# COP2210: Project 3

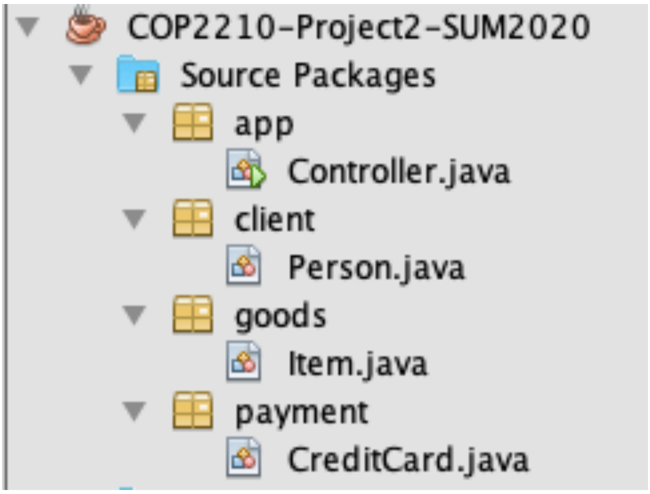
**Important:**
Do not copy someone's code. You may be randomly selected to explain your code. If you cannot explain your code, you will get a zero for the project grade. If you are stuck please ask me. I will gladly help you. Remember the goal is to learn.

**SUPER IMPORTANT !!!!**
Start looking at this code right away and work on it a little at a time. It will be a pain, but you will learn a lot from this project. Here is a road map to get this project right. Follow the order below and you will get the project done with you understanding each part of it.

**Grading**
What I am checking in your code:

- The Code is not a copy. If it is a copy you will get a grade of zero.
- Code indentation
- The Code runs correctly
- The Code output is correct and is in the correct format
- The Code adheres to the UML specifications

**Steps:**
1) Create a Netbeans Project that matches the figure below. The Controller.java has the main().
2) Create Package Structure such that it matches the structure shown below:



3) Create the Person.java, Item.java, CreditCard.java files and make sure they are in the correct package.

4) Copy and paste the Item.java code that the end of the document into your Item.java file. This code does not need to be modified.

5) You need to add the following file header on the top of the Controller.java file. Not doing this will result in a 40% reduction for your project grade.

**File Header**

```
//============================================================================
// PROGRAMMER:          Your name
// PANTHER ID:          Your panther ID
//
// CLASS:               COP2210
// SECTION:             Your class section: example U01
// SEMESTER:            The current semester: example Fall 2018
// CLASSTIME:           Your COP2210 course meeting time :example T/TH 9:00-10:15 am
//
// Project:             Put what this project is: example Lab 5 or Project 1
// DUE:
//
// CERTIFICATION:       I understand FIU's academic policies, and I certify that this work is my
//                      own and that none of it is the work of any other person.
//============================================================================
```

6) Copy and paste the Person.java code into your Person.java file. Write the code for 1 and 2 that are illustrated below. Look at the comments in the code they state what you need to do.

```
package client;

import java.util.ArrayList;
import payment.CreditCard;


public class Person {

    //----------------------------------------------------------------------------
    // Instance Variable
    //----------------------------------------------------------------------------
    // check the UML diagram to see which instance variable you need
1)  // YOUR CODE HERE

    ArrayList creditCards;


    //----------------------------------------------------------------------------
    // Constructor
    //----------------------------------------------------------------------------

    public Person(String firstName,
                  String lastName,
                  String streetAddress,
                  String suiteAddress,
                  String cityAddress,
                  String stateAddress) {

        // assign the input values to the instance variables
2) // YOUR CODE HERE

        creditCards =  new ArrayList<>();

    }//end constructor
```

7) In your project's Person.java file complete the code for the remaining setter and getters methods at location 1. Look at the UML diagram to see which methods you need to write. IMPORTANT: Keep the displayInfo() method commented out for now.

```
    //---------------------------------------------------------------------------
    // Setter and Getters
    //---------------------------------------------------------------------------
    public String getFirstName() {
        return firstName;
    }

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public void setLastName(String lastName) {
        this.lastName = lastName;
    }

    public String getStreetAddress() {
        return streetAddress;
    }

    public void setStreetAddress(String streetAddress) {
        this.streetAddress = streetAddress;
    }

    public String getSuiteAddress() {
        return suiteAddress;
    }

    public void setSuiteAddress(String suiteAddress) {
        this.suiteAddress = suiteAddress;
    }

    // write the code for the needed setter and getters
    // check the UML Diagram
1)  // YOUR CODE HERE
```

