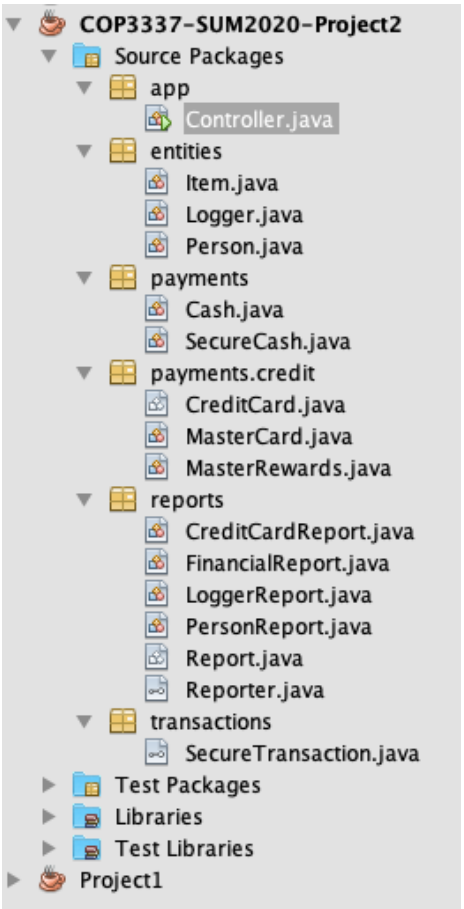
COP3337
Summer 2020- Project 2

Read through the document. Study the package structure and UML diagrams carefully. All the steps you need to do to complete the project are listed at the end of this document. Please download the Project2 zipped file which can be found on the canvas assignment page. This zipped file contains the shell of the project.

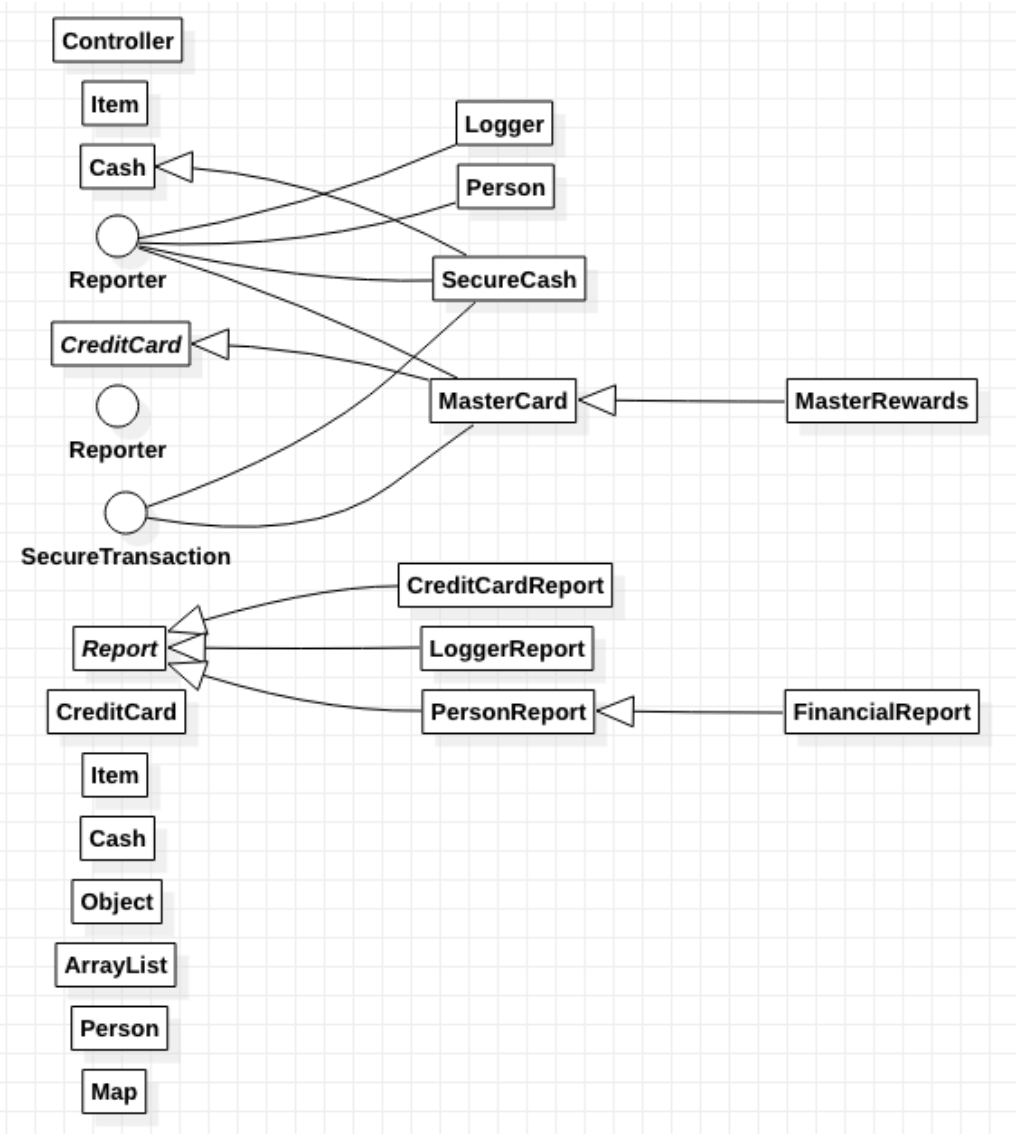
Important:
Do not copy someone's code. You may be randomly selected to explain your code. If you cannot explain your code, you will get a zero for the project grade. If you are stuck please ask me. I will gladly help you. Remember the goal is to learn. ☺

