ASSIGNMENT 3 **Olga Chumak**

Scenario. Database for a small store.

Table structure

Consists of six tables

1. Suppliers. Includes information about suppliers: **company code, name, telephone, city.**

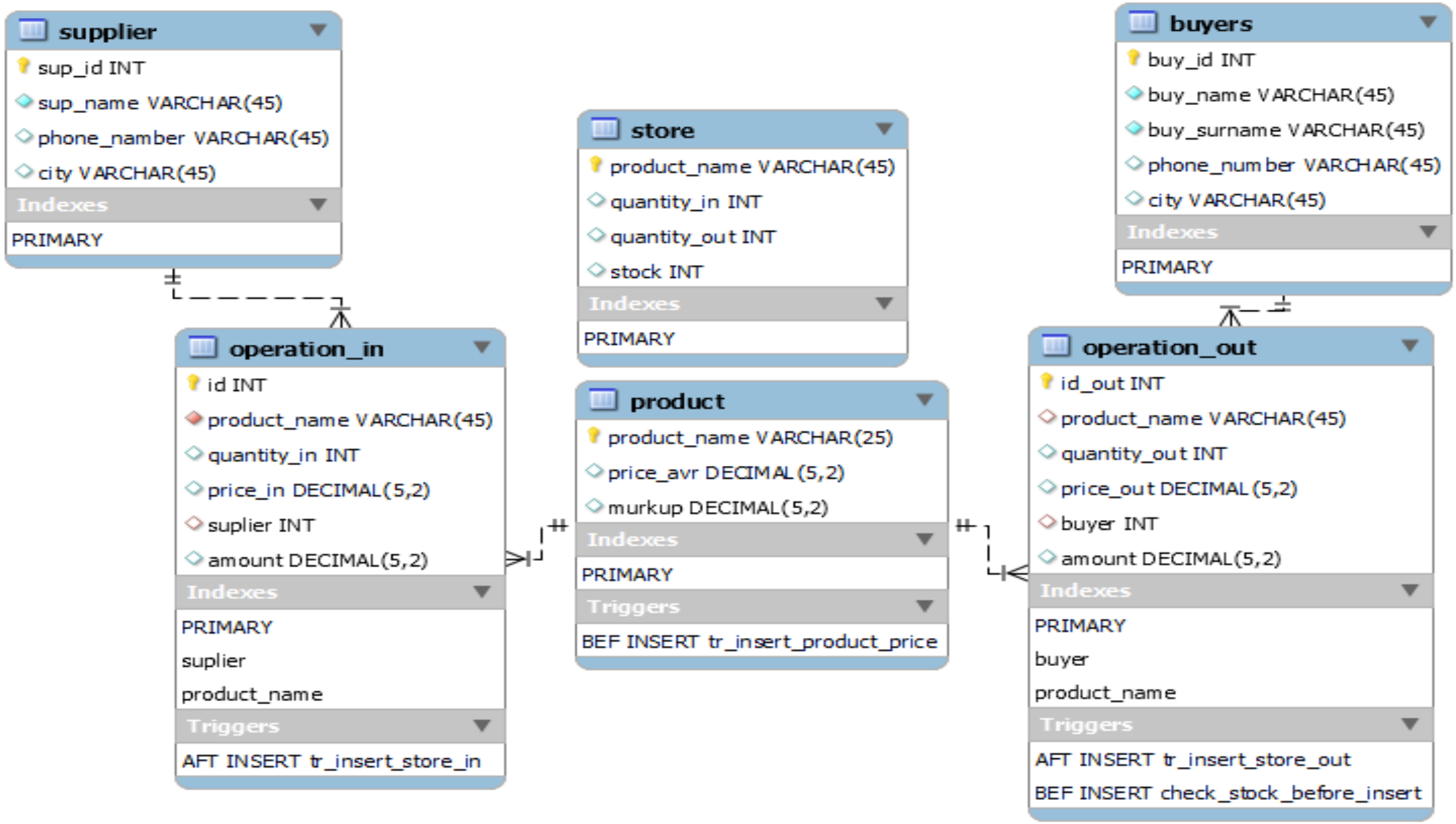
2. Buyers. Includes information about suppliers**: code, first name, last name, telephone, city.**

3. Operation_in. A table in which deliveries are recorded**. Company, product, quantity, price, amount (***the amount is calculated automatically, we set it up when we create the table***))**

4. Operation_out. A table in which sales are recorded**. Buyer, product, quantity, price** *(filled in automatically using a trigger that fires when data is inserted into the table. Calculates the average delivery price of the product and adds a markup)***, amount** *(amount is calculated automatically, set when creating the table)*

5. Product. Contains information about the product**. Name** *(unique and is a foreing key for other tables)***. Average incoming price of goods***(Average incoming price of the product (filled in automatically using a trigger that fires after inserting information into the incoming records Operation_out).* **Murkup**,trade margin *(set manually).*

6. Store. Includes warehouse information. **Product Name**.**Quantity_in** - incoming quantity of goods (*filled in automatically using a trigger that fires after inserting information into the table Operation_in).* **Quantity_out -** incoming quantity of goods (*filled in automatically using a trigger that fires after inserting information into the table Operation_out).* **STOK -** Information about stock balances (*filled in automatically)*


supplier
sup_id INT
sup_name VARCHAR(45)
phone_namber VARCHAR(45)
city VARCHAR(45)
Indexes
PRIMARY
store
product_name VARCHAR(45)
quantity_in INT
quantity_out INT
stock INT
Indexes
PRIMARY
buyers
buy_id INT
buy_name VARCHAR(45)
buy_surname VARCHAR(45)
phone_number VARCHAR(45)
city VARCHAR(45)
Indexes
PRIMARY
operation_in
id INT
product_name VARCHAR(45)
quantity_in INT
price_in DECIMAL(5,2)
suplier INT
amount DECIMAL(5,2)
Indexes
PRIMARY
suplier
product_name
Triggers
AFT INSERT tr_insert_store_in
product
product_name VARCHAR(25)
price_avr DECIMAL(5,2)
murkup DECIMAL(5,2)
Indexes
PRIMARY
Triggers
BEF INSERT tr_insert_product_price
operation_out
id_out INT
product_name VARCHAR(45)
quantity_out INT
price_out DECIMAL(5,2)
buyer INT
amount DECIMAL(5,2)
Indexes
PRIMARY
buyer
product_name
Triggers
AFT INSERT tr_insert_store_out
BEF INSERT check_stock_before_insert

Tasks

1. Automatically fill the warehouse with information on incoming and outgoing transactions. See warehouse balances in real time. Do not allow outgoing transactions if there are not enough balances in the warehouse. In this case, there are not enough products to show the message**.**

2. Automatically determine the selling price of a product depending on the purchase price.

3. Create procedures that allow you to quickly insert data into incoming and outgoing operations

4. Create various types of reports and queries: Trade balance, customer report, warehouse report, queries that allow you to find optimal suppliers, study the geography of sales, check the database for accuracy.