8) Copy and paste the Controller.java code into your Controller.java file. Write the code for 1 that is illustrated below.

```
package app;

//imports here
// YOUR CODE HERE

public class Controller {


    public static void main(String[] args) throws InterruptedException{


        // create a person variable named john with following info:
        // FirstName: John
        // LastName: Doe
        // Street Address: 1100 Brickell Ave
        // Suite Address: Apt 102
        // City: Miami
        // State: Florida
1)      // YOUR CODE HERE
```

9) Copy and paste the CreditCard.java code into your CreditCard.java file. Write the code for 1 that is illustrated below. Look at the comments in the code they state what you need to do.

```
package payment;

import client.Person;
import goods.Item;
import java.util.ArrayList;
import java.util.Date;


public class CreditCard {

    Person cardHolder;
    String type;

    String cardNumber;

    double creditLimit;
    double currentBalance;
    double nextPaymentAmount;

    ArrayList<Item> transactions;
    ArrayList<Date> transactionsTimeStamps;

    //--------------------------------------------------------------------------
    // Constructor
    //--------------------------------------------------------------------------

    public CreditCard(Person cardHolder, String type, double creditLimit) {


        // assign the input to the instance variables
1)       // YOUR CODE HERE

        transactions = new ArrayList<Item>();
        transactionsTimeStamps = new ArrayList<Date>();
    }
```

10) In your CreditCard.java file write the code for 1 and 2 that are illustrated below. Look at the comments in the code they state what you need to do. In 1 you are checking if you can make the charge by seeing if your item price less than the credit card's available credit remaining. **Hint you do not have an instance variable for this. Do not create an instance variable for this. If you do you will lose many points.**

```
   public void makeCharge(Item item){


1)         if( // YOUR CODE HERE){

            transactions.add(item);
            Date date = new Date();
            transactionsTimeStamps.add(date);
            currentBalance += item.getPrice();


            System.out.println("");
            System.out.println("Charging: " + item.getName());
            System.out.println("Transaction completed successfully");
            System.out.println("Please remove your " + type );
            System.out.println("");
        }else{
            System.out.println("");
            System.out.println("Charging: " + item.getName());
            System.out.println("Transaction was not successful");
            System.out.println("Credit Limit Issue");
            System.out.println("Please remove your " + type );
            System.out.println("");
        }

    }//end makeCharge


    public void transactionsReport(){

        System.out.println("");

System.out.println("============================================================================");
        System.out.println(type + " Transaction Report" );

System.out.println("============================================================================");
        System.out.printf("%-20s $%-10.2f\n", "Credit Limit:",creditLimit);

        // you need print out the Available Credit
2)        // YOUR CODE HERE


        System.out.printf("%-20s $%-10.2f\n", "Current Balance:", currentBalance);
        System.out.println("-------------------------------------------------------------------------");

        double totalCharges = 0.0;

        for(int i=0; i<transactions.size();i++){


            Item item = transactions.get(i);
            Date date = transactionsTimeStamps.get(i);
            System.out.printf("%-20s %-10s $%-10.2f %-10s\n",item.getName(),
                                                             item.getCatagory(),
                                                             item.getPrice(),
                                                             date.toString() );
            totalCharges += item.getPrice();
        }//end for


        System.out.println("-------------------------------------------------------------------------");
        System.out.printf("%-15s Total Charges:  $%-10.2f\n","", totalCharges);

    }//end transactionsReport


}
```

11)

```
      // create a Credit Card variable named masterCard with the following info:
      // cardHolder: the person that you create above
      // Type: MasterCard
      // Credit Limit: 2500.00
1)     // YOUR CODE HERE

      // create a Credit Card variable named ax with the following info:
      // cardHolder: the person that you create above
      // Type: American Express
      // Credit Limit: 5000.00
2)     // YOUR CODE HERE

      //adding the masterCard to john
3)      //john.getCreditCards().add(masterCard);

      // add the American Express to the person john
4)      // YOUR CODE HERE

      // creating and Item variable named cafeMocha with the following info:
5)      //Item cafeMocha = new Item("Food", "Cafe Mocha", 4.77);

      // creating and Item variable named gucciSlipper with the following info:
      // Cataory: Clothing
      // Name: Gucci Princetown
      // Price: 2650.00
6)     // YOUR CODE HERE

      // creating and Item variable named coke with the following info:
      // Cataory: Food
      // Name: Coke
      // Price: 1.99
7)     // YOUR CODE HERE

      // creating and Item variable named airlinesTicket with the following info:
      // Cataory: Travel
      // Name: MIA-SFO
      // Price: 823.26
 8)     // YOUR CODE HERE
```

