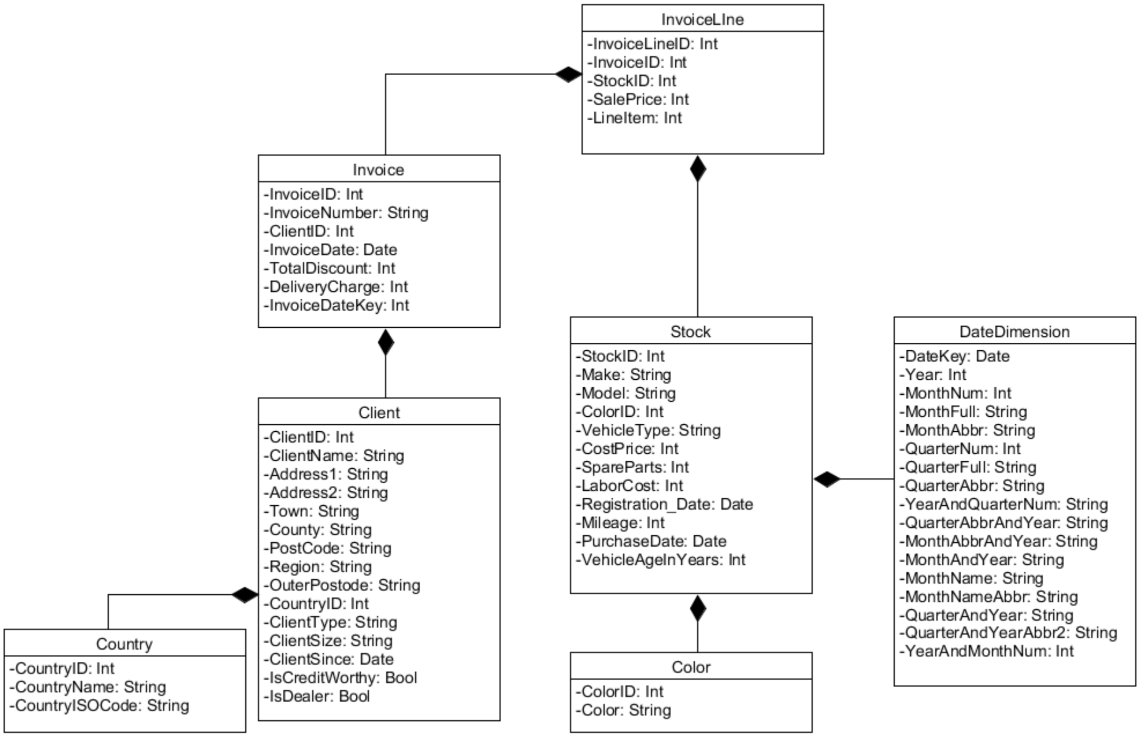
Internship program

Luis Fernando Villegas Campos

Exercise 1 – Data Modeling

My first approach with the problem and the file made me think if the file is a "database" that cannot be modified because it is already on duty, or it could be just a base of the information that the company could need for a complete development including database and programs to access the information. So, for this problem at first, I created the data model based completely from the spreadsheet.

