
# ESKİŞEHİR TECHNICAL UNIVERSITY

# ALGORITMS AND COMLEXITY

# EEM480

# HOMEWORK3

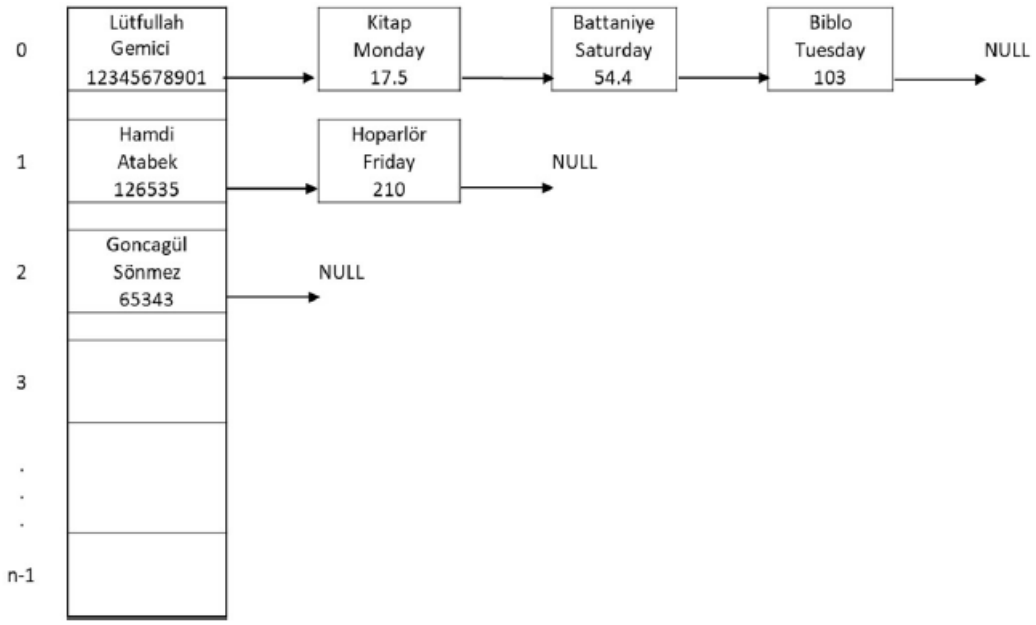

# HALİL İBRAHİM ÖZTÜRK

# 14467071268

This is a database program of our COM480 E-trade company.

The program basically keeps a customer information and his/her trade information.

Our computer department engineers require a following type of structure.




I write this Java project using Linked List and getter and setter methods.

I used Linked List on store items not on the customers.

I make  just arrays for customers.

Items for the customer will be stored as a linked list.

In this requirements the customers will be kept in Customer class as :

```
public class Customer{
private String Name;
private String Surname;
private int ID;
private Item Link;
@Override
public String toString(){
String retstr = "Name " + Name;
retstr = retstr + " Surname " + Surname;
retstr = retstr + " ID " + ID;
return retstr;
}
}
```

The items will be kept in Item class as :

```
public class Item {
String ItemName;
String Date;
float Price;
Item Link;
}
```

The Data Base Interface is given as :

```
public interface DB_Interface {
public void addCustomer(Customer newCustomer);
public void listItems(int ID);
public Customer getNewCustomer(String Name, String Surname, int ID);
public void addNewItem (Integer ID, String ItemName, String Date, float
Price);
public Float getTotalTradeofCustomer(int ID);
public Float getTotalTrade();
public void readFromFile(String path);
public Customer search_Customer(int ID);
}
```

When you are implemented the necessary methods :

addCustomer add the customer object to the customer array.

getNewCustomer gets an information of new customer from user. The information will contain Name,
Surname and ID.

addNewItem adds the new item to the linked list of corresponding array location which contains the ID. If ID is
not found, IDNotFoundException has to be thrown.

getTotalTradeofCustomer gets the ID of the user and finds the total amount of expenses of her/him and
put the result on screen.