12) In your Controller.java file write the code for 2, 3, 4, and 5. Uncomment the code for 1 and use it to figure out how to write the code for 2 and 3. Also, uncomment the code for 4 and use it to figure out the code for 5.

```
        // john is buying cafeMocha using his MasterCard credit card
   1)     //((CreditCard)john.getCreditCards().get(0)).makeCharge(cafeMocha);

        // john is buying gucciSlipper using his MasterCard credit card
        // hint this may run into credit limit issues
   2)     // YOUR CODE HERE


        // john is buying gucciSlipper using his American Express  credit card
   3)    // YOUR CODE HERE

        // john is running a transaction Report on his MasterCard
   4)      //((CreditCard)john.getCreditCards().get(0)).transactionsReport();

        // john is running a transaction Report on his American Express
   5)     // YOUR CODE HERE
```

13) In your Controller.java file write the code for 1,2, and 3. Read the comments to get hints how to write the code. Also look at the code execution output to figure out the code.

```
// buying 7 cafeMocha using different credit cards
        for(int i=1; i<=7; i++){

            // DO NOT MESS WITH THE SLEEP CODE
            // sleep for a random time upto 1 seconds
            Thread.sleep(generator.nextInt(1001));

            // buy cafeMocha on MasterCard if the cafeMocha is a multiple  3
            // else buy it on the AX card
            // example
            // i = 1 -> Buy it on the AX
            // i = 2 -> Buy it on the AX
            // i = 3 -> Buy it on the MasterCard
            // i = 4 -> Buy it on the AX
            // ...
            // hint use if else statement
  1)          // YOUR CODE HERE
        }//end for

        // buying 5 airlinesTicket using different credit cards
        for(int i=1; i<=5; i++){

            // DO NOT MESS WITH THE SLEEP CODE
            // sleep for a random time upto 1 seconds
            Thread.sleep(generator.nextInt(1001));

            // buy airlinesTicket on MasterCard if i is even
            // else buy it on the AX card
            // hint use if else statement
 2)           // YOUR CODE HERE

        }//end for

        // buying 10 cokes using different credit cards
        for(int i=1; i<=10; i++){

            // sleep for a random time upto 1 seconds
            Thread.sleep(generator.nextInt(1001));

            // this is use to randomly select a credit card
            int randomSelectCard = generator.nextInt(2);

            // if randomSelectCard is 0 use the MasterCard
            // if this is the case print out "randomSelectCard: MasterCard"
            // if randomSelectCard is 1 use the American Express
            // if this is the case print out "randomSelectCard: MasterCard"
            // hint use if else statement
 3)           // YOUR CODE HERE


        }//end for
```

13) In your Controller.java file write the code for 1 and 2. Read the comments to get hints how to write the code. Check your code execution output. If something does not look correct at your Credit Card reporting code.

```
      // john is running a transaction Report on his masterCard
 1)     // YOUR CODE HERE


        // john is running a transaction Report on his American Express
 2)       // YOUR CODE HERE
```

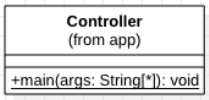
13) In your Controller.java file write the code for 1. Also you need to uncomment the displayInfo method in the Person class for it to output anything on the console when running the code.

```
        // john is running displayInfo method
    1)// YOUR CODE HERE
```