Package Structure

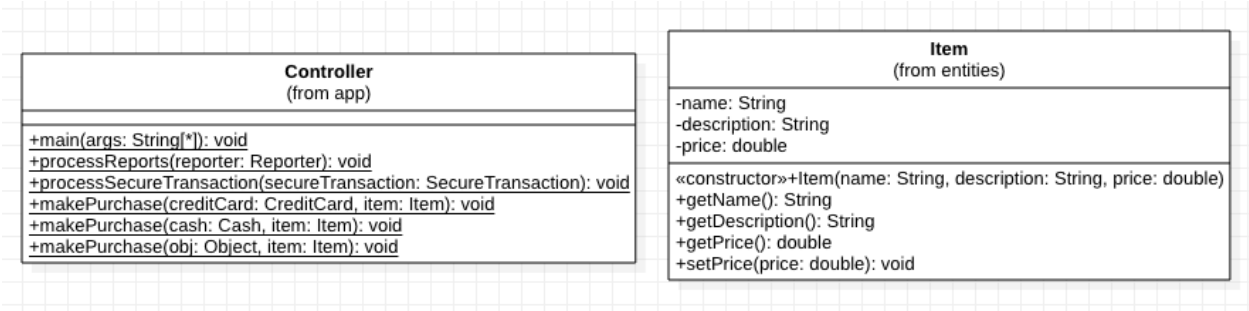


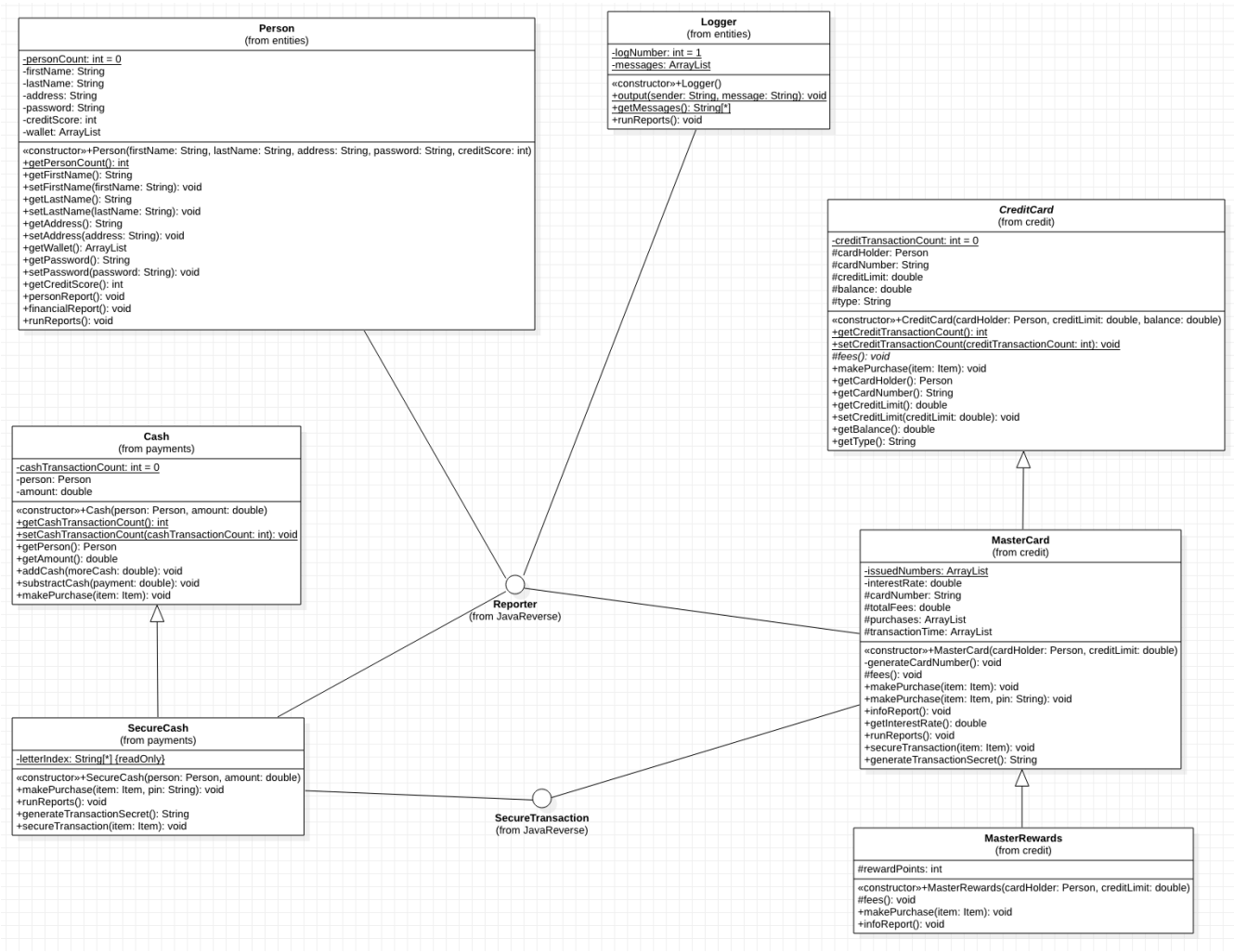
UML

Hierarchy Structure



UML Class Diagrams and Connections




Person
(from entities)
-personCount: int = 0
-firstName: String
-lastName: String
-address: String
-password: String
-creditScore: int
-wallet: ArrayList
«constructor»+Person(firstName: String, lastName: String, address: String, password: String, creditScore: int)
+getPersonCount(): int
+getFirstName(): String
+setFirstName(firstName: String): void
+getLastName(): String
+setLastName(lastName: String): void
+getAddress(): String
+setAddress(address: String): void
+getWallet(): ArrayList
+getPassword(): String
+setPassword(password: String): void
+getCreditScore(): int
+personReport(): void
+financialReport(): void
+runReports(): void
Logger
(from entities)
-logNumber: int = 1
-messages: ArrayList
«constructor»+Logger()
+output(sender: String, message: String): void
+getMessages(): String[*]
+runReports(): void
CreditCard
(from credit)
-creditTransactionCount: int = 0
#cardHolder: Person
#cardNumber: String
#creditLimit: double
#balance: double
#type: String
«constructor»+CreditCard(cardHolder: Person, creditLimit: double, balance: double)
+getCreditTransactionCount(): int
+setCreditTransactionCount(creditTransactionCount: int): void
#fees(): void
+makePurchase(item: Item): void
+getCardHolder(): Person
+getCardNumber(): String
+getCreditLimit(): double
