	Course	<b>Databases and Information Systems</b>			2024
	Exercise Sheet	<b>6</b>			
	Points	–			
	Release Date	<b>June 11<sup>th</sup> 2024</b>	Due Date	<b>June 26<sup>th</sup> 2024</b>	

## 6 Data Warehousing

In this exercise, you create a data warehouse to analyse sales of various items of the department store chain *Superstore*. It is also your task to implement a tool that puts the manager of the department store chain in the position to analyse the sales made in the different department stores. The sales development of products or product families per shop, region and country over different periods such as the day, month, and quarter are of special interest.

### Note

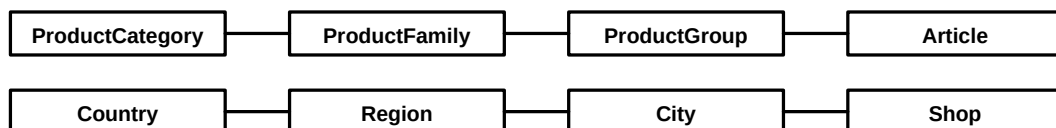
- The cross table in Exercise 6.2 can simply be printed to the console. You can choose a different table layout for your convenience as long as your table contains the required information.
- You can choose your own programming language, but we will only assist you with questions regarding java and python.
- To **avoid performance problems**, execute inserts in batches (e.g. `java.sql.PreparedStatement.addBatch()`).
- You will need the `ressources.zip` from moodle:  
<https://lernen.min.uni-hamburg.de/mod/resource/view.php?id=164340>


### 6.1 ETL Process

The first step is to implement an ETL process to populate the data warehouse prior to data analysis. The data required for analysis will be in two places:

- The data for the individual stores and the product ranges are provided in the **stores-and-products.sql**. Copy the whole code and execute it in your postgres database. This will create the tables *Country*, *Region*, *City*, *Shop*, *Article*, *ProductGroup*, *ProductFamily* and *ProductCategory*.

Unfortunately, there is no proper documentation for the tables in the corporate database. However, the following hierarchies can be reconstructed easily from the foreign key relationships:



	Course	<b>Databases and Information Systems</b>			2024
	Exercise Sheet	<b>6</b>			
	Points	–			
	Release Date	<b>June 11<sup>th</sup> 2024</b>	Due Date	<b>June 26<sup>th</sup> 2024</b>	

- The number of sold products and their turnover per department store, day and product are given in form of a CSV file. You can find the sales data of the last 5 months within the **sales.csv**. In the first line of the CSV file, there is a short description of all columns. The turnover is given in Euros. Note that, as in the real world, data may not always be as expected. So be prepared to handle corrupt data.

Implement the ETL process as an application that extracts data from both data sources and loads them into a destination schema that you also have to define the destination schema in the data warehouse is to be realised as a star schema.

During the transformation, be careful when converting data (number and date formats) and schemas (hierarchies in flat dimension tables) for the destination schema of the data warehouse. Please provide your program code with console printouts, so that your ETL process is easily understandable. Finally, populate the data warehouse with your ETL tool.


## 6.2 Data Analysis

For reasonable data analysis, the manager requires another application that enables him to navigate in the data cube. Implement an application that outputs the data as in the following cross table:

	sales	product 1	product 2	...	total
Hamburg	quarter 1, 2018	12	48	...	...
	quarter 2, 2018	31	12	...	...
	quarter 3, 2018	50	1	...	...
	quarter 4, 2018	2	0	...	...
	total	95	61	...	...
Bayern	quarter 1, 2018	11	88	...	...
	quarter 2, 2018	12	99	...	...
	quarter 3, 2018	15	75	...	...
	quarter 4, 2018	9	12	...	...
	total	47	247	...	...
...	...	...	...	...	...
total		142	335	...	...

Furthermore, your application should be able to navigate along the dimensional hierarchies (drill down, roll up). You may use extensions like `GROUPING SETS`, `ROLLUP` or `CUBE`.

**You don't have to implement a graphical user interface. To give you an idea, you could implement the following interface and the resulting console output (as shown above):**

	Course	<b>Databases and Information Systems</b> 2024		
	Exercise Sheet	<b>6</b>		
	Points	–		
	Release Date	<b>June 11<sup>th</sup> 2024</b>	Due Date	<b>June 26<sup>th</sup> 2024</b>

```

/**
 * Produces output that the manager can use.
 * The desired granularity level
 * of each dimension is given by the parameters;
 * e.g. geo = "country" is the most general
 * and geo = "shop" is the most fine-grained
 * granularity level for the geographical dimension.
 *
 * @param geo
 * admissible values: shop, city, region, country
 * @param time
 * admissible values: date, day, month, quarter, year
 * @param product
 * admissible values: article, productGroup, productFamily, productCategory
 * @throws SQLException
 */
private static void analysis(String geo, String time, String product)

```