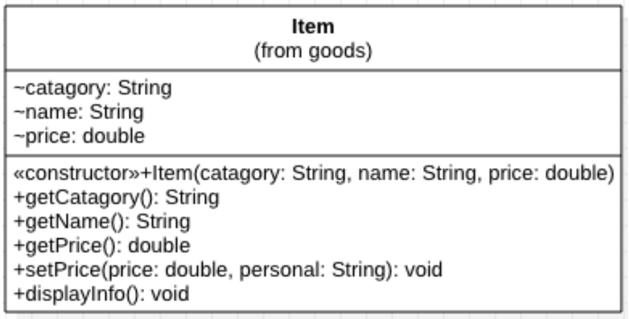
Code for the Person Class……

```
    //---------------------------------------------------------------------------
    // Utility Methods
    //---------------------------------------------------------------------------
/*
    public void displayInfo(){
        System.out.println("");

System.out.println("===========================================================================");
        System.out.println("Display Information" );

System.out.println("===========================================================================");
        System.out.printf("%-20s %s %s \n","Name:", firstName, lastName);
        System.out.printf("%-20s %-20s \n", "Address", streetAddress);
        // write the code needed for the output to match the project doc output
        // hint look at the code above to have the correct format
        // YOUR CODE HERE

        System.out.println("------------------");
        System.out.println("Credit Card Info");
        System.out.println("------------------");

         for(int i=0; i<creditCards.size();i++){

            String cardName = ((CreditCard)creditCards.get(i)).getType();

            // get the creditCards credit limit
            // hint look at the code for cardName above
            double creditLimit = // YOUR CODE HERE

            // get the creditCards current balance
            // hint look at the code for cardName above
            double currentBalance = // YOUR CODE HERE

            System.out.printf("%-20s %-20s \n", "CreditCard:", cardName);
            System.out.printf("%-20s %-20.2f \n", "Credit Limit:", creditLimit);
            System.out.printf("%-20s %-20.2f \n", "Current Balance:", currentBalance);
            System.out.println("");
        }

    }

*/
```