+setCreditLimit(creditLimit: double): void
+getBalance(): double
+getType(): String
Cash
(from payments)
-cashTransactionCount: int = 0
-person: Person
-amount: double
«constructor»+Cash(person: Person, amount: double)
+getCashTransactionCount(): int
+setCashTransactionCount(cashTransactionCount: int): void
+getPerson(): Person
+getAmount(): double
+addCash(moreCash: double): void
+substractCash(payment: double): void
+makePurchase(item: Item): void
Reporter
(from JavaReverse)
MasterCard
(from credit)
-issuedNumbers: ArrayList
-interestRate: double
#cardNumber: String
#totalFees: double
#purchases: ArrayList
#transactionTime: ArrayList
«constructor»+MasterCard(cardHolder: Person, creditLimit: double)
-generateCardNumber(): void
#fees(): void
+makePurchase(item: Item): void
+makePurchase(item: Item, pin: String): void
+infoReport(): void
+getInterestRate(): double
+runReports(): void
+secureTransaction(item: Item): void
+generateTransactionSecret(): String
SecureCash
(from payments)
-letterIndex: String[*] {readOnly}
«constructor»+SecureCash(person: Person, amount: double)
+makePurchase(item: Item, pin: String): void
+runReports(): void
+generateTransactionSecret(): String
+secureTransaction(item: Item): void
SecureTransaction
(from JavaReverse)
MasterRewards
(from credit)
#rewardPoints: int
«constructor»+MasterRewards(cardHolder: Person, creditLimit: double)
#fees(): void
+makePurchase(item: Item): void
+infoReport(): void

Steps you need to do:

Step1: (Controller.java)

```
System.out.println("//////////////////////////////////////////////");
System.out.println("Section 1");
System.out.println("//////////////////////////////////////////////");

// STEP 1


       // create four items instance
       // with the following data

       /*
       item 1  data -> "Mitutoyo 513-403-10E",
                       "DIAL TEST INDICATOR, BASIC SET, STANDARD, .008 IN, . 0001 IN GRAD, WHITE DIAL ",
                       139.16

       item 2 data -> "Garmin Forerunner 935",
                      "Premium GPS running/triathlon watch with wrist-based heart rate ",
                      284.49



       item 3 data -> "CROC Classic Clog",
                      "Iconic clog that started a comfort revolution around the world",
                      40.99

       item4 data -> "Adidas Ultra Boost PB",
                     "Offers a plush, yet bouncy ride with an updated upper is built for speed.",
                     105.95

       item5 data -> "Macbook Pro 13",
                     "1.4GHz quad-core 8th-generation Intel Core i5 processor",
                     1399.00

       // YOUR CODE HERE


       // create 3 person instances
       // with the following data

        person1 -> "Ava", "Davis", "704 Brickell Ave", "Superman", 760
        person2 -> "John", "Smith", "340 S.W. 13 Street", "myPassword123", 680
        person3 -> "Bill", "Johnson", "100 Ocean Dr. S.W. ", "Mom12345", 530

        // do you have passwords like this ..... :)

       // YOUR CODE HERE

//you will not have any output from this step see project document output section

/*


Console output from this step
```

```
//////////////////////////////////////////////
Section 1
//////////////////////////////////////////////
```

Step2: (Controller.java)

```
System.out.println("");
System.out.println("//////////////////////////////////////////////");
System.out.println("Section 2");
System.out.println("//////////////////////////////////////////////");

// GOTO THE MASTERCARD CLASS AND WRITE THE CODE FOR SECTION 2.1 FIRST


        // Setting up Ava's wallet
        // Create cash instance for Ava -> person1 with $400
        // make sure that you wrote section 2.1 code before write this code
        // Create Master Card instance for Ava -> person1 with credit limit of $3500
        // YOUR CODE HERE

        // add the cash and master card instance to Ava's wallet
        // YOUR CODE HERE

        // runner a financial report on Ava
        //YOUR CODE HERE


        // Setting up John's wallet
        // Create cash instance for John -> person2 with $400
        // Create Master Card instance for John -> person2 with credit limit $500
        // Create Master Reward instance for John -> person2 with credit limit $1500
        // YOUR CODE HERE

        // add the cash, master card and master reward instance to John's wallet
        // YOUR CODE HERE


        // runner a financial report on John
        //YOUR CODE HERE


         // Setting up Bill's wallet
        // Create secure cash instance for Bill -> person3 with $1000
        // Create Master Card instance for Bill -> person3 with credit limit $1000
        // Create Master Reward instance for Bill -> person3 with credit limit $1500
        // YOUR CODE HERE



         // add the secure cash, master card and master reward instance to Bill's wallet
        // YOUR CODE HERE

        // runner a financial report on Bill
        //YOUR CODE HERE
```

Console output from this step

```
//////////////////////////////////////////////
Section 2
//////////////////////////////////////////////

//////////////////////////////////////////////
Section 2.1
//////////////////////////////////////////////


======================================
Person Financial Report
======================================
First Name:                          Ava
Last Name:                           Davis
Address:                             704 Brickell Ave
Wallet:                              Cash     MasterCard
Total Credit Limit:                  3500.00
Credit Balance:                      0.00
Cash:                                400.00


//////////////////////////////////////////////
Section 2.1
//////////////////////////////////////////////

//////////////////////////////////////////////
Section 2.1
//////////////////////////////////////////////


======================================
Person Financial Report
======================================
First Name:                          John
Last Name:                           Smith
Address:                             340 S.W. 13 Street
Wallet:                              Cash     MasterCard        Master Rewards
Total Credit Limit:                  2000.00
Credit Balance:                      0.00
Cash:                                400.00


//////////////////////////////////////////////
Section 2.1
//////////////////////////////////////////////

//////////////////////////////////////////////
Section 2.1
//////////////////////////////////////////////


======================================
Person Financial Report
======================================
First Name:                          Bill
Last Name:                           Johnson
Address:                             100 Ocean Dr. S.W.
Wallet:                              Cash     MasterCard        Master Rewards
Total Credit Limit:                  2500.00
Credit Balance:                      0.00
Cash:                                1000.00
```

Step3: (Controller.java and MasterCard.java)

```
System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 3.0");
System.out.println("////////////////////////////////////////////");

// GOTO THE MasterCard.java file write the code for Section 3.1



System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 3.2");
System.out.println("////////////////////////////////////////////");

            // call the infoReport method on Ava's mastercard
            // YOUR CODE HERE


            // call the infoReport method on John's mastercard
            // YOUR CODE HERE


            // call the infoReport method on Bill's master rewards
            // YOUR CODE HERE
```

