Gebze Technical University
Computer Engineering

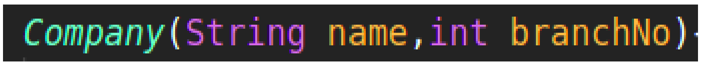
CSE 222/505 – Spring 2021

Homework 3 Report

Baran Solmaz
1801042601

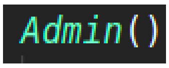
1.SYSTEM REQUIREMENTS

First of all, a **company** must be created. You have to determine company name and initial branch number.



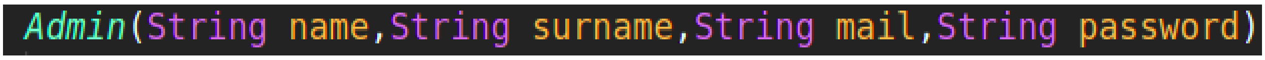
```
Company(String name,int branchNo)
```

Then, there must be an **admin**. You can create an admin by two ways. First one is simple one, just call it and it will ask for necessary information to you .



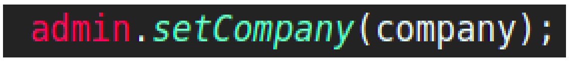
```
Admin()
```

Second one is more advanced. You have to determine admins name, surname, mail address, and password.



```
Admin(String name,String surname,String mail,String password)
```

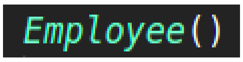
You need to give the company to admin.



```
admin.setCompany(company);
```

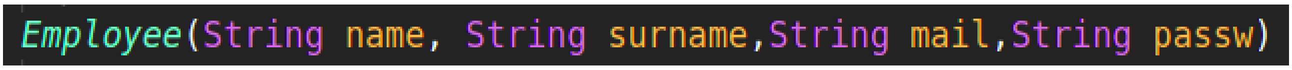
Now you need at least one employee and customer for the system to work.

To create employee:



```
Employee()
```

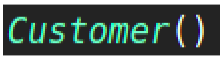
This one asks for necessary information to you .



```
Employee(String name, String surname,String mail,String passw)
```

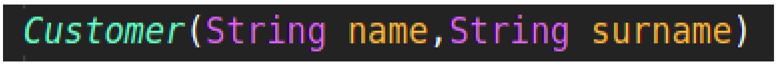
But this one take all necessary information from you.

To create customer:



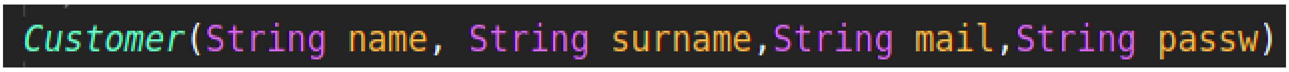
```
Customer()
```

This one asks for necessary information to you .



```
Customer(String name,String surname)
```

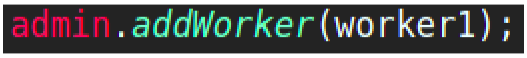
This one only needs the name and surname of employee then asks for mail and password.



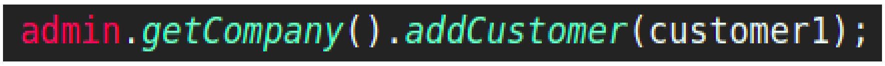
```
Customer(String name, String surname,String mail,String passw)
```

But this one take all necessary information from you.

Finally, you need to add these people to your company.



```
admin.addWorker(worker1);
```

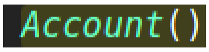


```
admin.getCompany().addCustomer(customer1);
```

Admin and Company can add both employee and customer.

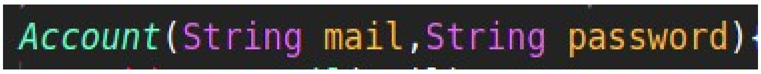
These are the basic requirements to start system.

To log in the system, you must have an account. **To create an account:**



```
Account()
```

This one asks for necessary information to you .

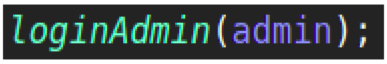


```
Account(String mail,String password)
```

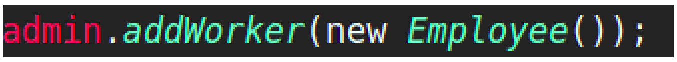
This one needs mail address and password to create an account.

But you do not need to use these account functions. When creating employee and customer, the system asks for account information automatically.

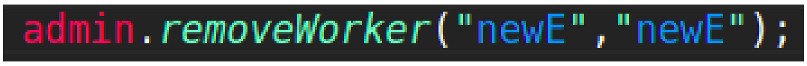
If you log into the system as admin, you can add/remove employees, add/remove branches,see requests from employees and accept those requests.



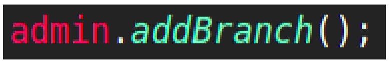
```
loginAdmin(admin);
```



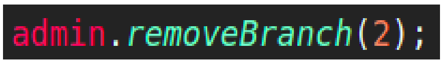
```
admin.addWorker(new Employee());
```



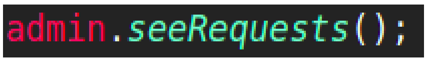
```
admin.removeWorker("newE","newE");
```



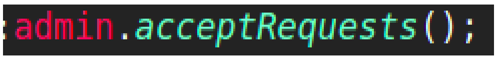
```
admin.addBranch();
```



```
admin.removeBranch(2);
```



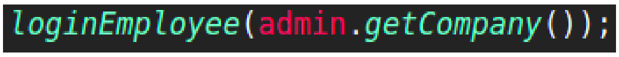
```
admin.seeRequests();
```



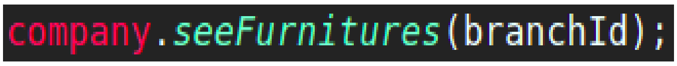
```
admin.acceptRequests();
```

All the necessary information such as employee name, id will be asked in the functions.

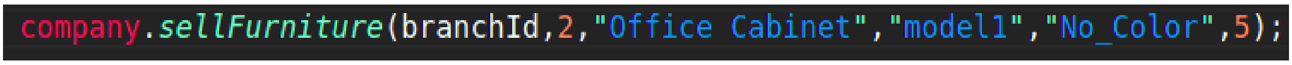
If you log into the system as an employee, you can see stocks,sell furniture to customer,request furniture from admin,get customer information like their previous orders, add new customer to system. The employee does these through the company.