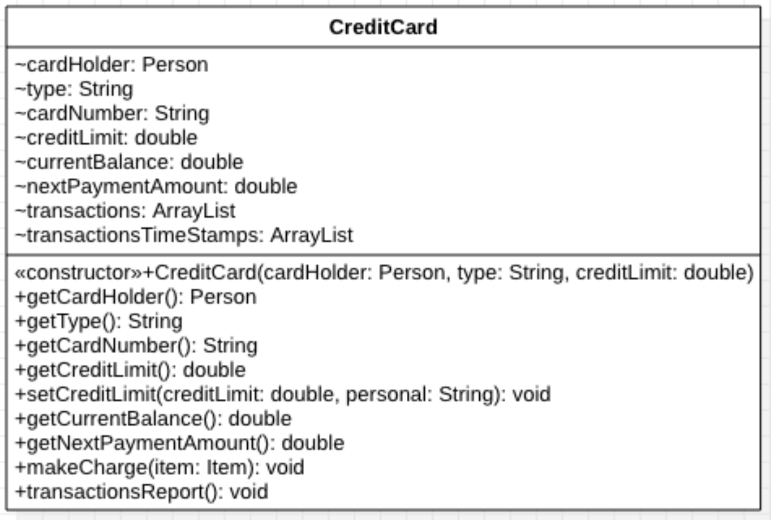
## UML Diagrams



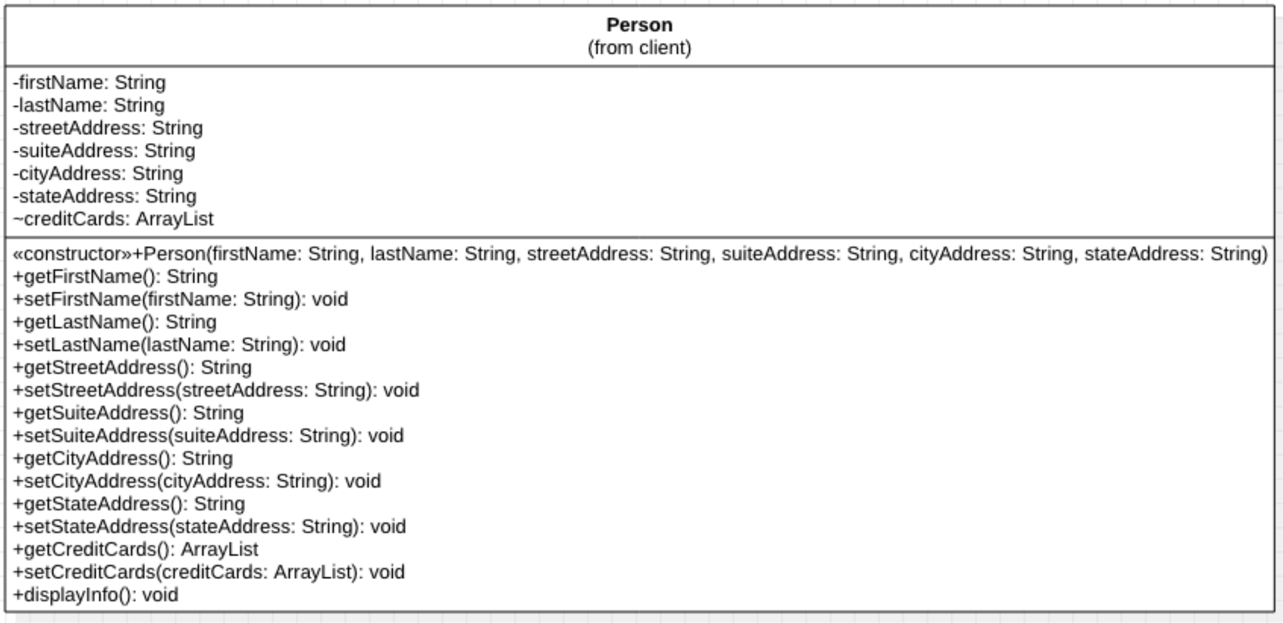
```
Controller
(from app)

+main(args: String[*]): void
```



```
Item
(from goods)

~catagory: String
~name: String
~price: double

«constructor»+Item(catagory: String, name: String, price: double)
+getCatagory(): String
+getName(): String
+getPrice(): double
+setPrice(price: double, personal: String): void
+displayInfo(): void
```



```
CreditCard

~cardHolder: Person
~type: String
~cardNumber: String
~creditLimit: double
~currentBalance: double
~nextPaymentAmount: double
~transactions: ArrayList
~transactionsTimeStamps: ArrayList

«constructor»+CreditCard(cardHolder: Person, type: String, creditLimit: double)
+getCardHolder(): Person
+getType(): String
+getCardNumber(): String
+getCreditLimit(): double
+setCreditLimit(creditLimit: double, personal: String): void
+getCurrentBalance(): double
+getNextPaymentAmount(): double
+makeCharge(item: Item): void
+transactionsReport(): void
```



```
Person
(from client)

-firstName: String
-lastName: String
-streetAddress: String
-suiteAddress: String
-cityAddress: String
-stateAddress: String
~creditCards: ArrayList

«constructor»+Person(firstName: String, lastName: String, streetAddress: String, suiteAddress: String, cityAddress: String, stateAddress: String)
+getFirstName(): String
+setFirstName(firstName: String): void
+getLastName(): String
+setLastName(lastName: String): void
+getStreetAddress(): String
+setStreetAddress(streetAddress: String): void
+getSuiteAddress(): String
+setSuiteAddress(suiteAddress: String): void
+getCityAddress(): String
+setCityAddress(cityAddress: String): void
+getStateAddress(): String
+setStateAddress(stateAddress: String): void
+getCreditCards(): ArrayList
+setCreditCards(creditCards: ArrayList): void
+displayInfo(): void
```

## Controller.java code

```
package app;

import java.util.Random;
//imports here
// YOUR CODE HERE

public class Controller {


    public static void main(String[] args) throws InterruptedException{


        // create a person variable named john with following info:
        // FirstName: John
        // LastName: Doe
        // Street Address: 1100 Brickell Ave
        // Suite Address: Apt 102
        // City: Miami
        // State: Florida
        // YOUR CODE HERE


        // create a Credit Card variable named masterCard with the following info:
        // cardHolder: the person that you create above
        // Type: MasterCard
        // Credit Limit: 2500.00
        // YOUR CODE HERE

        // create a Credit Card variable named ax with the following info:
        // cardHolder: the person that you create above
        // Type: American Express
        // Credit Limit: 5000.00
        // YOUR CODE HERE

        //adding the masterCard to john
        //john.getCreditCards().add(masterCard);

        // add the American Express to the person john
        // YOUR CODE HERE

        // creating and Item variable named cafeMocha with the following info:
        //Item cafeMocha = new Item("Food", "Cafe Mocha", 4.77);

        // creating and Item variable named gucciSlipper with the following info:
        // Cataory: Clothing
        // Name: Gucci Princetown
        // Price: 2650.00
        // YOUR CODE HERE

        // creating and Item variable named coke with the following info:
        // Cataory: Food
        // Name: Coke
        // Price: 1.99
        // YOUR CODE HERE

        // creating and Item variable named airlinesTicket with the following info:
        // Cataory: Travel
        // Name: MIA-SFO
        // Price: 823.26
        // YOUR CODE HERE


        // john is buying cafeMocha using his MasterCard credit card
        //((CreditCard)john.getCreditCards().get(0)).makeCharge(cafeMocha);

        // john is buying gucciSlipper using his MasterCard credit card
        // hint this may run into credit limit issues
        // YOUR CODE HERE


        // john is buying gucciSlipper using his American Express  credit card
        // YOUR CODE HERE
```