Console output from this step

```
////////////////////////////////////////////
Section 3.0
////////////////////////////////////////////

////////////////////////////////////////////
Section 3.2
////////////////////////////////////////////

=======================================
MasterCard Info Report
=======================================
Card Holder:                Ava Davis
Number:
Interest Rate:              10.99%
Credit Limit:               3500.00
Balance:                    0.00
Remaining Credit:           3500.00
Fees:                       0.00
Transaction Count:          0


=======================================
MasterCard Info Report
=======================================
Card Holder:                John Smith
Number:
Interest Rate:              12.50%
Credit Limit:               500.00
Balance:                    0.00
Remaining Credit:           500.00
Fees:                       0.00
Transaction Count:          0


=======================================
Master Rewards Info Report
=======================================
Card Holder:                Bill Johnson
Number:
Interest Rate:              14.99%
Credit Limit:               1500.00
Balance:                    0.00
Remaining Credit:           1500.00
Reward Points:              0
Fees:                       0.00
Transaction Count:          0
```

Step4: (Controller.java)

```
System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 4");
System.out.println("////////////////////////////////////////////");


        //=========================================================
        // people buying stuff
        //=========================================================

        System.out.println("\n");
        System.out.println("-----------------------------------------");
        System.out.println(person1.getFirstName() + " " + person1.getLastName() + " buying");
        System.out.println("-----------------------------------------");
        // buying items
        // Ava is buying stuff
        // use the make Purchase method for the following
        // Use Ava's master card to buy item1
        // Use Ava's master card to buy item2
        // Use Ava's cash to buy item2

       // YOUR CODE HERE


        // look at this difference here
        // finish the code below for Ava to buy item5 using her master card
        // MAKE USE YOU GOT THE RIGHT CARD I WILL CHECK THIS
        // COMPLETED CODE BELOW

        ((MasterCard)person1 ….

         // run the person financial report on Ava
         // YOUR CODE HERE


        System.out.println("\n");
        System.out.println("-----------------------------------------");
        System.out.println(person2.getFirstName() + " " + person2.getLastName() + " buying");
        System.out.println("-----------------------------------------");

        // buying items
        // John is buying stuff
        // use the makePurchase method for the following
        // Use John's master card to buy item1
        // Use John's master card to buy item2
        // Use John's master card to buy item3
        // Use John's master card to buy item4

        // Use John's cash to buy item2

        // Use John's master rewards to buy item1
        // Use John's master rewards to buy item2
        // Use John's master rewards to buy item3
        // Use John's master rewards to buy item4
        // Use John's master rewards to buy item5

        // look at this difference here
        // finish the code below for John to buy item1 using her master rewards
        // MAKE USE YOU GOT THE RIGHT CARD I WILL CHECK THIS
        // COMPLETED CODE BELOW

        ((MasterRewards)person2. …

         // run the person financial report on John
         // YOUR CODE HERE



        System.out.println("\n");
        System.out.println("-----------------------------------------");
        System.out.println(person3.getFirstName() + " " + person3.getLastName() + " buying");
        System.out.println("-----------------------------------------");

         // buying items
        // Bill is buying stuff
        // use the makePurchase method for the following
        // Use Bill's secure cash to buy item1
```

```
        // Use Bill's secure cash to buy item2
        // Use Bill's secure cash to buy item5

        // Use Bill's master card to buy item4

        // Use Bill's master rewards to buy item3


        // run the person financial report on Bill
        // YOUR CODE HERE
```