getTotalTrade finds and shows the total amount of trades of the company.

listItems list all the items that customer with ID to the screen.

search_Customer returns the customer object of ID

readFromFile read the trade text data from file. The file must be as given below as example:

E:\Mydrive\Mydata.txt

Lutfullah Gemici 13456
13456 Biblo Tuesday 103
Hamdi Atabek 126535
13456 Battaniye Saturday 54.4
Goncagul Sonmez 65543
126535 Hoparlor Friday 210
13456Kitap Monday 17.5

As you have noticed if a line starts with an integer it belongs to an item, otherwise it belongs to a customer.
When the file given in example has been read the database given in figure 1 is created.

You are required to write the EEM480Database class which implements the DB_Interface. If the following code
is run :

```
public static void main(String[] args) {
    // TODO code application logic here
    EEM480DataBase MyDataBase = new EEM480DataBase();
    Customer DummyCustomer = new Customer();
    MyDataBase.readFromFile("e:\\MyData.txt");
    Float exps = MyDataBase.getTotalTradeofCustomer(13456);
    System.out.println(MyDataBase.search_Customer(13456) + " Total Expense : " + exps);
    System.out.println(" The Total Trade : " + MyDataBase.getTotalTrade());
    MyDataBase.listItems(13456);
    Customer newc = new Customer();
    newc = MyDataBase.getNewCustomer("Ali", "Veli", 4950);
    MyDataBase.addCustomer(newc);
    MyDataBase.addNewItem(4950, "Karburator", "Monday", 145.8);
    MyDataBase.addNewItem(4950, "Laptop" , "Tuesday", 2340);
    System.out.println(" The Total Trade : " +
            MyDataBase.getTotalTrade());
    MyDataBase.listItems(4950);
}
```

The program will produce the following output.

The output :

The content of file has been read
Name Lutfullah Surname Gemici ID 13456 Total Expense : 174.9
The Total Trade : 384.9
Lutfullah Gemici 13456 Item List :
Kitap Monday 17.5
Battaniye Saturday 54.4
Biblo Tuesday 103
The Total Trade : 2870.7
Ali Veli 4950 Item List :
Laptop Tuesday 2340
Karburator Monday 145.8

My codes on my project :

## Halil_Ibrahim_Ozturk_HW3.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package halil_ıbrahim_ozturk_hw3;

/**
 *
 * @author halilibrahim
 */
public class Halil_Ibrahim_Ozturk_HW3 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) throws Exception {
        // TODO code application logic here

        EEM480DataBase MyDataBase =new EEM480DataBase();
        Customer DummyCustomer =new Customer();
        MyDataBase.readFromFile("C:\\Mydrive\\Mydata.txt");
        Float exps = MyDataBase.getTotalTradeofCustomer(13456);
        System.out.println(MyDataBase.search_Customer(13456)+"Total Expense :"+exps);
        System.out.println(" The Total Trade : " + MyDataBase.getTotalTrade());
        MyDataBase.listItems(13456);
        Customer newc =new Customer ();
        newc=MyDataBase.getNewCustomer("Ali","Veli",4950);
```

```
        MyDataBase.addCustomer(newc);
        MyDataBase.addNewItem(4950,"Karburator","Monday",(float)145.8);
        MyDataBase.addNewItem(4950,"Laptop","Tuesday",2340);
        System.out.println("The total trade :"+ MyDataBase.getTotalTrade());
        MyDataBase.listItems(4950);
    }

}
```

## Customer.java

```
package halil_ıbrahim_ozturk_hw3;
public class Customer {
  private String Name;
  private String SurName;
  private int ID;
  private Item Link;



  public String toString() {
     String retstr = "Name " + Name;
     retstr += "Surname " + SurName;
     retstr += "ID " + ID;
     return retstr;
  }

  public String getName() {//get name (get method)
     return Name;
  }

  public void setName(String name) {//adding new customer name (set method)
     Name = name;
  }
```