## Controller.java code continue…

```
        // john is running a transaction Report on his MasterCard
       //((CreditCard)john.getCreditCards().get(0)).transactionsReport();

        // john is running a transaction Report on his American Express
        // YOUR CODE HERE

        Random generator = new Random();

        // buying 7 cafeMocha using different credit cards
        for(int i=1; i<=7; i++){

            // DO NOT MESS WITH THE SLEEP CODE
            // sleep for a random time upto 1 seconds
            Thread.sleep(generator.nextInt(1001));

            // buy cafeMocha on MasterCard if the cafeMocha is a multiple  3
            // else buy it on the AX card
            // example
            // i = 1 -> Buy it on the AX
            // i = 2 -> Buy it on the AX
            // i = 3 -> Buy it on the MasterCard
            // i = 4 -> Buy it on the AX
            // ...
            // hint use if else statement
            // YOUR CODE HERE

        }//end for

        // buying 5 airlinesTicket using different credit cards
        for(int i=1; i<=5; i++){

            // DO NOT MESS WITH THE SLEEP CODE
            // sleep for a random time upto 1 seconds
            Thread.sleep(generator.nextInt(1001));

            // buy airlinesTicket on MasterCard if i is even
            // else buy it on the AX card
            // hint use if else statement
            // YOUR CODE HERE

        }//end for

        // buying 10 cokes using different credit cards
        for(int i=1; i<=10; i++){

            // sleep for a random time upto 1 seconds
            Thread.sleep(generator.nextInt(1001));

            // this is use to randomly select a credit card
            int randomSelectCard = generator.nextInt(2);

            // if randomSelectCard is 0 use the MasterCard
            // if this is the case print out "randomSelectCard: MasterCard"
            // if randomSelectCard is 1 use the American Express
            // if this is the case print out "randomSelectCard: MasterCard"
            // hint use if else statement
            // YOUR CODE HERE
        }//end for

        // john is running a transaction Report on his masterCard
        // YOUR CODE HERE


        // john is running a transaction Report on his American Express
        // YOUR CODE HERE


        // john is running displayInfo method
        // YOUR CODE HERE

    }//end main

}//end class
```

## CreditCard.java code

```
package payment;

import client.Person;
import goods.Item;
import java.util.ArrayList;
import java.util.Date;


public class CreditCard {

    Person cardHolder;
    String type;

    String cardNumber;

    double creditLimit;
    double currentBalance;
    double nextPaymentAmount;

    ArrayList<Item> transactions;
    ArrayList<Date> transactionsTimeStamps;

    //---------------------------------------------------------------------------
    // Constructor
    //---------------------------------------------------------------------------

    public CreditCard(Person cardHolder, String type, double creditLimit) {


        // assign the input to the instance variables
        // YOUR CODE HERE

        transactions = new ArrayList<Item>();
        transactionsTimeStamps = new ArrayList<Date>();
    }

    //---------------------------------------------------------------------------
    // Setters and Getters
    //---------------------------------------------------------------------------

    public Person getCardHolder() {
        return cardHolder;
    }



    public String getType() {
        return type;
    }


    public String getCardNumber() {
        return cardNumber;
    }


    public double getCreditLimit() {
        return creditLimit;
    }

    public void setCreditLimit(double creditLimit, String personal) {
        this.creditLimit = creditLimit;
    }

    public double getCurrentBalance() {
        return currentBalance;
    }

    public double getNextPaymentAmount() {
        return nextPaymentAmount;
    }


    //---------------------------------------------------------------------------
    // Utility Methods
    //---------------------------------------------------------------------------

    public void makeCharge(Item item){
```

## CreditCard.java code continue…

```
        if( // YOUR CODE HERE){

            transactions.add(item);
            Date date = new Date();
            transactionsTimeStamps.add(date);
            currentBalance += item.getPrice();


            System.out.println("");
            System.out.println("Charging: " + item.getName());
            System.out.println("Transaction completed successfully");
            System.out.println("Please remove your " + type );
            System.out.println("");
        }else{
            System.out.println("");
            System.out.println("Charging: " + item.getName());
            System.out.println("Transaction was not successful");
            System.out.println("Credit Limit Issue");
            System.out.println("Please remove your " + type );
            System.out.println("");
        }

    }//end makeCharge


    public void transactionsReport(){

        System.out.println("");

System.out.println("============================================================================");
        System.out.println(type + " Transaction Report" );

System.out.println("============================================================================");
        System.out.printf("%-20s $%-10.2f\n", "Credit Limit:",creditLimit);

        // you need ound the Avaiable Credit
        // YOUR CODE HERE


        System.out.printf("%-20s $%-10.2f\n", "Current Balance:", currentBalance);
        System.out.println("------------------------------------------------------------------------");

        double totalCharges = 0.0;

        for(int i=0; i<transactions.size();i++){


            Item item = transactions.get(i);
            Date date = transactionsTimeStamps.get(i);
            System.out.printf("%-20s %-10s $%-10.2f %-10s\n",item.getName(),
                                                             item.getCatagory(),
                                                             item.getPrice(),
                                                             date.toString() );
            totalCharges += item.getPrice();
        }//end for


        System.out.println("------------------------------------------------------------------------");
        System.out.printf("%-15s Total Charges:  $%-10.2f\n","", totalCharges);

    }//end transactionsReport


}
```