Console output from this step

```
//////////////////////////////////////////////
Section 4
//////////////////////////////////////////////


----------------------------------------------
Ava Davis buying
----------------------------------------------

[1]:[MC-Davis] -> <06-09-2020|17:19:14>  Charged Mitutoyo 513-403-10E for $139.16

[2]:[MC-Davis] -> <06-09-2020|17:19:14>  Charged Garmin Forerunner 935 for $284.49

[3]:[CASH-Davis] -> <06-09-2020|17:19:14>  Purchased Garmin Forerunner 935 for $284.49

[4]:[MC-Davis] -> <06-09-2020|17:19:14>  Charged Macbook Pro 13 for $1399.0

[5]:[MC-Davis] -> <06-09-2020|17:19:14>  $40.06 fee charged


======================================
Person Financial Report
======================================
First Name:                          Ava
Last Name:                           Davis
Address:                  704 Brickell Ave
Wallet:                              Cash     MasterCard
Total Credit Limit:                  3500.00
Credit Balance:                      1822.65
Cash:                                115.51



----------------------------------------------
John Smith buying
----------------------------------------------

[6]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged Mitutoyo 513-403-10E for $139.16

[7]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged Garmin Forerunner 935 for $284.49

[8]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged CROC Classic Clog for $40.99

[9]:[MC-Smith] -> <06-09-2020|17:19:14>  $11.62 fee charged

[10]:[MC-Smith] -> <06-09-2020|17:19:14> Charge declined due to credit limits

[11]:[CASH-Smith] -> <06-09-2020|17:19:14>  Purchased Garmin Forerunner 935 for $284.49

[12]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged Mitutoyo 513-403-10E for $139.16

[13]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged Garmin Forerunner 935 for $284.49

[14]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged CROC Classic Clog for $40.99

[15]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged Adidas Ultra Boost PB for $105.95

[16]:[MC-Smith] -> <06-09-2020|17:19:14> Charge declined due to credit limits

[17]:[MC-Smith] -> <06-09-2020|17:19:14>  Charged Mitutoyo 513-403-10E for $139.16

[18]:[MC-Smith] -> <06-09-2020|17:19:14>  $22.18 fee charged


======================================
Person Financial Report
======================================
First Name:                          John
Last Name:                           Smith
Address:                  340 S.W. 13 Street
Wallet:                              Cash     MasterCard        Master Rewards
Total Credit Limit:                  2000.00
Credit Balance:                      1174.39
Cash:                                115.51
```

```
-------------------------------------------
Bill Johnson buying
-------------------------------------------

[19]:[CASH-Johnson] -> <06-09-2020|17:19:14>  Purchased Mitutoyo 513-403-10E for $139.16

[20]:[CASH-Johnson] -> <06-09-2020|17:19:14>  Purchased Garmin Forerunner 935 for $284.49

[21]:[CASH-Johnson] -> <06-09-2020|17:19:14>  Do not enough cash to buy Macbook Pro 13 for $1399.0

[22]:[MC-Johnson] -> <06-09-2020|17:19:14>  Charged Adidas Ultra Boost PB for $105.95

[23]:[MC-Johnson] -> <06-09-2020|17:19:14>  Charged CROC Classic Clog for $40.99


======================================
Person Financial Report
======================================
First Name:                           Bill
Last Name:                            Johnson
Address:                   100 Ocean Dr. S.W.
Wallet:                               Cash     MasterCard        Master Rewards
Total Credit Limit:                   2500.00
Credit Balance:                       146.94
Cash:                                 576.35
```

Step5: (Controller.java and others)

```
System.out.println("");
System.out.println("//////////////////////////////////////////////");
System.out.println("Section 5");
System.out.println("//////////////////////////////////////////////");


        //Class Static Info
        System.out.println("");
        System.out.println("--------------------------------------------------");
        System.out.println("Class Statics");
        System.out.println("--------------------------------------------------");

        // complete the system.out.println code to get the right output
        // DO NOT CHANGE THE ACCESS MODIFIER OF THE VARAIBLE IN THE CLASS
        // IF YOU DO THIS YOU WILL GET POINTS TAKEN OFF
        // hint: you may need to add method to the some classes...
        //       you need figure out which classes. Also you may need to add some
        //       code to some constructors maybe and other methods…. ☺
        // COMPLETE CODE BELOW

        System.out.println("Number of Person in System:\t\t\t"
        System.out.println("Number of all Cash type transactions:\t\t"
        System.out.println("Number of all Credit Card type transactions:\t"
        System.out.println("--------------------------------------------------");
```

Console output from this step

```
//////////////////////////////////////////////
Section 5
//////////////////////////////////////////////

--------------------------------------------------
Class Statics
--------------------------------------------------
Number of Person in System:                  3
Number of all Cash type transactions:        4
Number of all Credit Card type transactions:  6
--------------------------------------------------
```

Step6: (Controller.java)

```
System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 6");
System.out.println("////////////////////////////////////////////");


        System.out.println("\n\n");
        System.out.println("--------------------------------------------------");
        System.out.println("Logger Data");
        System.out.println("--------------------------------------------------");

        // I wish to ssee all the messages store in the logger
        // write the code to do this
        // YOUR CODE HERE


        System.out.println("--------------------------------------------------");
```