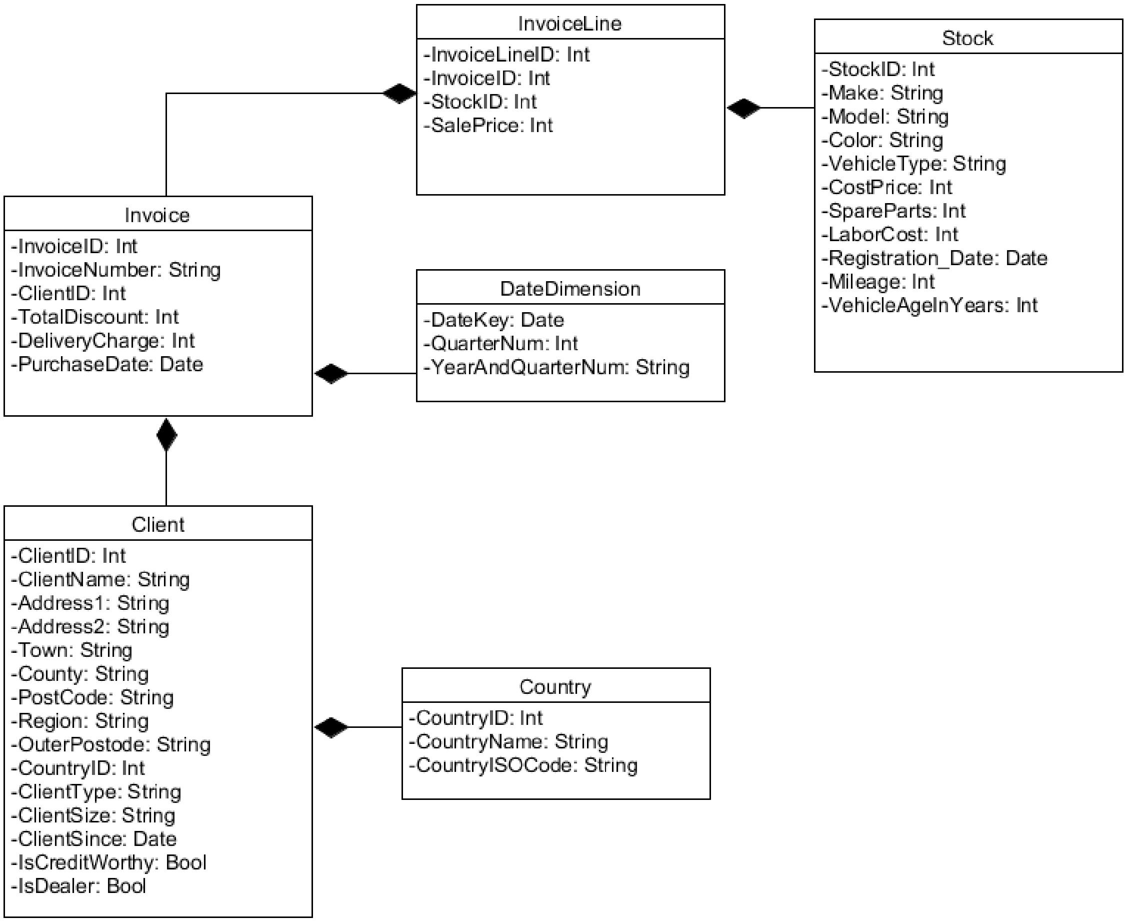


When I finished this model I saw some errors, for better understanding I will list some of them.

1. DateDimension could only be used in Stock in the attribute PurchaseDate because of the DD/MM/YYYY, so the InvoiceDateKey attribute in Invoice is doing nothing; and this table (DateDimension) its not even needed because it displays every day even if the day is not used.
2. DateDimension repeats the quarters and the years in most of the information saved, so it will be a waste of space for every day.
3. Invoice attributes such as InvoiceDateKey and InvoiceDate do not make sense, the first one doesn't work as a key for DateDimension and the second one is saved with default values of 00:00,0.

4. LineItem attribute from InvoiceLine could be delete, there is no need to save the line, simply when you retrieve the information it gets back in order of entry.
5. Color should not exist because the full table just have a key to access and one word to save, its better to have that info directly in Stock.

After the errors and comments wrote before I made another model that works better and save more space, this is when the first problem arrives, what kind of information I could move or not depends on the problem that I wrote in the first paragraph. Looking at the exercise 2 point 1 (Define the required classes, methods and properties **based** on CarSalesDataforReports Excel file.) I will assume that it cannot be modified and this exercise 1 is just to know problem solving skills.



DateDimension keeps the YearAndQuarterNum for a faster search in the lines. DateDimension also is moved from Stock to avoid duplicate dates in “n” stocks, instead is better to put a general date in the Invoice. Color was removed and even the attributes listed before.

Exercise 2 – Object Oriented Programming

- Define the required classes, methods and properties based on CarSalesDataforReports Excel file.

  For this exercise I will do the code in Java and replace the keysIDs with its respective class. Getter and setter methods are always the same, but the most interesting one is a DateDimension constructor that takes a timestamp and fill every attribute needed.