# Person.java code

```
package client;

import java.util.ArrayList;
import payment.CreditCard;


public class Person {

    //---------------------------------------------------------------------------
    // Instance Variable
    //---------------------------------------------------------------------------
    // check the UML diagram to see which instance variable you need
    // YOUR CODE HERE

    ArrayList creditCards;


    //---------------------------------------------------------------------------
    // Constructor
    //---------------------------------------------------------------------------

    public Person(String firstName,
                  String lastName,
                  String streetAddress,
                  String suiteAddress,
                  String cityAddress,
                  String stateAddress) {

        // assign the input values to the instance variables
        // YOUR CODE HERE

        creditCards =  new ArrayList<>();

    }//end constructor


    //---------------------------------------------------------------------------
    // Setter and Getters
    //---------------------------------------------------------------------------
    public String getFirstName() {
        return firstName;
    }

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    public String getLastName() {
        return lastName;
    }

    public void setLastName(String lastName) {
        this.lastName = lastName;
    }

    public String getStreetAddress() {
        return streetAddress;
    }

    public void setStreetAddress(String streetAddress) {
        this.streetAddress = streetAddress;
    }

    public String getSuiteAddress() {
        return suiteAddress;
    }

    public void setSuiteAddress(String suiteAddress) {
        this.suiteAddress = suiteAddress;
    }
```

## Person.java code continue…

```
    // write the code for the needed setter and getters
    // check the UML Diagram
    // YOUR CODE HERE

    //--------------------------------------------------------------------------
    // Utility Methods
    //--------------------------------------------------------------------------
/*
    public void displayInfo(){
        System.out.println("");

System.out.println("==========================================================================="
);
        System.out.println("Display Information" );

System.out.println("==========================================================================="
);
        System.out.printf("%-20s %s %s \n","Name:", firstName, lastName);
        System.out.printf("%-20s %-20s \n", "Address", streetAddress);
        // write the code needed for the output to match the project doc output
        // hint look at the code above to have the correct format
        // YOUR CODE HERE

        System.out.println("------------------");
        System.out.println("Credit Card Info");
        System.out.println("------------------");

         for(int i=0; i<creditCards.size();i++){

            String cardName = ((CreditCard)creditCards.get(i)).getType();

            // get the creditCards credit limit
            // hint look at the code for cardName above
            double creditLimit = // YOUR CODE HERE

            // get the creditCards current balance
            // hint look at the code for cardName above
            double currentBalance = // YOUR CODE HERE

            System.out.printf("%-20s %-20s \n", "CreditCard:", cardName);
            System.out.printf("%-20s %-20.2f \n", "Credit Limit:", creditLimit);
            System.out.printf("%-20s %-20.2f \n", "Current Balance:", currentBalance);
            System.out.println("");
        }

    }

*/

    //---------------------------------------------------



}//end class
```

## Item.java code

```java
package goods;


public class Item {

    String catagory;
    String name;
    double price;

    public Item(String catagory, String name, double price) {
        this.catagory = catagory;
        this.name = name;
        this.price = price;
    }

    public String getCatagory() {
        return catagory;
    }

    public String getName() {
        return name;
    }

    public double getPrice() {
        return price;
    }

    public void setPrice(double price, String personal) {
        this.price = price;
    }

    public void displayInfo(){

    }

}
```