Console output from this step

```
////////////////////////////////////////////
Section 6
////////////////////////////////////////////

--------------------------------------------------
Logger Data
--------------------------------------------------

======================================
Logger Report
======================================
[1]:[MC-Davis] -> <06-09-2020|18:50:33>  Charged Mitutoyo 513-403-10E for $139.16
[2]:[MC-Davis] -> <06-09-2020|18:50:33>  Charged Garmin Forerunner 935 for $284.49
[3]:[CASH-Davis] -> <06-09-2020|18:50:33>  Purchased Garmin Forerunner 935 for $284.49
[4]:[MC-Davis] -> <06-09-2020|18:50:33>  Charged Macbook Pro 13 for $1399.0
[5]:[MC-Davis] -> <06-09-2020|18:50:33>  $40.06 fee charged
[6]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged Mitutoyo 513-403-10E for $139.16
[7]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged Garmin Forerunner 935 for $284.49
[8]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged CROC Classic Clog for $40.99
[9]:[MC-Smith] -> <06-09-2020|18:50:33>  $11.62 fee charged
[10]:[MC-Smith] -> <06-09-2020|18:50:33> Charge declined due to credit limits
[11]:[CASH-Smith] -> <06-09-2020|18:50:33>  Purchased Garmin Forerunner 935 for $284.49
[12]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged Mitutoyo 513-403-10E for $139.16
[13]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged Garmin Forerunner 935 for $284.49
[14]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged CROC Classic Clog for $40.99
[15]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged Adidas Ultra Boost PB for $105.95
[16]:[MC-Smith] -> <06-09-2020|18:50:33> Charge declined due to credit limits
[17]:[MC-Smith] -> <06-09-2020|18:50:33>  Charged Mitutoyo 513-403-10E for $139.16
[18]:[MC-Smith] -> <06-09-2020|18:50:33>  $22.18 fee charged
[19]:[CASH-Johnson] -> <06-09-2020|18:50:33>  Purchased Mitutoyo 513-403-10E for $139.16
[20]:[CASH-Johnson] -> <06-09-2020|18:50:33>  Purchased Garmin Forerunner 935 for $284.49
[21]:[CASH-Johnson] -> <06-09-2020|18:50:33>  Do not enough cash to buy Macbook Pro 13 for $1399.0
[22]:[MC-Johnson] -> <06-09-2020|18:50:33>  Charged Adidas Ultra Boost PB for $105.95
[23]:[MC-Johnson] -> <06-09-2020|18:50:33>  Charged CROC Classic Clog for $40.99
```

Step7: (Controller.java and Others)

```
System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 7");
System.out.println("////////////////////////////////////////////");


        // USING Interfaces
        System.out.println("\n\n\n");
        System.out.println("--------------------------------------------------");
        System.out.println("REPORTER INTERFACE");
        System.out.println("--------------------------------------------------");

        // Here you are working with interfaces...
        // PAY ATTENTION TO TYPES OF AN OBJECT
        // ask yourself the question what are all types the object you are
        // working with can be


        // processReport is in this class
        // run the process report with Ava (person1)
        // run the process report with the logger object
        // run the process report with Bill's secure cash object
        // run the process report with Ava's master card object

        // YOU WILL NEED TO ADD CODE THE NEED CLASSES IN THIS STEP TO MAKE IT WORK
        // YOU NEED FIGURE OUT WHICH CLASSES
        // Hint: look at the UML Diagrams
        // YOUR CODE HERE
```

Console output from this step

```
////////////////////////////////////////////
Section 7
////////////////////////////////////////////

--------------------------------------------------
REPORTER INTERFACE
--------------------------------------------------


======================================
Person Financial Report
======================================
First Name:                         Ava
Last Name:                          Davis
Address:                            704 Brickell Ave
Wallet:                             Cash     MasterCard
Total Credit Limit:                 3500.00
Credit Balance:                     1822.65
Cash:                               115.51


======================================
Logger Report
======================================
[1]:[MC-Davis] -> <06-09-2020|19:38:07>  Charged Mitutoyo 513-403-10E for $139.16
[2]:[MC-Davis] -> <06-09-2020|19:38:07>  Charged Garmin Forerunner 935 for $284.49
[3]:[CASH-Davis] -> <06-09-2020|19:38:07>  Purchased Garmin Forerunner 935 for $284.49
[4]:[MC-Davis] -> <06-09-2020|19:38:07>  Charged Macbook Pro 13 for $1399.0
[5]:[MC-Davis] -> <06-09-2020|19:38:07>  $40.06 fee charged
[6]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged Mitutoyo 513-403-10E for $139.16
[7]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged Garmin Forerunner 935 for $284.49
[8]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged CROC Classic Clog for $40.99
[9]:[MC-Smith] -> <06-09-2020|19:38:07>  $11.62 fee charged
[10]:[MC-Smith] -> <06-09-2020|19:38:07> Charge declined due to credit limits
[11]:[CASH-Smith] -> <06-09-2020|19:38:07>  Purchased Garmin Forerunner 935 for $284.49
[12]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged Mitutoyo 513-403-10E for $139.16
[13]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged Garmin Forerunner 935 for $284.49
[14]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged CROC Classic Clog for $40.99
[15]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged Adidas Ultra Boost PB for $105.95
[16]:[MC-Smith] -> <06-09-2020|19:38:07> Charge declined due to credit limits
[17]:[MC-Smith] -> <06-09-2020|19:38:07>  Charged Mitutoyo 513-403-10E for $139.16
[18]:[MC-Smith] -> <06-09-2020|19:38:07>  $22.18 fee charged
[19]:[CASH-Johnson] -> <06-09-2020|19:38:07>  Purchased Mitutoyo 513-403-10E for $139.16
[20]:[CASH-Johnson] -> <06-09-2020|19:38:07>  Purchased Garmin Forerunner 935 for $284.49
[21]:[CASH-Johnson] -> <06-09-2020|19:38:07>  Do not enough cash to buy Macbook Pro 13 for $1399.0
[22]:[MC-Johnson] -> <06-09-2020|19:38:07>  Charged Adidas Ultra Boost PB for $105.95
[23]:[MC-Johnson] -> <06-09-2020|19:38:07>  Charged CROC Classic Clog for $40.99
```

```
======================================
MasterCard Info Report
======================================
Card Holder:                Ava Davis
Number:                     7210 7907 7451 1639
Interest Rate:              10.99%
Credit Limit:               3500.00
Balance:                    1822.65
Remaining Credit:           1677.35
Fees:                       40.06
Transaction Count:          3
```