```
  public String getSurName() { //getting surname (get method)
     return SurName;
  }

  public void setSurName(String surName) {//adding new customer name (set method)
     SurName = surName;
  }

  public int getID() {//(get method)
     return ID;
  }

  public void setID(int iD) {//(set method)
     ID = iD;
  }

  public Item getLink() {//(get method)
     return Link;
  }

  public void setLink(Item link) {//(set method)
     Link = link;
  }



}
```

## Item.java

```
package halil_ıbrahim_ozturk_hw3;


public class Item {
  String ItemName;
  String Date;
  float Price;
  Item Link;
public String toString() {
  return ItemName+" "+Date+" "+Price;
}
public String getItemName() {
  return ItemName;
}
public void setItemName(String itemName) {
  ItemName = itemName;
}
public String getDate() {
  return Date;
}
public void setDate(String date) {
  Date = date;
}
public float getPrice() {
  return Price;
}
public void setPrice(float price) {
  Price = price;
}
public Item getLink() {
  return Link;
```

```
}
public void setLink(Item link) {
  Link = link;
}




}
```

## DB_Interface.java

```
package halil_ıbrahim_ozturk_hw3;
public interface DB_Interface {
  public void addCustomer(Customer newCustomer);

  public void listItems(int ID);

  public Customer getNewCustomer(String Name, String SurName, int ID);

  public void addNewItem(Integer ID,String ItemName, String Date, float Price) throws Exception;

  public Float getTotalTradeofCustomer(int ID);

  public Float getTotalTrade();

  public void readFromFile(String path);

  public Customer search_Customer(int ID);
}
```

## EEM480DataBase.java

```
package halil_ıbrahim_ozturk_hw3;

import java.io.File;// Import the File class
import java.io.IOException;//general class of exceptions
import java.util.Scanner; // Import the Scanner class to read text files
import java.util.StringTokenizer;//allows an application to break a string into tokens

public class EEM480DataBase implements DB_Interface {//implement Database interface
  Customer[] customerArray = new Customer[100];//new customer object
  int size = 0;//initialize from 0

  public static void main(String[] args) {
     EEM480DataBase companyDatabase = new EEM480DataBase(); //new database object
     companyDatabase.readFromFile("E:\\Mydrive\\Mydata.txt");//read from file always
     companyDatabase.listItems(5);//print items according to ID of customer  which scan from txt file
  }

  @Override
  public void addCustomer(Customer newCustomer) {//add new customer ( new list)
     customerArray[size] = newCustomer;//new list
     size++;//new size
     //add the customer object to the customer array.
  }

  @Override
  public void listItems(int ID) { //listing items of customer
     //list all the items that customer with ID to the screen.
     for (Customer customer : customerArray) {//task condition
        if (customer != null && customer.getID() == ID) {//determine ID
```

```
            Item item = customer.getLink();
            System.out.println(customer.toString() + " Item List:");//once customer name ,surname and
ıd
            while (item != null) {
               System.out.println(item.toString());//printing items
               item = item.getLink();
            }
            break;
         }
      }

   }

   @Override
   public Customer getNewCustomer(String Name, String SurName, int ID) {
      //gets an information of new customer
      //from user. The information will contain Name,
      //Surname and ID.
      Customer cust = new Customer(); //new object
      cust.setID(ID);
      cust.setName(Name);
      cust.setSurName(SurName);
      return cust;
   }

   @Override
   public void addNewItem(Integer ID, String ItemName, String Date, float Price) throws Exception {
      //adds the new item to the linked list of
      //corresponding array location which contains the ID.
      //If ID is not found, IDNotFoundException has to be thrown.
      boolean flag = false; //create a condition for readability(to make easy algorithm)
```

