GEBZE TECHNICAL UNIVERSITY
Department of Computer Engineering
CSE 222/505 - Spring 2021

Merve GÜRLER
1901042685

1. **SYSTEM REQUIREMENTS**
   1. **Introduction**
      1. **Aim and Scope**
         1. **Aim**

            I aimed to design a complete automated system for a furniture company that has 4 branches to make easier the tasks of administrators and branch employees and, to give a better service to customers.

         2. **Scope**

            In this project, branch employees and administrator will not log in to the system with an email and password. They will only choose the branch employee option.

      2. **Goals and Criteria for Success**
         1. **Goals**
            - Easy access to product lists.
            - East access to previous orders of a customer by customer or branch employee.
            - Quick addition of branch and branch employee.
            - Quick addition of product.
            - Quick deletion of branch and branch employee.
            - Quick deletion of product.
            - Quick access to query the products in the branches.
            - Quick access to searching products in a branch.
            - Quick access to branches that has the requested products.
            - Easy access to online buying.
            - Easy access to product situations in the stock.
            - Give information to administrator quickly.
            - Easy access to addition of new product.

         2. **Criteria for Success**
            - Helping customers to buy products quickly, see products in a requested branch, see previous orders and see if a requested product is available or not in any branch.
            - Helping branch employees to check situations of products, give information to administrator, sell product, demonstrate previous order of a customer by using her/his customer number, add new order to given customers previous orders list, add product to a branch and remove product from a branch.
            - Helping administrator to add branch, remove branch, add branch employee, remove branch employee and query the products in the requested branches.

3. General Overview
This project is a complete automated system for a furniture company that has 4 branches. Administrator can add new branches.

2. System Requirements
1. Functional Requirements
Firstly, user must create a company with a company name and administrator name or continue with the default company that is created by me.

```
Company(String name, String adminName){
```

Then user can choose which user type she/he want to continue with.

For administrator;
There is no need to subscription to system for administrators.

They can add branches;
```
KWLinkedList<Branch> addBranch(String[] Branchname, int numOfNewBranches, Company com) {
```

They can remove branches;
```
KWLinkedList<Branch> removeBranch(String[] BranchName, int numOfUnnecessaryBranch, Company com) {
```

They can add branch employees;
```
KWArrayList<BranchEmployees> addBranchEmployee(String name, String surname, String email, BranchEmployees emp, Company com) {
```

They can remove branch employees;
```
KWArrayList<BranchEmployees> removeBranchEmployees(String name, String surname, String email, BranchEmployees emp, Company com) {
```

They can query the products;
```
int query(int choose, int branchNum, Company com) {
```

For Branch Employees;

They can check product situation
```
void checkProductSituation(Company com, int branchNum, int requested, int productNum) {
```

They can demonstrate the previous orders
```
void demonstratePreOrders(int id, Company com) {
```

They can give information to administrators
```
void giveInfoToAdmin(int branchNum, int productNum, Company com) {
```

They can sell products
```
String[] sellProduct(int branchNum, Company com, String requestedAmount, int customerNum, String productName, String productModel, String productColor) {
```

They can add products
```
KWLinkedList<Branch> addProduct(String productName, int branchNum, int amount, int colorAmount[], String modelName[], String colorName[], Company com) {
```

They can remove products
```
KWLinkedList<Branch> removeProduct(String productName, int branchNum, String modelName, String colorName, Company com) {
```

For customers;

They can search products, search branches that has the requested product;

```
KWArrayList<Customers> searchProduct(String productName, int isRequested, String productColor, String modelName, int returnStatement, int customerNum, Company com) {
```

They can see the product list

```
void productLists(Company com) {
```

They can shop online

```
void shopOnline(String productName, String productColor, String modelName, int customerNum, Company com) {
```

They can check previous orders

```
void checkPreviousOrders(Company com, int id) {
```

They can subscribe

```
KWArrayList<Customers> setCustomer(String name, String surname, String email, String password, int driver, int NumofCustomer, Company com) {
```

2. Non-Functional Requirements

1. Utility

Menu selection interface is written for users to use system easily.
Users are informed by error checking messages if give an invalid input.