Step8: (Controller.java and Others)

```
System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 8");
System.out.println("////////////////////////////////////////////");


        System.out.println("\n\n\n");
        System.out.println("--------------------------------------------------");
        System.out.println("SECURE INTERFACE");
        System.out.println("--------------------------------------------------");

        // Here you are working with interfaces....
        // PAY ATTENTION TO TYPES OF AN OBJECT
        // ask yourself the question what are all types the object you are
        // working with can be

        // processReport is in this class

        // run the process report with Ava's master card object
        // run the process report with BILL's secure cash object

        // WHY DO THESE NOT WORK
        // processSecureTransaction on the person1 object
        // processSecureTransaction no the logger object

        // why does this work ----> does it implements SecureTransactions
        // processSecureTransaction on the mcReward3 object

        // YOU WILL NEED TO ADD CODE THE NEED CLASSES IN THIS STEP TO MAKE IT WORK
        // YOU NEED FIGURE OUT WHICH CLASSES
        // YOUR CODE HERE


    }// end main()
```

Console output from this step (NOTICE NO PINS IN OUTPUT)

```
--------------------------------------------------
SECURE INTERFACE
--------------------------------------------------
Ava Davis (MASTER CARD)  secure transaction pin:
Bill Johnson (SECURE CASH) secure  transaction pin:
```

Step9: (Secure.java)

```
public String  generateTransactionSecret(){


System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 9");
System.out.println("////////////////////////////////////////////");

        //if you are cash type
        // take the first four letters of the Last name
        // then for each letter index letter to location in the alphabet
        // same index for upper and lower case
        // example
        // Index:  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 ..... 25 26
        // Letter: a b c d e f g h i j  k  l  m  n  o  ....  y  z
        // mod each letter index by 10 to generate pin

        // Example: first Name = Richard
        // W   h   i   t <- first four letters
        // 23  8   9   20 <- index to alphabet location
        // 3   8   9   0 <- %10
        // 3890 <- final pin

         String pin = "";

         //YOUR CODE HERE

        return pin;
     }
```

Console output from this step

```
////////////////////////////////////////////
Section 9
////////////////////////////////////////////
Bill Johnson (SECURE CASH) secure transaction pin:0584
```

Step10: (MasterCard.java)

```
        public String  generateTransactionSecret(){

System.out.println("");
System.out.println("////////////////////////////////////////////");
System.out.println("Section 10");
System.out.println("////////////////////////////////////////////");


        // if you are a credit card type
        // from left to right make the pin from following digits of the card number
        // 1, 5, 9, 13
        // Example
        // 2341 4833 9332 9933   <- card number
        // 2    4    9    9      <- selected number
        // pin is 2499           <- generated pin

        String pin = "";

        // YOUR CODE HERE


        return pin;
     }
```

Console output from this step

```
////////////////////////////////////////////
Section 10
////////////////////////////////////////////
Ava Davis (MASTER CARD)  secure transaction pin:3462

////////////////////////////////////////////
Section 9
////////////////////////////////////////////
Bill Johnson (SECURE CASH) secure  transaction pin:0584
```

Step11:
You are done ☺
Zip your project and upload it using the canvas assignment link.

Hope you found this project a good learning exercise. It is a little overwhelming to look at all this code and make sense of it, but I belief you understand coding a little better after this project. In time you will be very good at this. ^|^