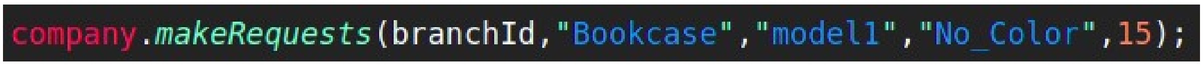
```
loginEmployee(admin.getCompany());
```



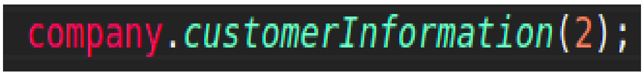
```
company.seeFurnitures(branchId);
```



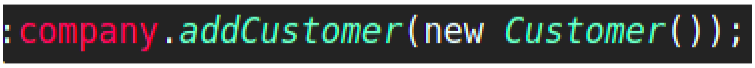
```
company.sellFurniture(branchId,2,"Office Cabinet","model1","No_Color",5);
```



```
company.makeRequests(branchId,"Bookcase","model1","No_Color",15);
```



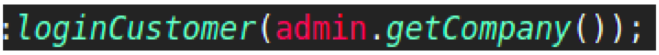
```
company.customerInformation(2);
```



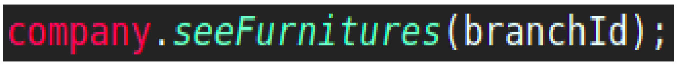
```
company.addCustomer(new Customer());
```

All the necessary information such as customer id, furniture type will be asked in the functions.

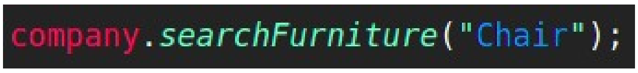
If you log into the system as customer, you can see products that in the stocks , search furniture ,order furniture and see previous orders.



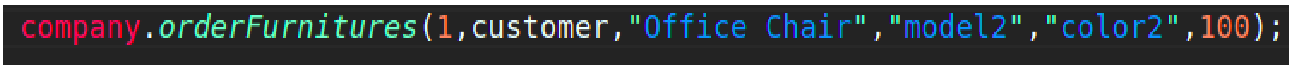
```
loginCustomer(admin.getCompany());
```



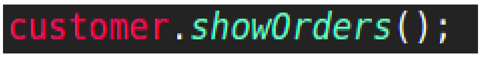
```
company.seeFurnitures(branchId);
```



```
company.searchFurniture("Chair");
```



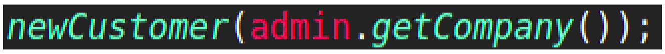
```
company.orderFurnitures(1,customer,"Office Chair","model2","color2",100);
```



```
customer.showOrders();
```

All the necessary information such as furniture type ,model color and keyword for product search will be asked in the functions.

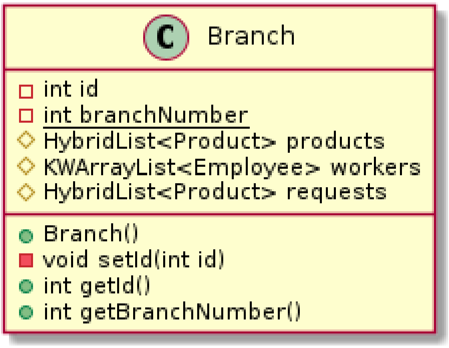
<u>If you do not have account to enter system, you can create new customer account.</u>

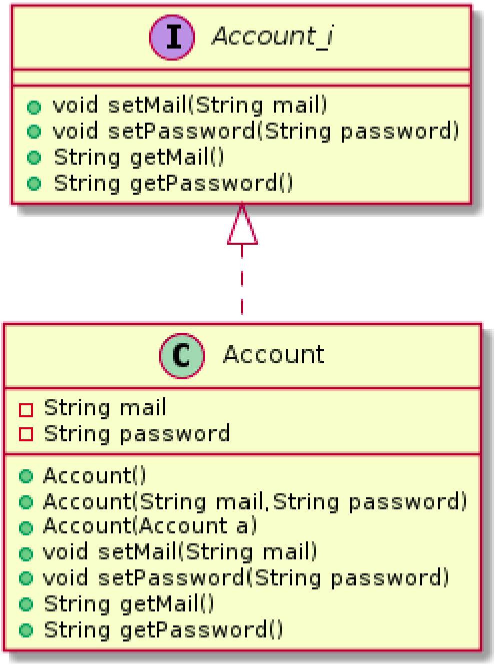


```
newCustomer(admin.getCompany());
```

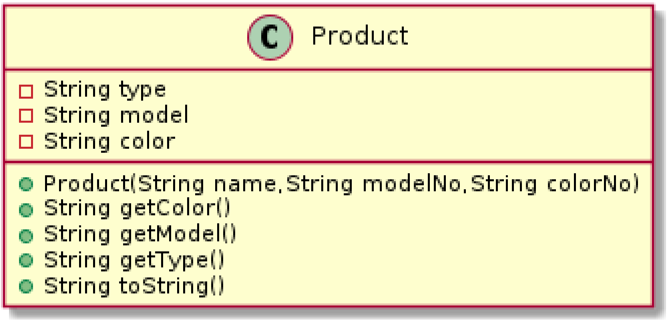
Then,you can log into the system as a customer.