2. Reliability

For any error system will give an error message to the user and user will have an opportunity to give a new input.

3. Performance

This project can compile with any java compiler or terminal.
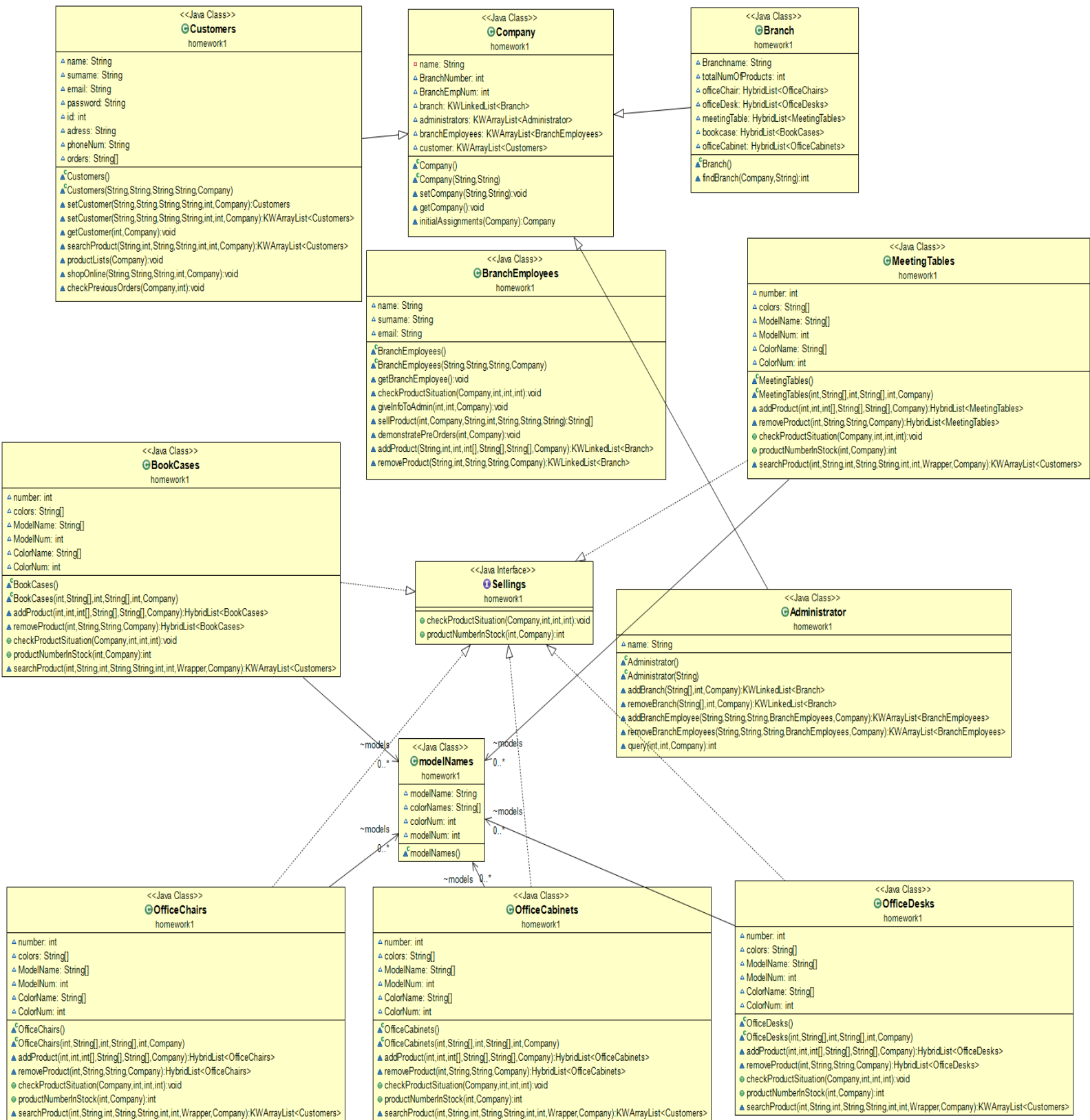
3. Supportability