## Code Execution Output

**IMPORTANT: Your code output will be different because of the random selection when buying the 10 cokes. Hence the transaction reports and balances will be different at each run.**

```
Charging: Cafe Mocha
Transaction completed successfully
Please remove your MasterCard


Charging: Gucci Princetown
Transaction was not successful
Credit Limit Issue
Please remove your MasterCard


Charging: Gucci Princetown
Transaction completed successfully
Please remove your American Express


============================================================================
MasterCard Transaction Report
============================================================================
Credit Limit:        $2500.00
Avaiable Credit:     $2495.23
Current Balance:     $4.77
----------------------------------------------------------------------------
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:10 EDT 2020
----------------------------------------------------------------------------
                Total Charges:  $4.77

============================================================================
American Express Transaction Report
============================================================================
Credit Limit:        $5000.00
Avaiable Credit:     $2350.00
Current Balance:     $2650.00
----------------------------------------------------------------------------
Gucci Princetown     Clothing   $2650.00    Thu Mar 26 14:11:10 EDT 2020
----------------------------------------------------------------------------
                Total Charges:  $2650.00

Charging: Cafe Mocha
Transaction completed successfully
Please remove your American Express


Charging: Cafe Mocha
Transaction completed successfully
Please remove your American Express


Charging: Cafe Mocha
Transaction completed successfully
Please remove your MasterCard


Charging: Cafe Mocha
Transaction completed successfully
Please remove your American Express


Charging: Cafe Mocha
Transaction completed successfully
Please remove your American Express


Charging: Cafe Mocha
Transaction completed successfully
Please remove your MasterCard


Charging: Cafe Mocha
Transaction completed successfully
Please remove your American Express
```

```
Charging: MIA-SFO
Transaction completed successfully
Please remove your American Express


Charging: MIA-SFO
Transaction completed successfully
Please remove your MasterCard


Charging: MIA-SFO
Transaction completed successfully
Please remove your American Express


Charging: MIA-SFO
Transaction completed successfully
Please remove your MasterCard


Charging: MIA-SFO
Transaction was not successful
Credit Limit Issue
Please remove your American Express

randomSelectCard: MasterCard

Charging: Coke
Transaction completed successfully
Please remove your MasterCard

randomSelectCard: American Express

Charging: Coke
Transaction completed successfully
Please remove your American Express

randomSelectCard: American Express

Charging: Coke
Transaction completed successfully
Please remove your American Express

randomSelectCard: MasterCard

Charging: Coke
Transaction completed successfully
Please remove your MasterCard

randomSelectCard: American Express

Charging: Coke
Transaction completed successfully
Please remove your American Express

randomSelectCard: American Express

Charging: Coke
Transaction completed successfully
Please remove your American Express

randomSelectCard: American Express

Charging: Coke
Transaction completed successfully
Please remove your American Express

randomSelectCard: MasterCard

Charging: Coke
Transaction completed successfully
Please remove your MasterCard

randomSelectCard: MasterCard
```

```
Charging: Coke
Transaction completed successfully
Please remove your MasterCard

randomSelectCard: MasterCard

Charging: Coke
Transaction completed successfully
Please remove your MasterCard


============================================================================
MasterCard Transaction Report
============================================================================
Credit Limit:        $2500.00
Avaiable Credit:     $829.22
Current Balance:     $1670.78
----------------------------------------------------------------------------
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:10 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:12 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:12 EDT 2020
MIA-SFO              Travel     $823.26     Thu Mar 26 14:11:14 EDT 2020
MIA-SFO              Travel     $823.26     Thu Mar 26 14:11:15 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:17 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:20 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:21 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:22 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:22 EDT 2020
----------------------------------------------------------------------------
                Total Charges:  $1670.78

============================================================================
American Express Transaction Report
============================================================================
Credit Limit:        $5000.00
Avaiable Credit:     $669.68
Current Balance:     $4330.32
----------------------------------------------------------------------------
Gucci Princetown     Clothing   $2650.00    Thu Mar 26 14:11:10 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:10 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:11 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:12 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:12 EDT 2020
Cafe Mocha           Food       $4.77       Thu Mar 26 14:11:13 EDT 2020
MIA-SFO              Travel     $823.26     Thu Mar 26 14:11:13 EDT 2020
MIA-SFO              Travel     $823.26     Thu Mar 26 14:11:15 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:18 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:19 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:20 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:20 EDT 2020
Coke                 Food       $1.99       Thu Mar 26 14:11:21 EDT 2020
----------------------------------------------------------------------------
                Total Charges:  $4330.32

============================================================================
Display Information
============================================================================
Name:                John Doe
Address              1100 Brickell Ave
Suite:               Apt 102
City:                Miami
State:               Florida
------------------
Credit Card Info
------------------
CreditCard:          MasterCard
Credit Limit:        2500.00
Current Balance:     1670.78

CreditCard:          American Express
Credit Limit:        5000.00
Current Balance:     4330.32
```