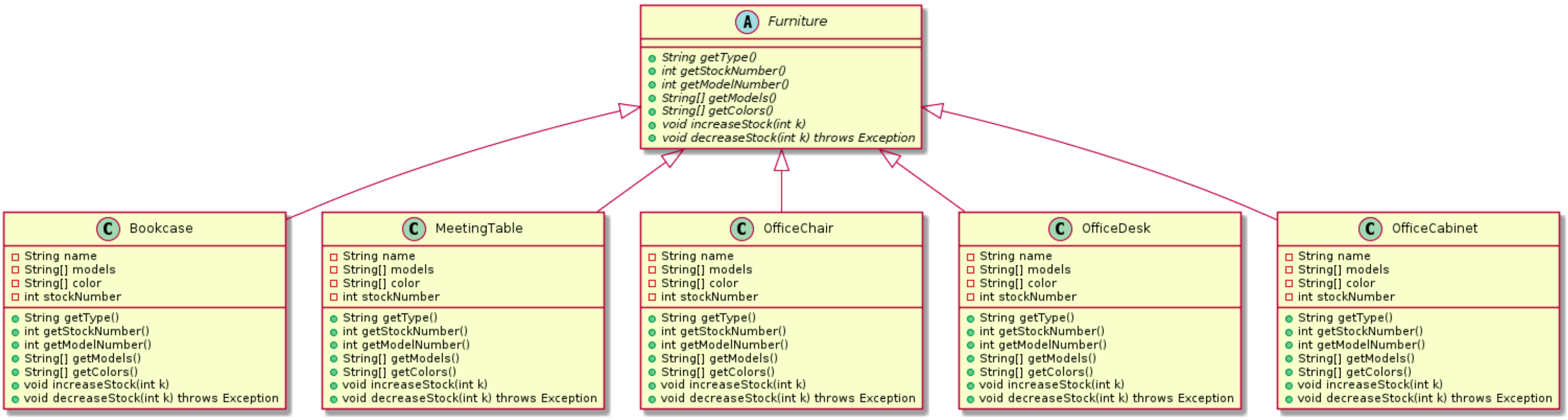
2.1 Class Diagrams

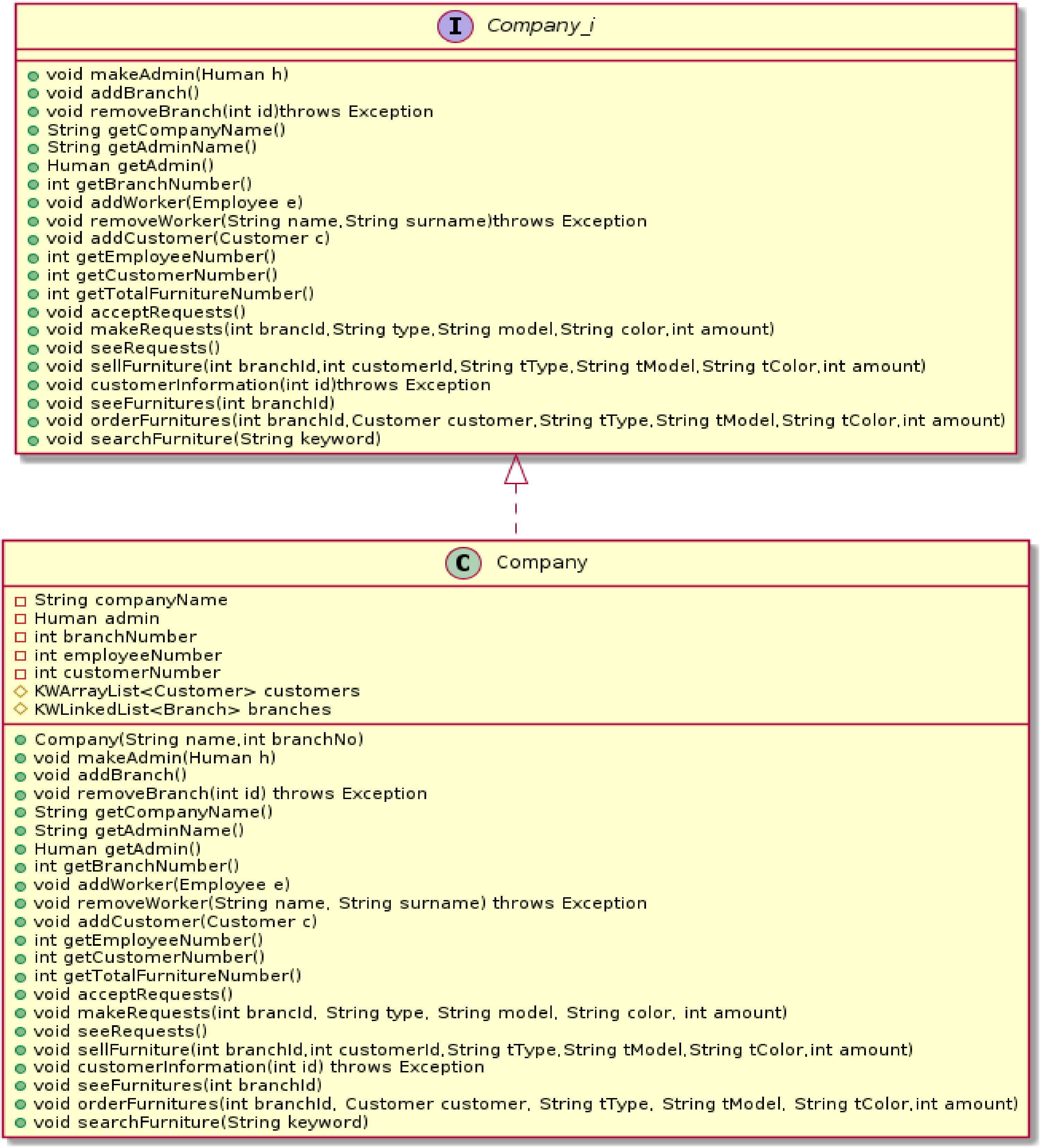


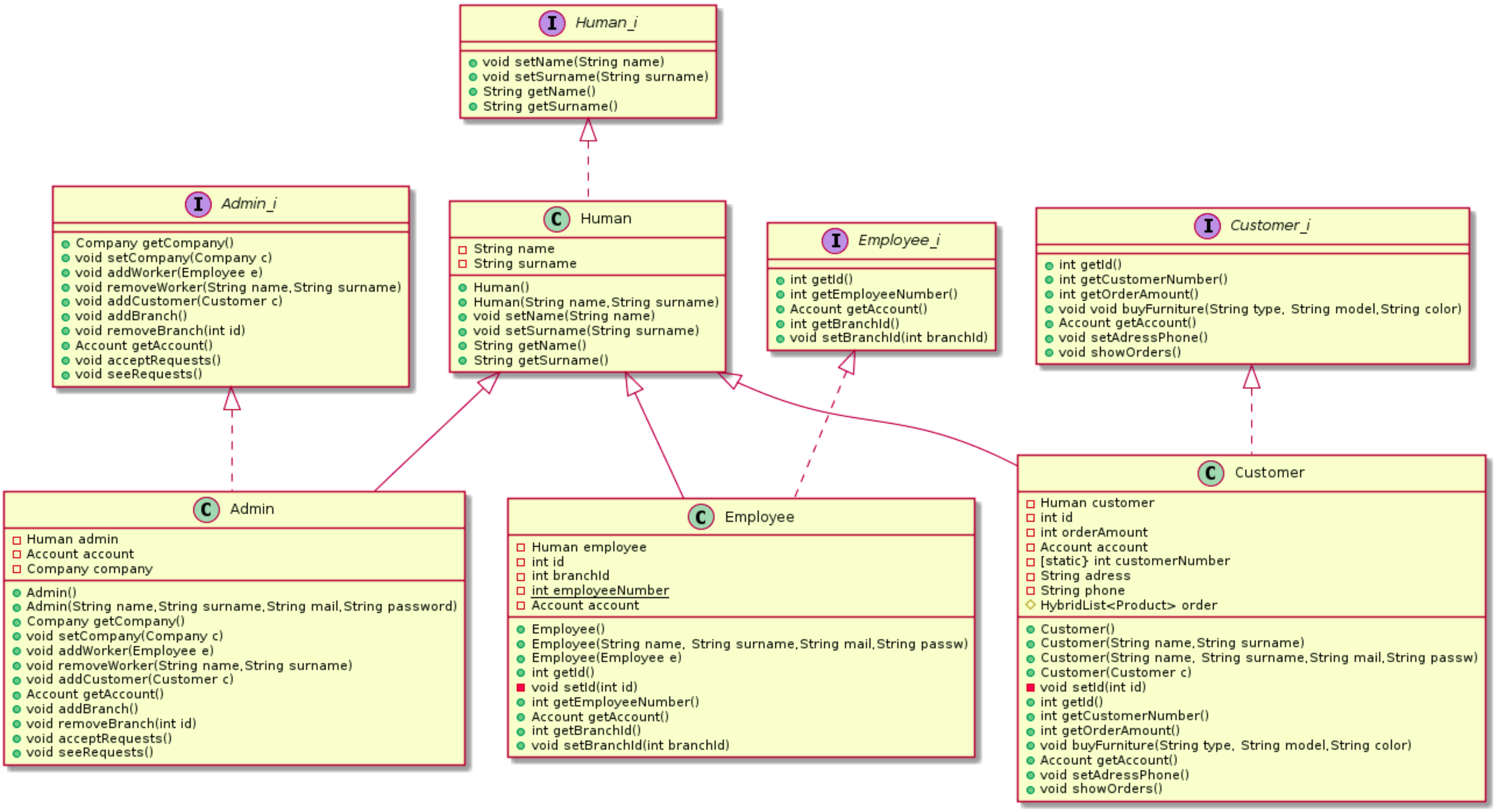


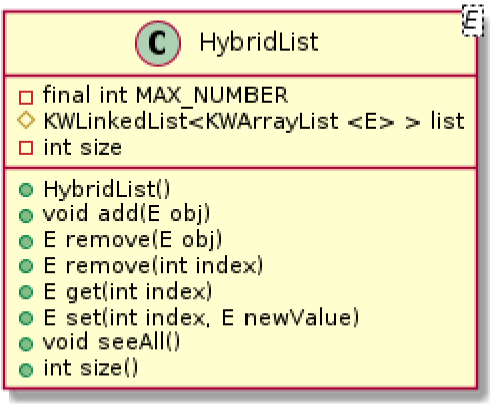
I Account_i
void setMail(String mail)
void setPassword(String password)
String getMail()
String getPassword()
C Account
String mail
String password
Account()
Account(String mail,String password)
Account(Account a)
void setMail(String mail)
void setPassword(String password)
String getMail()
String getPassword()