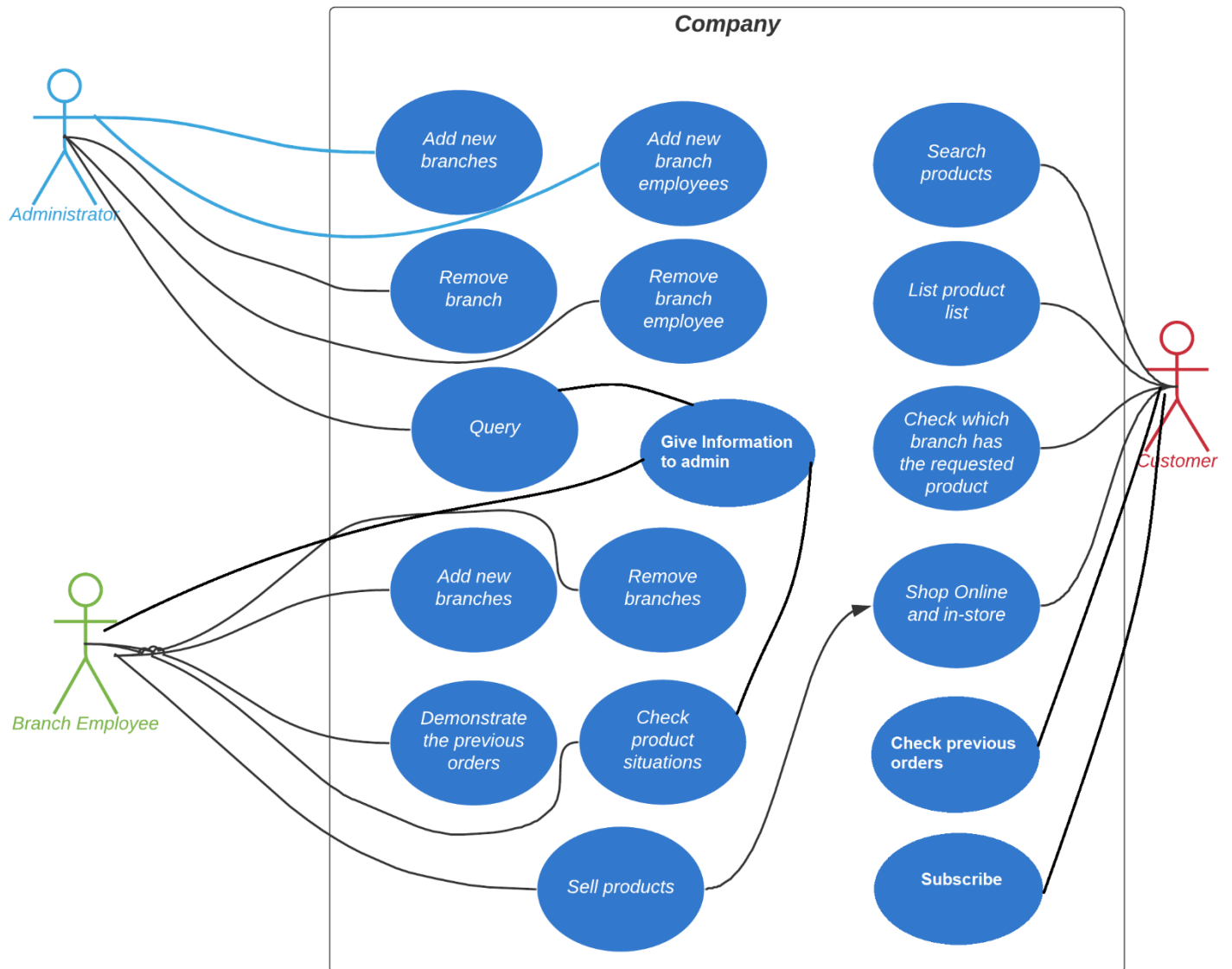System must be compiled in a java version that's level must be more than 10.

