```
package EFile;
import java.util.Scanner;
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
import java.io.FileNotFoundException;
public class EFileClient {

   public static Node<Person> loadFile(){

      LinkedList<Person> persons = new LinkedList<>();
      String fileName = "persons.txt";
      String[] data;
      String line = null;

      try{
         FileReader fileReader = new FileReader(fileName);
         BufferedReader bufferedReader = new BufferedReader(fileReader);
         while((line = bufferedReader.readLine()) != null) {
            data = line.split(",");
            if(data[0].equals("~")){
               persons.add();
               persons.tail.obj = new Person(data[1], data[2], data[3], data[4],
                     data[5], data[6], data[7], Integer.parseInt(data[8]), data[9],
                     Integer.parseInt(data[10]));
            }else if(data[0].equals("$")){
               persons.tail.obj.addFinancialAcc(data[1], data[2], Double.parseDouble(data[3]));
            }
         }
```

```
            bufferedReader.close();
        }
        catch(FileNotFoundException ex) {
            System.out.println("unable to open file: '" + fileName + "'\n");
        }
        catch(IOException ex) {
            System.out.println("error reading file: '" + fileName + "'\n");
        }
        return persons.head;
    }

    public static String financialAccount(Node<FinancialAcc> x, Person user, String a, Scanner input){

        double z = 0.0;
        String[] cmd;

        while(!a.equalsIgnoreCase("back") && !a.equalsIgnoreCase("logout")){
            System.out.print(user.getUserName() + ".financial." + x.obj.getAccountNumber() + "> ");
            a = input.nextLine();
            cmd = a.split(" ");
            if(a.equalsIgnoreCase("dis")){
                System.out.println("account Type: " + x.obj.getAccountType());
                System.out.println("balance: " + x.obj.getBalance());
                System.out.println("number of transactions: " + x.obj.getTransactions() + "\n");
            }else if(a.equalsIgnoreCase("help")){
                System.out.println("'dis' - displays account information.\n"
                        + "'help' - displays help section.\n"
                        + "'back' - go back a directory.\n"
                        + "'withd' - withdraw from account.\n");
```

```
      }else if(cmd[0].equalsIgnoreCase("withd")){
         if(cmd.length < 2){
            System.out.println("withdraw failed. try: withd [amount].\n");
         }else{
            z = x.obj.withdraw(Double.parseDouble(cmd[1]));
            if(z == -1){
               System.out.println("withdraw failed. insufficient funds.\n");
            }else{
               System.out.println("balance: " + z + "\n");
            }
         }
      }else if(cmd[0].equalsIgnoreCase("deposit")){
         if(cmd.length < 2){
            System.out.println("deposit failed. try: withd [amount].\n");
         }else{
            System.out.println("balance: " + x.obj.deposit(Double.parseDouble(cmd[1])) + "\n");
         }
      }else if (!a.equalsIgnoreCase("back") && !a.equalsIgnoreCase("logout")){
         System.out.println("uknown command.\n");
      }
   }
   if(a.equalsIgnoreCase("back")){
      a = "";
      System.out.println("");
   }
   return a;
}

public static String financial(Person user, Scanner input, String a, String[] cmd){
```

```
String x, y = "";
double z = 0.0;

while(!a.equalsIgnoreCase("back") && !a.equalsIgnoreCase("logout")){
   System.out.print(user.getUserName() + ".financial> ");
   a = input.nextLine();
   cmd = a.split(" ");
   if(a.equalsIgnoreCase("dis")){
      System.out.println(user.displayFinancialAccounts());
   }else if(a.equalsIgnoreCase("help")){
      System.out.println("'dis' - displays accounts.\n"
            + "'help' - displays help section.\n"
            + "'back' - go back a directory.\n"
            + "'new' - create new account.\n"
            + "'sacc' - select an account.\n");
   }else if(a.equalsIgnoreCase("new")){
      System.out.print("enter account number: ");
      x = input.nextLine();
      if(user.chkAccNum(x)){
         System.out.println("error, account number already exists.\n");
      }else{
         System.out.print("enter account type: ");
         y = input.nextLine();