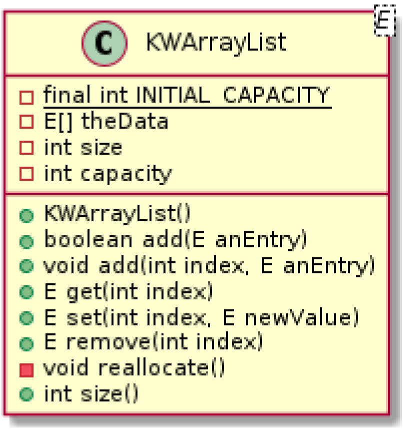




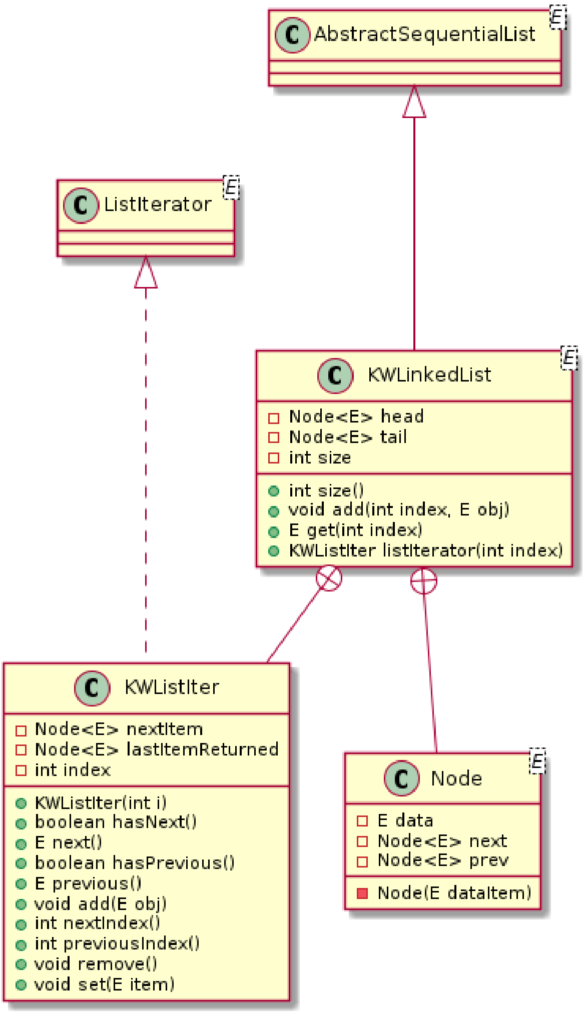

I Company_i
void makeAdmin(Human h)
void addBranch()
void removeBranch(int id)throws Exception
String getCompanyName()
String getAdminName()
Human getAdmin()
int getBranchNumber()
void addWorker(Employee e)
void removeWorker(String name,String surname)throws Exception
void addCustomer(Customer c)
int getEmployeeNumber()
int getCustomerNumber()
int getTotalFurnitureNumber()
void acceptRequests()
void makeRequests(int brancId,String type,String model,String color,int amount)
void seeRequests()
void sellFurniture(int branchId,int customerId,String tType,String tModel,String tColor,int amount)
void customerInformation(int id)throws Exception
void seeFurnitures(int branchId)
void orderFurnitures(int branchId,Customer customer,String tType,String tModel,String tColor,int amount)
void searchFurniture(String keyword)
C Company
String companyName
Human admin
int branchNumber
int employeeNumber
int customerNumber
KWArrayList<Customer> customers
KWLinkedList<Branch> branches
Company(String name,int branchNo)
void makeAdmin(Human h)
void addBranch()
void removeBranch(int id) throws Exception
String getCompanyName()
String getAdminName()
Human getAdmin()
int getBranchNumber()
void addWorker(Employee e)
void removeWorker(String name, String surname) throws Exception
void addCustomer(Customer c)
int getEmployeeNumber()
int getCustomerNumber()
int getTotalFurnitureNumber()
void acceptRequests()
void makeRequests(int brancId, String type, String model, String color, int amount)
void seeRequests()
void sellFurniture(int branchId,int customerId,String tType,String tModel,String tColor,int amount)
void customerInformation(int id) throws Exception
void seeFurnitures(int branchId)
void orderFurnitures(int branchId, Customer customer, String tType, String tModel, String tColor,int amount)
void searchFurniture(String keyword)



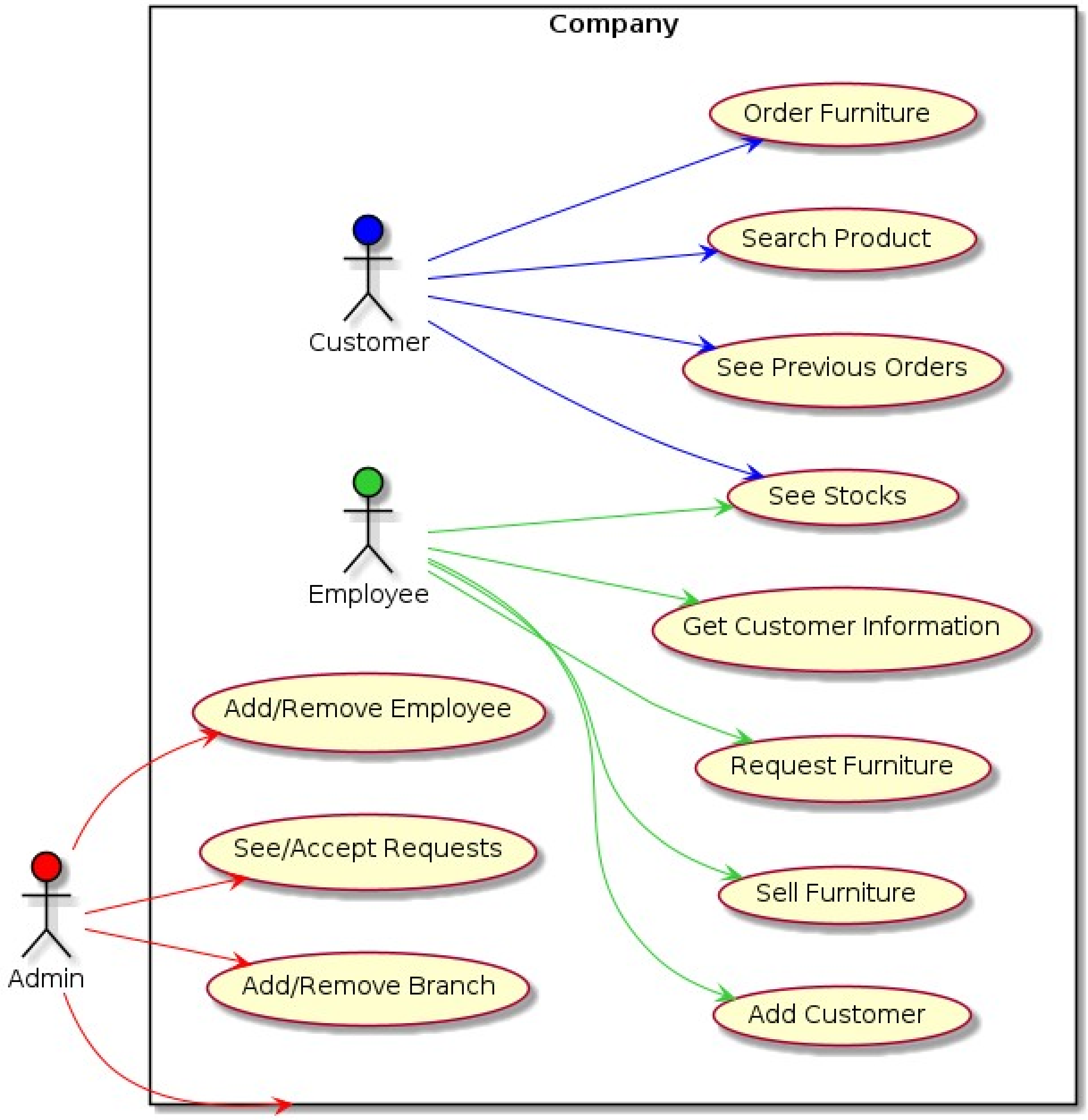




*These diagrams will be in the folder



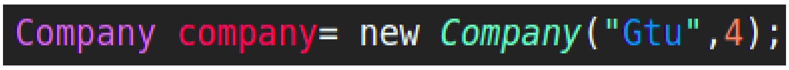
2.2 Use-Case Diagram



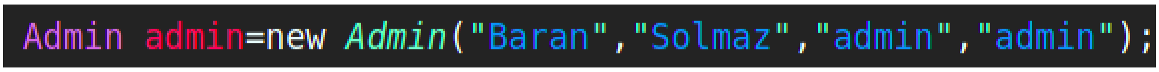
3. PROBLEM SOLUTION APPROACH

To solve problem , I used KWArrayList,KWLinkedList and HybridList Collections for system. KWArrayList for Users, KWLinkedList for Branchs and HybridList for products.