- Based on the point number one, define a class to include another kind of transportation vehicle. Include properties and methods that you consider required

  Facing this problem guides me to some solutions and problems, at first if this statement means to create an inheritance due to the problems to join this new class to the rest of the model due that Stock is an attribute in InvoiceLine. The inheritance main class could be a part of InvoiceLine and stock and the new class the "sons" of this main class.

  But if I misunderstood this and its just an interface class where I can access the information and add more vehicles, it's not that difficult, just a method that create an object Stock to add new vehicles.

  As I read from the class Stock, it has already a Vehicle type attribute, so I will assume that I need to develop a class interface that let the user add more vehicles in the stock.

Exercise 3 – Queries

For the zipped xml that is the xlsx file I used Apache POI library to read it and for the rest of the problem I continued solving it with Java code.

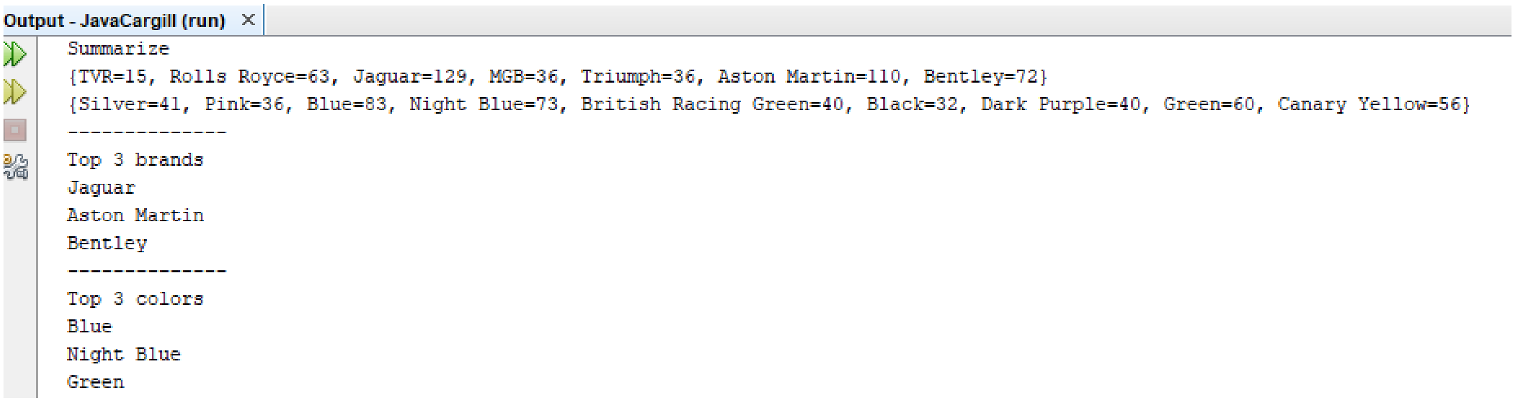
ALL THE CODE IS IN THE FILE ATTACHED

**JavaCargill is the class that has this exercise 3.

- Create a query that returns the top 3 car brands most sold (i.e., having the most car brand sales) during first and third quarter of year 2015.

  Normally is a must to search in the invoice lines to know which cars where sold and from the line get the stockID of the car to find it and don't mess with the unsold ones, but for this time there is no need to search for that because the file is a car sales report, so I could get directly to Stock sheet and read every car (proof: 462 Stocks = 462 InvoiceLines). This assumption will have the same impact for the next point.

- Create a query that shows the top 3 most sold car colors (i.e., having the most car color sales) for each quarter for the years 2012,2013,2014,2015.



```
Summarize
{TVR=15, Rolls Royce=63, Jaguar=129, MGB=36, Triumph=36, Aston Martin=110, Bentley=72}
{Silver=41, Pink=36, Blue=83, Night Blue=73, British Racing Green=40, Black=32, Dark Purple=40, Green=60, Canary Yellow=56}
--------------
Top 3 brands
Jaguar
Aston Martin
Bentley
--------------
Top 3 colors
Blue
Night Blue
Green
```

The time used to develop the complete evaluation wasn't calculated, but I spent some hours searching how to read from an xlsx in different programming languages to choose in which one should I work.