         System.out.print("enter account balance: ");
         z = Double.parseDouble(input.nextLine());
         user.addFinancialAcc(x, y, z);
         System.out.println("");
      }
```

```
        }else if(cmd[0].equalsIgnoreCase("sacc")){
            if(cmd.length < 2){
                System.out.println("change failed. try: sacc [account number].\n");
            }else if(user.getFinancialAccount(cmd[1]) == null){
                System.out.println("account not found.\n");
            }else{
                System.out.println("");
                a = financialAccount(user.getFinancialAccount(cmd[1]), user, a, input);
            }
        }else if (!a.equalsIgnoreCase("back") && !a.equalsIgnoreCase("logout")){
            System.out.println("uknown command.\n");
        }
    }
    if(a.equalsIgnoreCase("back")){
        a = "";
        System.out.println("");
    }
    return a;
}

public static boolean account(Person user, Scanner input, String a, String[] cmd){

    String x, y = "";
    double z = 0.0;

    while(!a.equalsIgnoreCase("logout")){
        System.out.print(user.getUserName() + "> ");
        a = input.nextLine();
        cmd = a.split(" ");
```

```
if(a.equalsIgnoreCase("dis")){
   System.out.println(user.display());
}else if(a.equalsIgnoreCase("help")){
   System.out.println("'dis' - displays account information.\n"
         + "'help' - displays help section.\n"
         + "'logout' - logout of user session.\n"
         + "'fname [firstname]' - change users first name.\n"
         + "'lname [lastname]' - change users last name.\n"
         + "'setadd [address]' - change users address.\n"
         + "'setphone [phone#]' - change users phone number.\n"
         + "'setbday [birthday]' - change users birthday.\n"
         + "'setheight [height]' - change users height.\n"
         + "'setage [age]' - change users age.\n"
         + "'setweight [weight]' - change users weight.\n"
         + "'finance' - account finances.\n");
}else if(cmd[0].equalsIgnoreCase("fname")){
   if(cmd.length < 2){
      System.out.println("change failed. try: fname [first name].\n");
   }else{
     user.setFirstName(cmd[1]);
   }
}else if(cmd[0].equalsIgnoreCase("lname")){
   if(cmd.length < 2){
      System.out.println("change failed. try: lname [last name].\n");
   }else{
     user.setLastName(cmd[1]);
   }
}else if(cmd[0].equalsIgnoreCase("setadd")){
   if(cmd.length < 2){
```

```
        System.out.println("change failed. try: setadd [address].\n");
    }else{
      user.setAddress(cmd[1]);
    }
}else if(cmd[0].equalsIgnoreCase("setphone")){
    if(cmd.length < 2){
        System.out.println("change failed. try: setphone [phone number].\n");
    }else{
      user.setPhone(cmd[1]);
    }
}else if(cmd[0].equalsIgnoreCase("setbday")){
    if(cmd.length < 2){
        System.out.println("change failed. try: setbday [birthday].\n");
    }else{
      user.setBirthday(cmd[1]);
    }
}else if(cmd[0].equalsIgnoreCase("setheight")){
    if(cmd.length < 2){
        System.out.println("change failed. try: setheight [height].\n");
    }else{
      user.setHeight(cmd[1]);
    }
}else if(cmd[0].equalsIgnoreCase("setage")){
    if(cmd.length < 2){
        System.out.println("change failed. try: setage [age].\n");
    }else{
      user.setAge(Integer.parseInt(cmd[1]));
    }
}else if(cmd[0].equalsIgnoreCase("setweight")){
```

```
        if(cmd.length < 2){
            System.out.println("change failed. try: setweight [weight].\n");
        }else{
          user.setWeight(Integer.parseInt(cmd[1]));
        }
    }else if(a.equalsIgnoreCase("finance")){
        if(user.hasFinancialAcc()){
            System.out.println("");
            a = financial(user, input, a, cmd);
        }else{
            System.out.println(user.getUserName() + " does not have a financial account.");
            System.out.print("would you like to create one? (y/n): ");
            a = input.nextLine();
            if(a.equalsIgnoreCase("y")){
                System.out.print("enter account number: ");
                x = input.nextLine();
                System.out.print("enter account type: ");
                y = input.nextLine();
                System.out.print("enter account balance: ");
                z = Double.parseDouble(input.nextLine());
                user.addFinancialAcc(x, y, z);