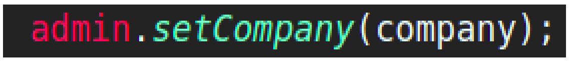
4. Test Cases – Running Commands and Results



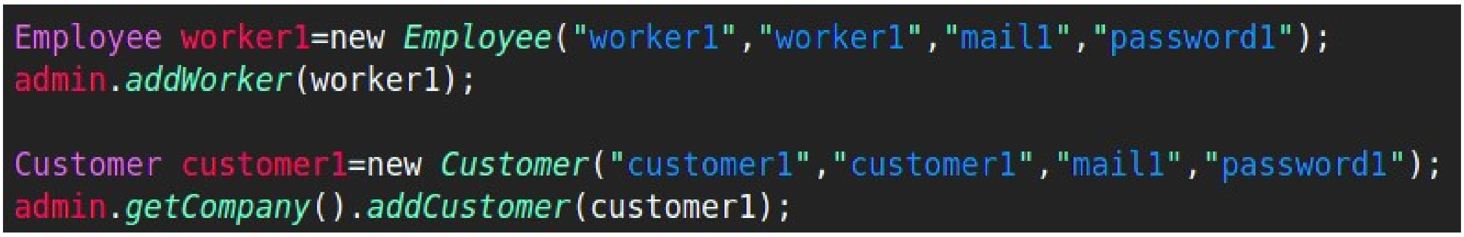
```
Company company= new Company("Gtu",4);
```



```
Admin admin=new Admin("Baran","Solmaz","admin","admin");
```

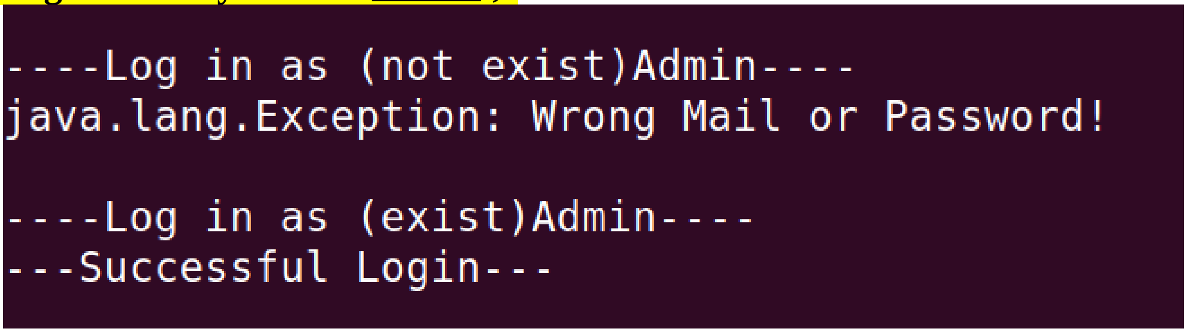


```
admin.setCompany(company);
```



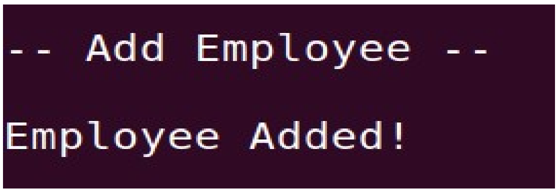
```
Employee worker1=new Employee("worker1","worker1","mail1","password1");
admin.addWorker(worker1);

Customer customer1=new Customer("customer1","customer1","mail1","password1");
admin.getCompany().addCustomer(customer1);
```

To Log into the system as Admin ;



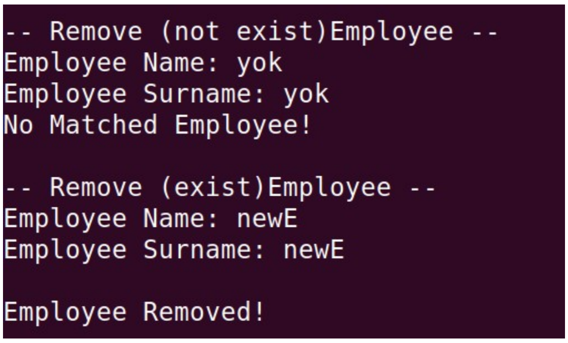
```
----Log in as (not exist)Admin----
java.lang.Exception: Wrong Mail or Password!

----Log in as (exist)Admin----
---Successful Login---
```

To add employee;



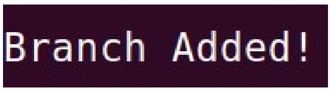
```
-- Add Employee --

Employee Added!
```

To remove employee;



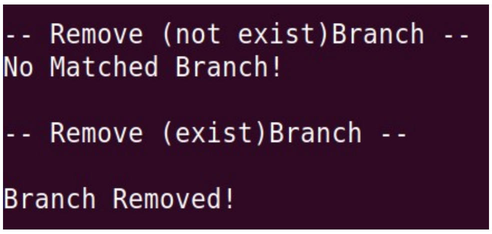
```
-- Remove (not exist)Employee --
Employee Name: yok
Employee Surname: yok
No Matched Employee!

-- Remove (exist)Employee --
Employee Name: newE
Employee Surname: newE

Employee Removed!
```

| To add branch;



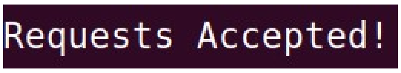
```
Branch Added!
```

|To remove branch;



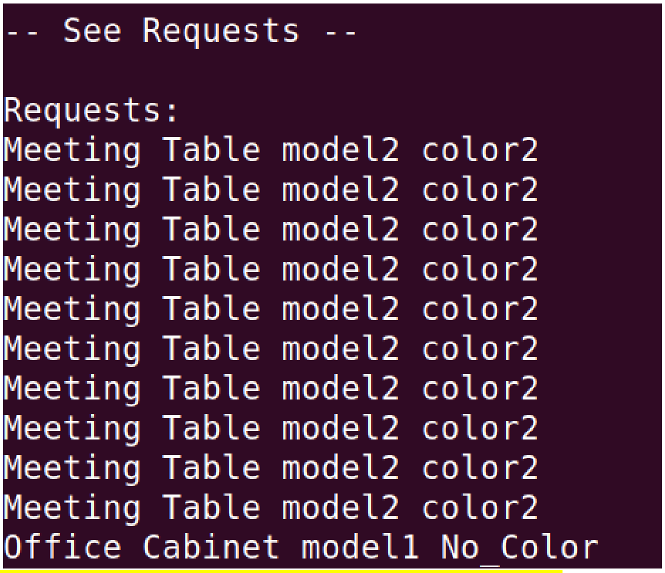
```
-- Remove (not exist)Branch --
No Matched Branch!

-- Remove (exist)Branch --

Branch Removed!
```

|To accept requests;