```
for (Customer customer : customerArray) { //task condition
   if (customer != null && customer.getID() == ID) {//find the ID
      Item item = customer.getLink();
      Item y = item;//add new item
      if (item == null) {
         Item itm = new Item();//new item object
         //set methods to determine item informations
         itm.setItemName(ItemName);
         itm.setDate(Date);
         itm.setPrice(Price);
         itm.setLink(null);
         customer.setLink(itm);
      } else {
         while (y != null) {
            if (y.getLink() == null) {
               Item itm = new Item();
               itm.setItemName(ItemName);
               itm.setDate(Date);
               itm.setPrice(Price);
               itm.setLink(null);
               y.setLink(itm);
               break;
            } else {
               y = y.getLink();
            }
         }
         customer.setLink(item);
      }

      flag = true;
      break;
```

```
        }
    }
    if (!flag) {
        throw new Exception();
    }
}

@Override
public Float getTotalTradeofCustomer(int ID) {
    //gets the ID of the user and
    //finds the total amount of expenses of her/him and
    //put the result on screen.
    float trade = 0;
    for (Customer customer : customerArray) {
        if (customer != null && customer.getID() == ID) {
            Item item = customer.getLink();
            while (item != null) {
                trade += item.getPrice();
                item = item.getLink();
            }
            break;
        }
    }
    return trade;
}

@Override
public Float getTotalTrade() {
    //finds and shows the total amount of trades of the company
    float trade = 0;
    for (Customer customer : customerArray) {
```

```
        if (customer != null) {
            Item item = customer.getLink();
            while (item != null) {
                trade += item.getPrice();
                item = item.getLink();
            }
        }
    }
    return trade;
}

@Override
public void readFromFile(String path) {
    prepareData(path, " ");
            System.out.println("The content of file has been read");
            //read the trade text data from file.

}

public void prepareData(String filepath, String delim) {
    File file = new File(filepath);//file object

    try (Scanner sc = new Scanner(file)) {//using scanner function to scan text file
        while (sc.hasNextLine()) {

            StringTokenizer st = new StringTokenizer(sc.nextLine(), delim);
            int j = 0;
            Customer customer = null;
            Item newItem = null;
            while (st.hasMoreTokens()) {
                if (j == 0) {
```

```
    String value = st.nextToken();
    try {
        int id = Integer.parseInt(value);
        customer = search_Customer(id);//returns the customer object of ID
        Item item = customer.getLink();
        Item y = item;
        if (item == null) {
            newItem = new Item();
            newItem.setLink(null);//set methods
            customer.setLink(newItem);//set methods
        } else {
            while (y != null) {
                if (y.getLink() == null) {
                    newItem = new Item();
                    newItem.setLink(null);//set methods
                    y.setLink(newItem);//set methods
                    break;
                } else {
                    y = y.getLink();//get methods
                }
            }
            customer.setLink(item);

        }

    } catch (NumberFormatException e) {
        customer = new Customer();
        customer.setName(value);
    }

} else if (j == 1) {
```

```
            String value = st.nextToken();
            if (newItem != null) {
                newItem.setItemName(value);
            } else {
                customer.setSurName(value);
            }

        } else if (j == 2) {
            String value = st.nextToken();
            if (newItem != null) {
                newItem.setDate(value);
            } else {
                customer.setID(Integer.parseInt(value));
            }
        } else if (j == 3) {
            String value = st.nextToken();
            if (newItem != null) {
                newItem.setPrice(Float.parseFloat(value));
            }
        }
        j++;
    }
    if (newItem == null) {
        customerArray[size] = customer;
        size++;
    } else {
        for (int i = 0; i < size; i++) {
            Customer custom = customerArray[i];
            if (custom.getID() == customer.getID()) {
                customerArray[i] = customer;
                break;
```

```
                }


            }
        }
    }

    } catch (IOException e) {
        e.printStackTrace();
    } catch (Exception e) {
        e.printStackTrace();
    }

}

@Override
public Customer search_Customer(int ID) {
    //returns the customer object of ID
    for (Customer customer : customerArray) {
        if (customer != null && customer.getID() == ID)
            return customer;
    }
    return null;
}

}
```