Chapter 3: Aggregation

3.1. Get all the unique wares in the alphabetic order with the minimal and maximal prices for each.

3.2. Show top 3 wares with the most difference between minimal and maximal prices.

3.3. Show top 3 companies producing the largest number of different products.

3.4. Show the price of the most expensive ware for each category. The result should be ordered by the category.

3.5. For each recipe show the company and lists of all the products and materials. The result must contain exactly one row per recipe and sorted by company. Lists in the result must be represented as strings with values separated with comma.

3.6. Show the companies in the alphabetical order that producing larger number of different wares than consuming.

3.7. Show all the companies that produce the same ware by more than 2 different ways (recipes).

3.8. Get all the unique companies producing at least one ware from each category in the set: **Fuel**, **Food** and **Mineral**. The query should be easily modifiable to use any set of categories.

3.9. For each company get the list of all the categories of materials used and the list of categories of products. The result must contain exactly one row per company and each list must contain only the unique entries.

3.10. For each company show all the production chains (separate row per company/chain). Here the production chain is defined as the intermediate product (ware) that both product for the one recipe and material for other where both recipes are owned by the same company. Each chain must be presented in the following form (MATERIAL1,MATERIAL2,…)-[RECIPE_ID1]->(INTERMEDIATE_PRODUCT)-[RECIPE_ID2]->(PRODUCT1, PRODUCT2,…). The result must be sorted by the company.