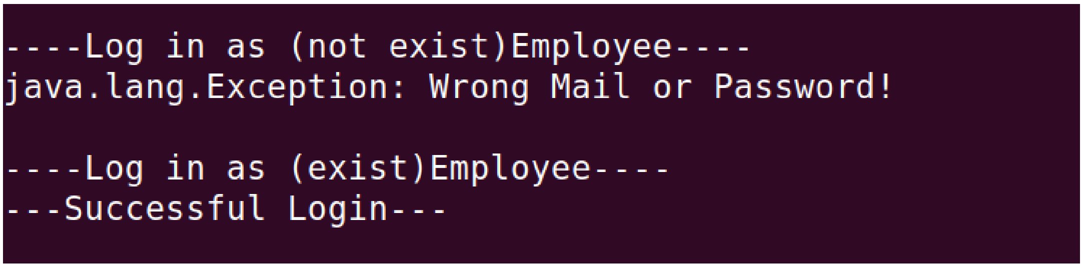
```
Requests Accepted!
```

To see requests;



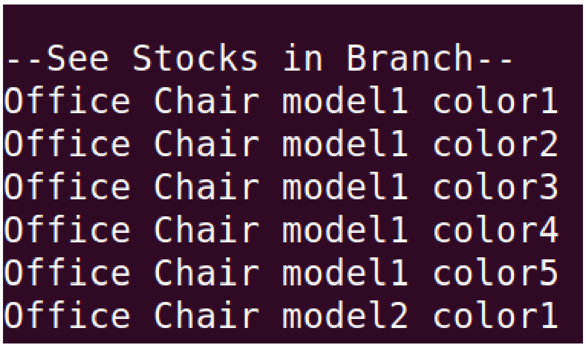
```
-- See Requests --

Requests:
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Meeting Table model2 color2
Office Cabinet model1 No_Color
```

To Log into the system as Employee ;



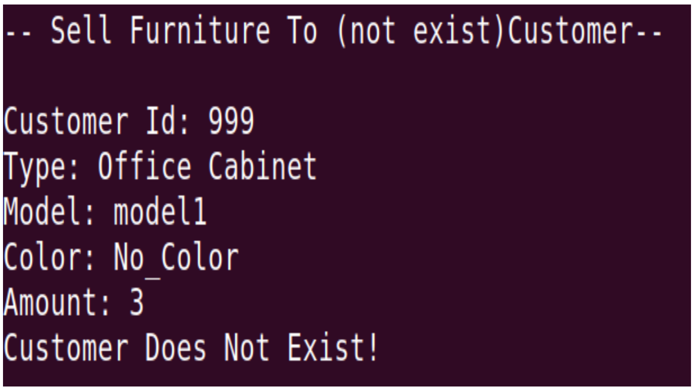
```
----Log in as (not exist)Employee----
java.lang.Exception: Wrong Mail or Password!

----Log in as (exist)Employee----
---Successful Login---
```

To see stocks;

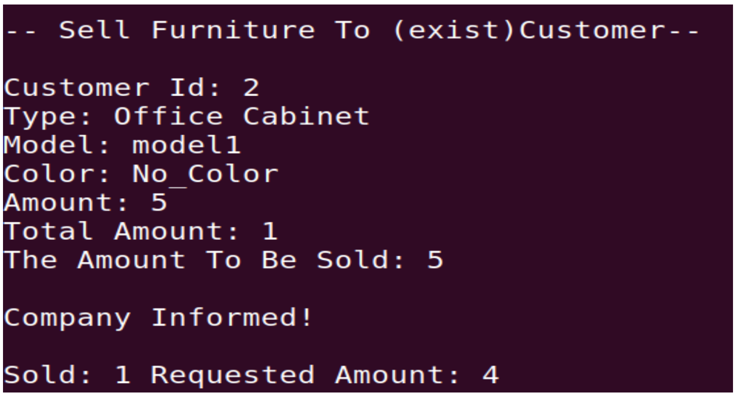


```
--See Stocks in Branch--
Office Chair model1 color1
Office Chair model1 color2
Office Chair model1 color3
Office Chair model1 color4
Office Chair model1 color5
Office Chair model2 color1
```

To sell furniture;

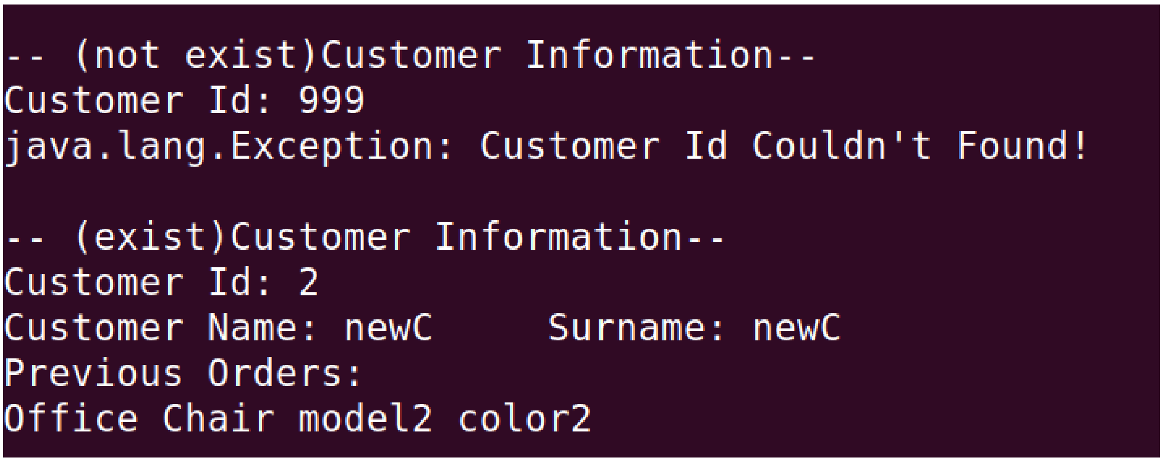


```
-- Sell Furniture To (not exist)Customer--

Customer Id: 999
Type: Office Cabinet
Model: model1
Color: No_Color
Amount: 3
Customer Does Not Exist!
```



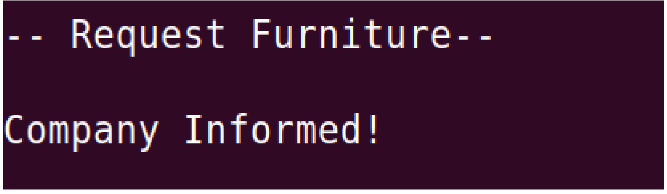
```
-- Sell Furniture To (exist)Customer--

Customer Id: 2
Type: Office Cabinet
Model: model1
Color: No_Color
Amount: 5
Total Amount: 1
The Amount To Be Sold: 5

Company Informed!

Sold: 1 Requested Amount: 4
```

