHEMA sales;
```

```
go
```

```
-- create tables
```

```
CREATE TABLE production.categories (  
    category_id INT IDENTITY (1, 1) PRIMARY KEY,  
    category_name VARCHAR (255) NOT NULL  
);
```

```
CREATE TABLE production.brands (  
    brand_id INT IDENTITY (1, 1) PRIMARY KEY,  
    brand_name VARCHAR (255) NOT NULL  
);
```

```
CREATE TABLE production.products (  
    product_id INT IDENTITY (1, 1) PRIMARY KEY,
```

```
product_name VARCHAR (255) NOT NULL,  
brand_id INT NOT NULL,  
category_id INT NOT NULL,  
model_year SMALLINT NOT NULL,  
list_price DECIMAL (10, 2) NOT NULL,  
FOREIGN KEY (category_id) REFERENCES production.categories (category_id) ON DELETE  
CASCADE ON UPDATE CASCADE,  
FOREIGN KEY (brand_id) REFERENCES production.brands (brand_id) ON DELETE CASCADE ON  
UPDATE CASCADE  
);
```

```
CREATE TABLE sales.customers (  
customer_id INT IDENTITY (1, 1) PRIMARY KEY,  
first_name VARCHAR (255) NOT NULL,  
last_name VARCHAR (255) NOT NULL,  
phone VARCHAR (25),  
email VARCHAR (255) NOT NULL,  
street VARCHAR (255),  
city VARCHAR (50),  
state VARCHAR (25),  
zip_code VARCHAR (5)  
);
```

```
CREATE TABLE sales.stores (  
store_id INT IDENTITY (1, 1) PRIMARY KEY,  
store_name VARCHAR (255) NOT NULL,  
phone VARCHAR (25),  
email VARCHAR (255),  
street VARCHAR (255),
```

```
    city VARCHAR (255),  
    state VARCHAR (10),  
    zip_code VARCHAR (5)  
);
```

```
CREATE TABLE sales.staffs (  
    staff_id INT IDENTITY (1, 1) PRIMARY KEY,  
    first_name VARCHAR (50) NOT NULL,  
    last_name VARCHAR (50) NOT NULL,  
    email VARCHAR (255) NOT NULL UNIQUE,  
    phone VARCHAR (25),  
    active tinyint NOT NULL,  
    store_id INT NOT NULL,  
    manager_id INT,  
    FOREIGN KEY (store_id) REFERENCES sales.stores (store_id) ON DELETE CASCADE ON UPDATE  
    CASCADE,  
    FOREIGN KEY (manager_id) REFERENCES sales.staffs (staff_id) ON DELETE NO ACTION ON  
    UPDATE NO ACTION  
);
```

```
CREATE TABLE sales.orders (  
    order_id INT IDENTITY (1, 1) PRIMARY KEY,  
    customer_id INT,  
    order_status tinyint NOT NULL,  
    -- Order status: 1 = Pending; 2 = Processing; 3 = Rejected; 4 = Completed  
    order_date DATE NOT NULL,  
    required_date DATE NOT NULL,  
    shipped_date DATE,  
    store_id INT NOT NULL,
```

```
        staff_id INT NOT NULL,

        FOREIGN KEY (customer_id) REFERENCES sales.customers (customer_id) ON DELETE CASCADE
ON UPDATE CASCADE,

        FOREIGN KEY (store_id) REFERENCES sales.stores (store_id) ON DELETE CASCADE ON UPDATE
CASCADE,

        FOREIGN KEY (staff_id) REFERENCES sales.staffs (staff_id) ON DELETE NO ACTION ON UPDATE
NO ACTION

);
```

```
CREATE TABLE sales.order_items (

    order_id INT,

    item_id INT,

    product_id INT NOT NULL,

    quantity INT NOT NULL,

    list_price DECIMAL (10, 2) NOT NULL,

    discount DECIMAL (4, 2) NOT NULL DEFAULT 0,

    PRIMARY KEY (order_id, item_id),

    FOREIGN KEY (order_id) REFERENCES sales.orders (order_id) ON DELETE CASCADE ON UPDATE
CASCADE,

    FOREIGN KEY (product_id) REFERENCES production.products (product_id) ON DELETE CASCADE
ON UPDATE CASCADE

);
```

```
CREATE TABLE production.stocks (

    store_id INT,

    product_id INT,

    quantity INT,

    PRIMARY KEY (store_id, product_id),

    FOREIGN KEY (store_id) REFERENCES sales.stores (store_id) ON DELETE CASCADE ON UPDATE
CASCADE,
```

FOREIGN KEY (product_id) REFERENCES production.products (product_id) ON DELETE CASCADE
ON UPDATE CASCADE

);

The resulting Schema is as follows:

