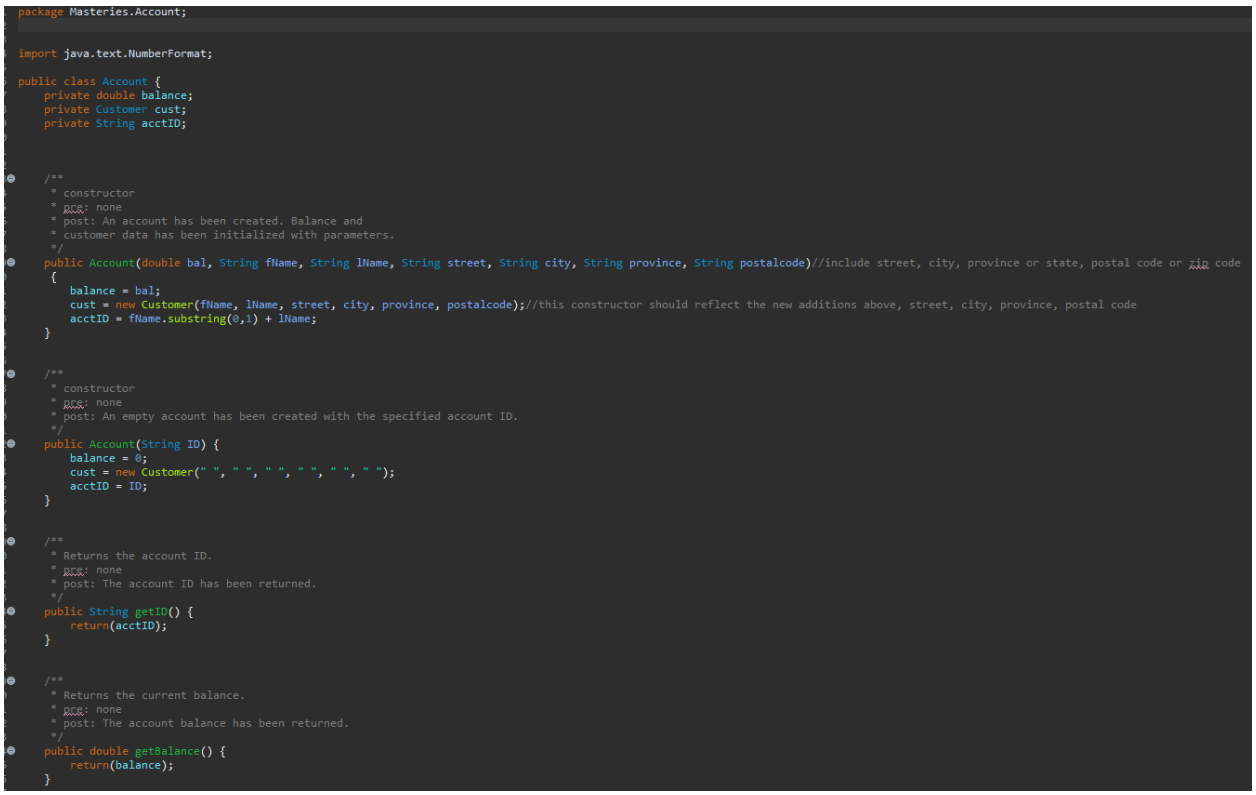
Credit Name: Chapter 8
Assignment Name: Account Mastery