4. User Interface

Project has a user interface with an interactive menu selection mode. User will choose the best option for herself/himself.

2. Class Diagrams

<<Java Class>> Customers (homework1)

Attributes:
- name: String
- surname: String
- email: String
- password: String
- id: int
- adress: String
- phoneNum: String
- orders: String[]

Methods:
- Customers()
- Customers(String,String,String,String,Company)
- setCustomer(String,String,String,String,int,Company):Customers
- setCustomer(String,String,String,String,int,int,Company):KWArrayList<Customers>
- getCustomer(int,Company):void
- searchProduct(String,int,String,String,int,int,Company):KWArrayList<Customers>
- productLists(Company):void
- shopOnline(String,String,String,int,Company):void
- checkPreviousOrders(Company,int):void

<<Java Class>> Company (homework1)

Attributes:
- name: String
- BranchNumber: int
- BranchEmpNum: int
- branch: KWLinkedList<Branch>
- administrators: KWArrayList<Administrator>
- branchEmployees: KWArrayList<BranchEmployees>
- customer: KWArrayList<Customers>

Methods:
- Company()
- Company(String,String)
- setCompany(String,String):void
- getCompany():void
- initialAssignments(Company):Company

<<Java Class>> Branch (homework1)

Attributes:
- Branchname: String
- totalNumOfProducts: int
- officeChair: HybridList<OfficeChairs>
- officeDesk: HybridList<OfficeDesks>
- meetingTable: HybridList<MeetingTables>
- bookcase: HybridList<BookCases>
- officeCabinet: HybridList<OfficeCabinets>

Methods:
- Branch()
- findBranch(Company,String):int

<<Java Class>> BranchEmployees (homework1)

Attributes:
- name: String
- surname: String
- email: String

Methods:
- BranchEmployees()
- BranchEmployees(String,String,String,Company)
- getBranchEmployee():void
- checkProductSituation(Company,int,int,int):void
- giveInfoToAdmin(int,int,Company):void
- sellProduct(int,Company,String,int,String,String,String):String[]
- demonstratePreOrders(int,Company):void
- addProduct(String,int,int,int[],String[],String[],Company):KWLinkedList<Branch>
- removeProduct(String,int,String,String,Company):KWLinkedList<Branch>

<<Java Class>> MeetingTables (homework1)

Attributes:
- number: int
- colors: String[]
- ModelName: String[]
- ModelNum: int
- ColorName: String[]
- ColorNum: int

Methods:
- MeetingTables()
- MeetingTables(int,String[],int,String[],int,Company)
- addProduct(int,int,int[],String[],String[],Company):HybridList<MeetingTables>
- removeProduct(int,String,String,Company):HybridList<MeetingTables>
- checkProductSituation(Company,int,int,int):void
- productNumberInStock(int,Company):int
- searchProduct(int,String,int,String,String,int,int,Wrapper,Company):KWArrayList<Customers>

<<Java Class>> BookCases (homework1)

Attributes:
- number: int
- colors: String[]
- ModelName: String[]
- ModelNum: int
- ColorName: String[]
- ColorNum: int

Methods:
- BookCases()
- BookCases(int,String[],int,String[],int,Company)
- addProduct(int,int,int[],String[],String[],Company):HybridList<BookCases>
- removeProduct(int,String,String,Company):HybridList<BookCases>
- checkProductSituation(Company,int,int,int):void
- productNumberInStock(int,Company):int
- searchProduct(int,String,int,String,String,int,int,Wrapper,Company):KWArrayList<Customers>

<<Java Interface>> Sellings (homework1)

- checkProductSituation(Company,int,int,int):void
- productNumberInStock(int,Company):int

<<Java Class>> Administrator (homework1)

Attributes:
- name: String

Methods:
- Administrator()
- Administrator(String)
- addBranch(String[],int,Company):KWLinkedList<Branch>
- removeBranch(String[],int,Company):KWLinkedList<Branch>
- addBranchEmployee(String,String,String,BranchEmployees,Company):KWArrayList<BranchEmployees>
- removeBranchEmployees(String,String,String,BranchEmployees,Company):KWArrayList<BranchEmployees>
- query(int,int,Company):int