To get information of customer;



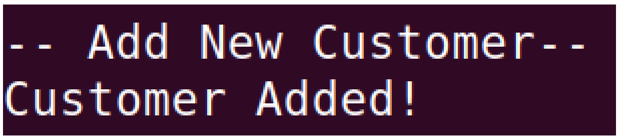
```
-- (not exist)Customer Information--
Customer Id: 999
java.lang.Exception: Customer Id Couldn't Found!

-- (exist)Customer Information--
Customer Id: 2
Customer Name: newC     Surname: newC
Previous Orders:
Office Chair model2 color2
```

To request furniture;



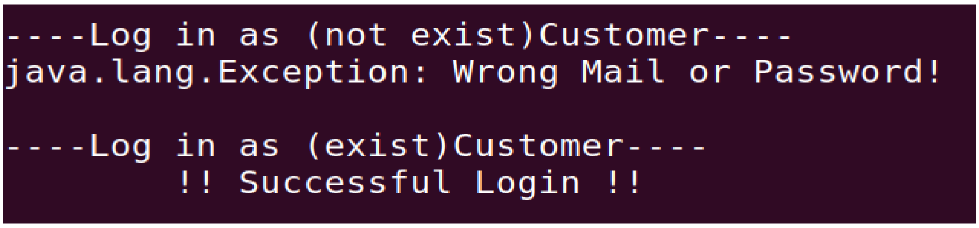
```
-- Request Furniture--

Company Informed!
```

To add new customer;



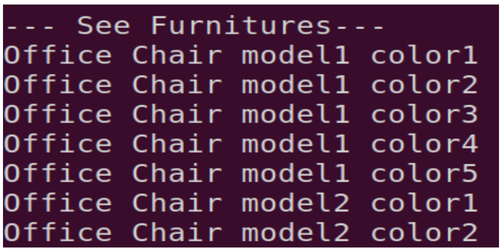
```
-- Add New Customer--
Customer Added!
```

To log into the system as Customer;



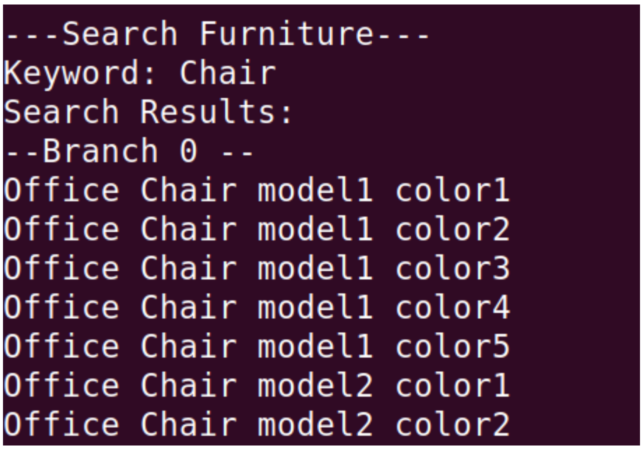
```
----Log in as (not exist)Customer----
java.lang.Exception: Wrong Mail or Password!

----Log in as (exist)Customer----
        !! Successful Login !!
```

To see stocks;*shows all furniture(different type* different model*different color)



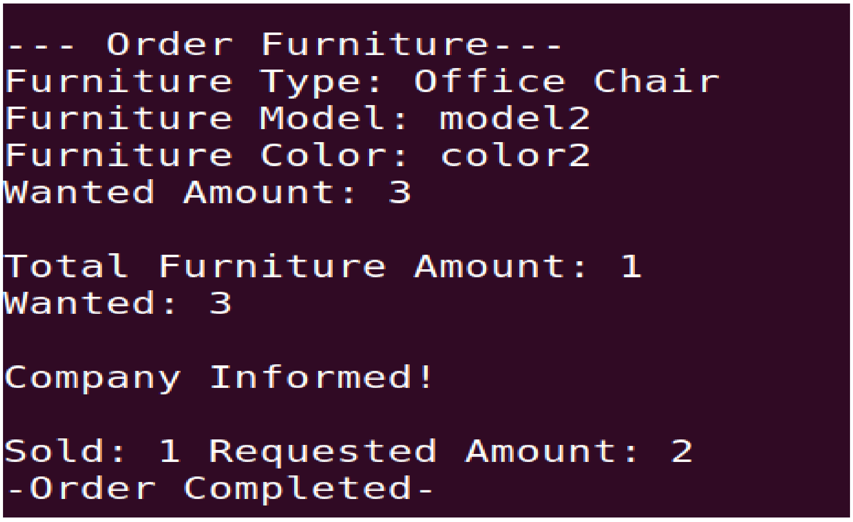
```
--- See Furnitures---
Office Chair model1 color1
Office Chair model1 color2
Office Chair model1 color3
Office Chair model1 color4
Office Chair model1 color5
Office Chair model2 color1
Office Chair model2 color2
```

To search product;*shows all furniture(different type* different model*different color)



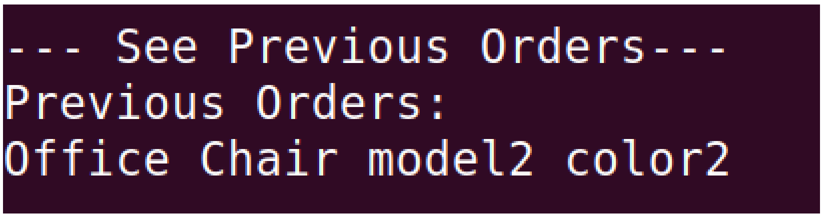
```
---Search Furniture---
Keyword: Chair
Search Results:
--Branch 0 --
Office Chair model1 color1
Office Chair model1 color2
Office Chair model1 color3
Office Chair model1 color4
Office Chair model1 color5
Office Chair model2 color1
Office Chair model2 color2
```

To order furniture;



```
--- Order Furniture---
Furniture Type: Office Chair
Furniture Model: model2
Furniture Color: color2
Wanted Amount: 3

Total Furniture Amount: 1
Wanted: 3

Company Informed!

Sold: 1 Requested Amount: 2
-Order Completed-
```

To see previous orders;



```
--- See Previous Orders---
Previous Orders:
Office Chair model2 color2
```