                System.out.println("");
                a = financial(user, input, a, cmd);
            }
        }
    }else if (!a.equalsIgnoreCase("logout")){
        System.out.println("uknown command.\n");
    }
}
```

```
    System.out.println("");
    return true;
}

public static void main(String[] args){

    String a = "";
    Boolean login = false;
    Node<Person> temp = new Node<>();
    LinkedList<Person> persons = new LinkedList<>();
    int numOfAccounts = 0, count = 1;
    String[] cmd;
    Scanner input = new Scanner(System.in);
    persons.head = loadFile();

    if(persons.head != null){
        numOfAccounts = persons.head.obj.getNumOfPersons();
    }

    System.out.println("==================================");
    System.out.println("---------------EFile---------------");
    System.out.println("==================================\n");
    System.out.println("type 'help' for a usage statement.\n");

    while(!a.equals("exit")){
        System.out.print(">>> ");
        a = input.nextLine();
        cmd = a.split(" ");
        if(a.equalsIgnoreCase("help")){
```

```
    System.out.println("'dis' - displays account names.\n"
            + "'help' - displays help section.\n"
            + "'new' - creates a new account.\n"
            + "'login [uname]' - login to an account.\n"
            + "'exit' - quits program.\n");
    }else if(a.equalsIgnoreCase("dis")){
        if(numOfAccounts == 0){
            System.out.println("no accounts to display.\n");
        }else{
            temp = persons.head;
            while(temp != null) {
                System.out.println("account " + count + ": " + temp.obj.getUserName() + "\n");
                count++;
                temp = temp.next;
            }
            count = 1;
        }
    }else if(a.equalsIgnoreCase("new")){
        persons.add();
        persons.tail.obj = new Person();
        System.out.print("create username for account: ");
        a = input.nextLine();
        persons.tail.obj.setUserName(a);
        System.out.print("enter password for "
                + persons.tail.obj.getUserName() + ": ");
        a = input.nextLine();
        persons.tail.obj.setPassword(a);
        System.out.print("enter first name for "
                + persons.tail.obj.getUserName() + ": ");
```

```
a = input.nextLine();
persons.tail.obj.setFirstName(a);
System.out.print("enter last name for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
persons.tail.obj.setLastName(a);
System.out.print("enter address for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
persons.tail.obj.setAddress(a);
System.out.print("enter phone number for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
persons.tail.obj.setPhone(a);
System.out.print("enter birthday for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
persons.tail.obj.setBirthday(a);
System.out.print("enter age for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
persons.tail.obj.setAge(Integer.parseInt(a));
System.out.print("enter height for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
persons.tail.obj.setHeight(a);
System.out.print("enter weight for "
      + persons.tail.obj.getUserName() + ": ");
a = input.nextLine();
```

```
            persons.tail.obj.setWeight(Integer.parseInt(a));
            numOfAccounts++;
            System.out.println("");
        }else if(cmd[0].equalsIgnoreCase("login")){
            if(cmd.length < 2){
                System.out.println("login failed. try: login [uname].\n");
            }else{
                temp = persons.head;
                while(temp != null) {
                    if(temp.obj.getUserName().equalsIgnoreCase(cmd[1])){
                        System.out.print("enter password: ");
                        a = input.nextLine();
                        System.out.println("");
                        if(temp.obj.getPassword().equalsIgnoreCase(a)){
                            login = account(temp.obj, input, a, cmd);
                        }
                    }
                    temp = temp.next;
                }
                if(!login){
                    System.out.println("login information incorrect.\n");
                }
            }
        }else if (!a.equalsIgnoreCase("exit")){
            System.out.println("uknown command.\n");
        }
    }
  }
}
```