Name: Grayson Ardron

Reflection log

First I imported the given code for account and Customer and filled in the needed information required as shown in the next 5 pictures.



```java
package Masteries.Account;

import java.text.NumberFormat;

public class Account {
    private double balance;
    private Customer cust;
    private String acctID;



    /**
     * constructor
     * pre: none
     * post: An account has been created. Balance and
     * customer data has been initialized with parameters.
     */
    public Account(double bal, String fName, String lName, String street, String city, String province, String postalcode)//include street, city, province or state, postal code or zip code
     {
        balance = bal;
        cust = new Customer(fName, lName, street, city, province, postalcode);//this constructor should reflect the new additions above, street, city, province, postal code
        acctID = fName.substring(0,1) + lName;
    }


    /**
     * constructor
     * pre: none
     * post: An empty account has been created with the specified account ID.
     */
    public Account(String ID) {
        balance = 0;
        cust = new Customer(" ", " ", " ", " ", " ", " ", " ");
        acctID = ID;
    }


    /**
     * Returns the account ID.
     * pre: none
     * post: The account ID has been returned.
     */
    public String getID() {
        return(acctID);
    }


    /**
     * Returns the current balance.
     * pre: none
     * post: The account balance has been returned.
     */
    public double getBalance() {
        return(balance);
    }
```

```java
    /**
     * A deposit is made to the account.
     * pre: none
     * post: The balance has been increased by the amount of the deposit.
     */
    public void deposit(double amt) {
        balance += amt;
    }


    /**
     * A withdrawal is made from the account if there is enough money.
     * pre: none
     * post: The balance has been decreased by the amount withdrawn.
     */
    public void withdrawal(double amt) {
        if (amt <= balance) {
            balance -= amt;
        } else {
            System.out.println("Not enough money in account.");
        }
    }



    //Create a changeAddress() method that calls the cust object from above in order to change
    //Street, city, province, postalCode
    public void changeAddress() {
        cust.changeStreet();
        cust.changeCity();
        cust.changeProvince();
        cust.changePostalCode();

    }



    /**
     * Returns a true when objects have matching account ids.
     * pre: none
     * post: true has been returned when the objects are equal,
     * false returned otherwise.
     */
    public boolean equals(Object acct) {
        Account testAcct = (Account)acct;
        if (acctID.equals(testAcct.acctID)) {
                return(true);
            } else {
                return(false);
            }
    }
```

```java
    /**
     * Returns a String that represents the Account object.
     * pre: none
     * post: A string representing the Account object has
     * been returned.
     */
    public String toString() {
        String accountString;
        NumberFormat money = NumberFormat.getCurrencyInstance();


        accountString = acctID + "\n";
        accountString += cust.toString();
        accountString += "Current balance is " + money.format(balance);
        return(accountString);
    }
}
```

```java
public class Customer {
    private String firstName, lastName;

    //create String variables street, city, province, postal code

    private String changeStreet, changeCity, changeProvince, changePostalCode;


    /**
     * constructor
     * pre: none
     * post: A Customer object has been created.
     * Customer data has been initialized with parameters.
     */
    public Customer(String fName, String lName, String street, String city, String province, String postalcode) //modify constructor to inclu
    {
        firstName = fName;
        lastName = lName;
        changeCity = city;
        changeStreet = street;
        changeProvince = province;
        changePostalCode = postalcode;

        //reflect the changes in the parameter
    }



    //create changeCity method that asks the user their city and records city in a variable above
    public void changeCity() {
        Scanner user = new Scanner(System.in);

        System.out.println("input city name: ");
        changeCity = user.nextLine();

    }

    //create changeStreet method that asks the user their street and records street in a variable above
    public void changeStreet() {
         Scanner user = new Scanner(System.in);

        System.out.println("input street name: ");
        changeStreet = user.nextLine();

    }
    //create changeProvince method that asks the user their province and records province in a variable above
    public void changeProvince() {
         Scanner user = new Scanner(System.in);

        System.out.println("input province name: ");
        changeProvince = user.nextLine();

    }
```

```java
//create changePostalCode method that asks the user their postal code and records postal code in a variable above
public void changePostalCode() {
     Scanner user = new Scanner(System.in);

    System.out.println("input postal code ");
    changePostalCode = user.nextLine();

}


/**
 * Returns a String that represents the Customer object.
 * pre: none
 * post: A string representing the Customer object has
 * been returned.
 */
public String toString() {
    String custString;

    //update this string so that it contains the street, city, province, and postal code
    custString = firstName + " " + lastName + " " + changeStreet + " " + changeCity + " " + changeProvince + " " + changePostalCode + "\n";
    return(custString);
}
```

Next I made business and personal accounts which embody the types of accounts given and later manipulated.

Business Account

```java
public class BusinessAcct extends Account {
    private static final double MIN_BALANCE = 500.0;
    private static final double PENALTY = 10.0;

    public BusinessAcct(double bal, String fName, String lName, String street, String city, String province, String postalcode) {
        super(bal, fName, lName, street, city, province, postalcode);
        applyPenalty();
    }

    public BusinessAcct(String ID) {
        super(ID);
        applyPenalty();
    }

    @Override
    public void withdrawal(double amt) {
        super.withdrawal(amt);
        applyPenalty();
    }

    private void applyPenalty() {
        if (getBalance() < MIN_BALANCE) {
            super.withdrawal(PENALTY);
        }
    }
public String toString() {

        return super.toString();
    }
}
```

Personal Account

```java
public class PersonalAcct extends Account {

    private static final double MIN_BALANCE = 100.0;
    private static final double PENALTY = 2.0;


    public PersonalAcct(double bal, String fName, String lName, String street, String city, String province, String postalcode) {
        super(bal, fName, lName, street, city, province, postalcode);
        applyPenalty();
    }

    public PersonalAcct(String ID) {
        super(ID);
        applyPenalty();
    }

    @Override
    public void withdrawal(double amt) {
        super.withdrawal(amt);
        applyPenalty();
    }

    private void applyPenalty() {
        if (getBalance() < MIN_BALANCE) {
            super.withdrawal(PENALTY);


        }
    }


    public String toString() {


        return super.toString();
    }
}
```