<<Java Class>> modelNames (homework1)

Attributes:
- modelName: String
- colorNames: String[]
- colorNum: int
- modelNum: int

Methods:
- modelNames()

Associations: ~models 0..*

<<Java Class>> OfficeChairs (homework1)

Attributes:
- number: int
- colors: String[]
- ModelName: String[]
- ModelNum: int
- ColorName: String[]
- ColorNum: int

Methods:
- OfficeChairs()
- OfficeChairs(int,String[],int,String[],int,Company)
- addProduct(int,int,int[],String[],String[],Company):HybridList<OfficeChairs>
- removeProduct(int,String,String,Company):HybridList<OfficeChairs>
- checkProductSituation(Company,int,int,int):void
- productNumberInStock(int,Company):int
- searchProduct(int,String,int,String,String,int,int,Wrapper,Company):KWArrayList<Customers>

<<Java Class>> OfficeCabinets (homework1)

Attributes:
- number: int
- colors: String[]
- ModelName: String[]
- ModelNum: int
- ColorName: String[]
- ColorNum: int

Methods:
- OfficeCabinets()
- OfficeCabinets(int,String[],int,String[],int,Company)
- addProduct(int,int,int[],String[],String[],Company):HybridList<OfficeCabinets>
- removeProduct(int,String,String,Company):HybridList<OfficeCabinets>
- checkProductSituation(Company,int,int,int):void
- productNumberInStock(int,Company):int
- searchProduct(int,String,int,String,String,int,int,Wrapper,Company):KWArrayList<Customers>

<<Java Class>> OfficeDesks (homework1)

Attributes:
- number: int
- colors: String[]
- ModelName: String[]
- ModelNum: int
- ColorName: String[]
- ColorNum: int

Methods:
- OfficeDesks()
- OfficeDesks(int,String[],int,String[],int,Company)
- addProduct(int,int,int[],String[],String[],Company):HybridList<OfficeDesks>
- removeProduct(int,String,String,Company):HybridList<OfficeDesks>
- checkProductSituation(Company,int,int,int):void
- productNumberInStock(int,Company):int
- searchProduct(int,String,int,String,String,int,int,Wrapper,Company):KWArrayList<Customers>

Use Case Diagrams

Company

- Add new branches
- Add new branch employees
- Search products
- Remove branch
- Remove branch employee
- List product list
- Query
- Give Information to admin
- Check which branch has the requested product
- Add new branches
- Remove branches
- Shop Online and in-store
- Demonstrate the previous orders
- Check product situations
- Check previous orders
- Sell products
- Subscribe

Administrator

Branch Employee

Customer

3.Test Cases, Running and Results

Create a company:

```
Company name: Gürler FurnitureAdministrator Name: merve
```

Add a new branch:

```
These are the initial branches

Branch Name 1: Merve
Branch Name 2: Gürler
Branch Name 3: Gebze
Branch Name 4: Teknik

Administrator adds new branch

Branch Name 1: Merve
Branch Name 2: Gürler
Branch Name 3: Gebze
Branch Name 4: Teknik
Branch Name 5: Merve Gürler Furniture
```

Remove branches:

```
These are the initial branches

Branch Name 1: Merve
Branch Name 2: Gürler
Branch Name 3: Gebze
Branch Name 4: Teknik

Administrator adds new branch

Branch Name 1: Merve
Branch Name 2: Gürler
Branch Name 3: Gebze
Branch Name 4: Teknik
Branch Name 5: Merve Gürler Furniture

Administrator removes branches from branch list

Branch Name 1: Gürler
Branch Name 2: Gebze
Branch Name 3: Teknik
```

Add branch employees:

```
These are the initial branch employees

merve gürler mgurler@gmail.com

Administrator adds branch employees

merve gürler mgurler@gmail.com
first employee first@employee.com
merve gürler mgurler@gtu.edu.tr
```

