# Exercises: JSON Processing

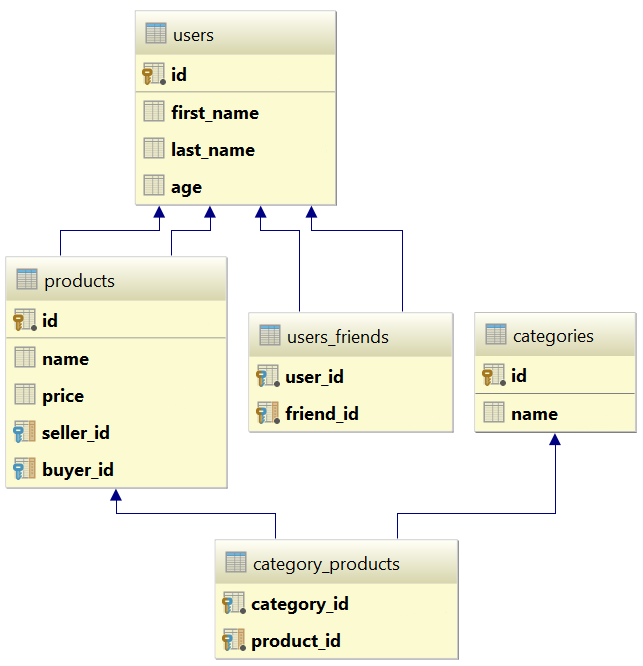
This document defines the **exercise assignments** for the ["Databases Advanced – Hibernate" course @ Software University.](https://softuni.bg/trainings/1444/databases-advanced-hibernate-october-2016)

## Products Shop

A products shop holds **users**, **products** and **categories** for the products. Users can **sell** and **buy** products.

* Users have an **id**, **first** **name** (optional) and **last** **name** (at least 3 characters) and **age** (optional).
* Products have an **id**, **name** (at least 3 characters), **price**, **buyerId** (optional) and **sellerId** as IDs of users.
* Categories have an **id** and **name** (from **3** to **15** characters)

Using Code First approach create a database following the above description.



Configure the following relations in your models:

* **Users** should have **many products sold** and **many products bought**.
* **Products** should have **many categories**
* **Categories** should have **many products**
* **Users** should have **many friends** (i.e. users)

## Seed the Database

**Import** the data from the provided files (**users.json**, **products.json**, **categories.json**).

Import the **users** first. When importing products, randomly **select the buyer** and **seller** from the existing users. Leave out some **products** that have **not been sold** (i.e. buyer is null).

Randomly **generate categories** for each product from the existing categories.

## Query and Export Data

Write the below described queries and **export** the returned data to the specified **format**.

#### Query 1 - Products In Range

Get all products in a specified **price range** (e.g. 500 to 1000) which have **no buyer**. Order them by price (from lowest to highest). Select only the **product name**, **price** and the **full name** **of the seller**. Export the result to JSON.

|  |
| --- |
| **products-in-range.json** |
| [  {  "name": "TRAMADOL HYDROCHLORIDE",  "price": 516.48,  "seller": "Christine Gomez"  },  {  "name": "Allopurinol",  "price": 518.50,  "seller": "Kathy Gilbert"  },  {  "name": "Parsley",  "price": 519.06,  "seller": "Jacqueline Perez"  },  ...  ] |

#### Query 2 - Successfully Sold Products

Get all users who have **at least 1 sold item** with a **buyer**. Order them by **last name**, then by **first name**. Select the person's **first** and **last name**. For each of the **sold products** (products with buyers), select the product's **name**, **price** and the buyer's **first** and **last name**.

|  |
| --- |
| **users-sold-products.json** |
| [  {  "firstName": "Carl",  "lastName": "Daniels",  "soldProducts": [  {  "name": "Peter Island Continous sunscreen kids",  "price": 471.30,  "buyerFirstName": "Anna",  "buyerLastName": "Parker"  },  {  "name": "Warfarin Sodium",  "price": 1379.79,  "buyerFirstName": "Brandon",  "buyerLastName": "Fuller"  }  ]  },  ...  ] |

#### Query 3 - Categories By Products Count

Get **all** **categories**. Order them by the **number of products** in that category (a product can be in many categories). For each category select its **name**, the **number of products**, the **average price of those products** and the **total revenue** (total price sum) of those products (regardless if they have a buyer or not).

|  |
| --- |
| **categories-by-products.json** |
| [  {  "category": "Sports",  "productsCount": 49,  "averagePrice": 754.327755,  "totalRevenue": 36962.06  },  {  "category": "Adult",  "productsCount": 46,  "averagePrice": 905.283478,  "totalRevenue": 41643.04  },  ...  ] |

#### Query 4 - Users and Products

Get all users who have **at least 1 sold product**. Order them by the **number of sold products** (from highest to lowest), then by **last name** (ascending). Select only their **first** and **last name**, **age** and for each product - **name** and **price**.

Export the results to **JSON**. Follow the format below to better understand how to structure your data.

|  |
| --- |
| **users-and-products.json** |
| {  "usersCount":35,  "users":  [  {  "firstName":"Carl",  "lastName":"Daniels",  "age":59,  "soldProducts":  {  "count":10,  "products":  [  {  "name":"Finasteride",  "price":1374.01  },  {  "name":"Peter Island Continous sunscreen kids",  "price":471.30  },  {  "name":"Warfarin Sodium",  "price":1379.79  },  {  "name":"Gilotrif",  "price":1454.77  },  {  "name":"Cold and Cough",  "price":218.14  },  ...  ]  }  },  {  "firstName": null,  "lastName": "Harris",  "age": 0,  "soldProducts":  {  "count":9,  "products":  [  {  "name":"Clarins Paris Skin Illusion – 114 cappuccino",  "price":811.42  },  ...  ]  }  },  ...  ]  } |