Data warehouses
Lab 2

This laboratory assignment consists of 2 tasks. If you cannot solve the task, try to give at least a partial solution or justification for the reason for the lack of a solution.

SQL Queries

SQL warmup task. Please do not use the build-in query editor – focus on pure SQL statements! Prepare adequate SQL queries to:

1. Provide information about the global sales amount (money), number of orders and volume (items sold) of the AdventureWorks business.
   An example of the query result is shown in Table 1.

Sales Amount	Volume	Number of orders
…	…	…

Table 1. Result relation for Task 1.1

2. Provide information about the sales amount, volume, and number of orders in individual years of operation of the business.
   An example of the query result is shown in Table 2.

Year	Sales amount	Volume	Number of orders
…	…	…	…

Table 2. Result relation for Task 1.2

3. Prepare a SQL query that provides top 5 customers with the highest number of orders, try using the customer name (it might be tricky).
   An example of the query result is shown in Table 3.

CustomerID	Last name, name	Number of orders
…	…	…

Table 3. Result relation for Task 1.3

4. Prepare a SQL query that provides the names of all individual customers with the total sum of purchases (use SalesOrderHeader.SubTotal) greater than USD 1500 – sorted (descending) by the total sales amount.
   An example of the query result is shown in Table 4.

CustomerID	Last name, name	Sales amount
…	…	…

Table 4. Result relation for Task 1.4

5. Prepare a query that provides information about average price, total sales amount, and total volume in individual product categories of the AdventureWorks business.
   An example of the query result is shown in Table 5.

CategoryID	Category name	Average price	Total sales amount	Total volume
…	…	…	…	…

Table 5. Result relation for Task 1.5

6. Display all subcategories whose average price is higher than the average price of all categories.
   An example of the query result is shown in Table 6.

SubcategoryID	Subcategory name	Average price	Average price (over all categories)
…	…	…	…

Table 6. Result relation for Task 1.6

7. Select sales territory (name) with sales in May 2013 higher than the average monthly sales per sales territory.
   An example of the query result is shown in Table 7.

Sales territory ID	Sales territory name	Sales (May 2013)	Average monthly sales (per territory)
…	…	…	…

Table 7. Result relation for Task 1.7

8. Create a list of sales territories (ids are enough) with an average number of orders (both real value and the largest integer less than the value) made by customers who have more than 10 orders in general (use CTE).
   An example of the query result is shown in Table 8.

TerritoryID	Average number of orders	Average number of orders (INT)

...	...	...

Table 8. Result relation for Task 1.8

9. Show monthly sales amount by each sales territory in 2013 and calculate the difference with the previous month (use 0 for 12/2012) to identify trends.

An example of the query result is shown in Table 9.

TerritoryID	Sales territory name	Month sales amount	Diff to prev.
…	…	…	…

Table 9. Result relation for Task 1.9

Data Modeling

Please analyze the conceptual data model of "Orders" (Fig. 1.), which is incomplete, but the classes and the relationships between them may represent a part of the reality under consideration.

Consider a straightforward situation (intuitive understanding) of handling orders made by customers, for a set of products, and handled by different shops. Please assume that a customer can repeatedly shop in the same store, and any customer can shop in any store. Each purchase is made by the customer in the store on a specific day and time. A store must offer at least one product and each store can individually propose the price and quantity of the offered product. Finally, the same product (type) can be offered by multiple stores.
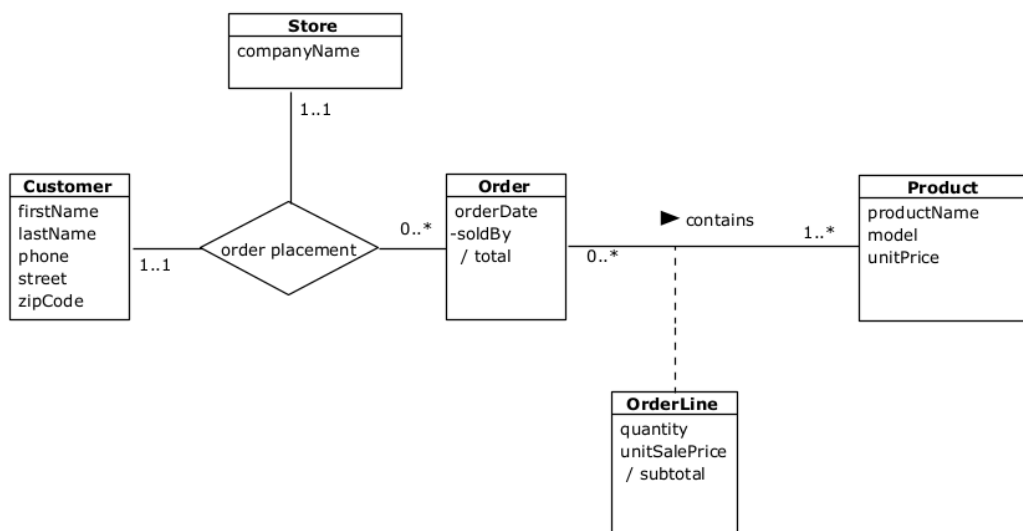

Fig. 1 Class diagram depicting conceptual data model in UML

Then, complete the following tasks. In general, modify the set of rules and constraints (by supplementing their definition) or correct the conceptual data model (diagram, justify your changes) to obtain a valid model.

1. Specify the set of rules and domain constraints. Domain rules and constraints – preliminary list (*should be revalidated and completed*):
    a. R01 – A customer can repeatedly shop in the same store
    b. R02 – Any customer can shop in any store
    c. R03 – Each purchase is made by the customer in the store on a specific day and time
    d. R04 – A store must offer at least one product
    e. R05 – The same product (type) can be offered by multiple stores
    f. R06 – Each store can individually propose the price and quantity of the offered product
    g. R07 – …
2. Provide a revised and complete version of the data model (complete UML class diagram).
3. Create a logical/physical data model as a DDL SQL script (including domain rules and constraints) while trying to comply with the SQL standard (omitting, if possible, native SQL implementation constructs). Run these in MS SQL Server – goal is to implement the database, as we need a physical data model of the modeled part of the reality.
4. Test the created database. Enter several records into each table, checking the correctness of the implementation (remember to check both correct data and inconsistent data with the applicable rules – please comment and explain the obtained messages from the DBMS system)