Remove branch employees:

```
These are the initial branch employees

merve gürler mgurler@gmail.com

Administrator adds branch employees

merve gürler mgurler@gmail.com
first employee first@employee.com
merve gürler mgurler@gtu.edu.tr

Administrator removes branch employees

merve gürler mgurler@gmail.com
first employee first@employee.com
```

Query products:

```
Administrator query the products in branch

There are 35 office chairs in Gürler Branch in different models and colors.
There are 20 office desks in Gürler Branch in different models and colors.
There are 40 office cabinets in Gebze Branch in different models and colors.
```

Check product situations:

```
Branch Employee checks office chairs situation in first branch

Number of office chairs in the branch: 35
Model name of the office chair in the branch: stripped
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray


Model name of the office chair in the branch: straigth
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray


Model name of the office chair in the branch: colored
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray


Model name of the office chair in the branch: etnic
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray


Model name of the office chair in the branch: antique
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray

Model name of the office chair in the branch: sport
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray


Model name of the office chair in the branch: stylish
Names of the color of office chairs in the branch: pink
Names of the color of office chairs in the branch: black
Names of the color of office chairs in the branch: purple
Names of the color of office chairs in the branch: blue
Names of the color of office chairs in the branch: gray
```

Branch employee checks product situation that is not a given product:

```
Branch Employee checks product situation that is not a given product

Invalid selection. Try again...
```

Branch Employee demonstrates the previous order of a customer:

```
Branch Employee demonstrates the previous orders of a customer

Product Name: Office Chairs
Product Model: Düz
Product Color: Mor
Branch Name: merve
Amount: 1
```

Branch Employee demonstrates the previous order of a customer that is not subscribed:

```
Branch Employee demonstrates the previous orders of a customer which is not subscribed

Customer number is not valid. Try again...
```

Branch employee add new product:

```
Branch Employee adds new product to first branch

Before the addition there are 12 bookcases in Gürler Branch in different models and colors.
After the addition there are 13 bookcases in Gürler Branch in different models and colors.

Branch Employee adds new product to third branch

Before the addition there are 12 meeting tables in Teknik Branch in different models and colors.
After the addition there are 13 meeting tables in Teknik Branch in different models and colors.
```

Branch employee removes a product:

In this part, customer removes a product with it's all colors. For example, office cabinets' one model has 4 products in 4 different colors. It deletes all 4 product.

```
Branch Employee removes one office cabinet from second branch

Before the deletion there are 40 office cabinets in Teknik Branch in different models and colors.
After the deletion there are 36 office cabinets in Teknik Branch in different models and colors.

Branch Employee removes one office desk from second branch

Before the deletion there are 20 office cabinets in Teknik Branch in different models and colors.
After the deletion there are 16 office cabinets in Teknik Branch in different models and colors.
```

Customer searches a product:

```
Customer searchs a product named Office Cabinets its model is stripped and color is pink
Product is available.

Customer searchs a product named Bookshelves(which is not a product name) its model is stripped and color is pink
Product is not available. Try again...

Customer searchs a product named Office Chairs its model is old and color is red
Product is available.
```

Customer checks which branch has the requested product:

```
Customer learns the branch of a product named Meeting Tables its model is stripped and color is pink
Product is available.
Name of the branch that product is available: Gürler

Customer learns the branch of a product named Office Tables (Which is not a product name) its model is stripped and color is pink
Product is not available. Try again...
```

Customer makes an online shopping:

```
Customer makes an online shop

Online shopping is succeeded.
```

Customer checks previous orders:

```
Customer checks previous orders

Product Name: Office Chairs
Product Model: Düz
Product Color: Mor
Branch Name: merve
Amount: 1
Product Name: OfficeChairs
Product Model: gray
Product Color: antique
Branch Name: Gürler
Amount: 1
```

Customer subscribes to system:

```
Customer subscribes to system

Name: denizSurname: deryaEmail: deryadenizPassword: dfsID: 10033550
```

Customer checks previous orders: (if the customer is not subscribed or give the wrong customer number)

```
Customer checks previous orders

There is nothing to show please shop first.
Customer number is not valid. Try again or subscribe again...
```