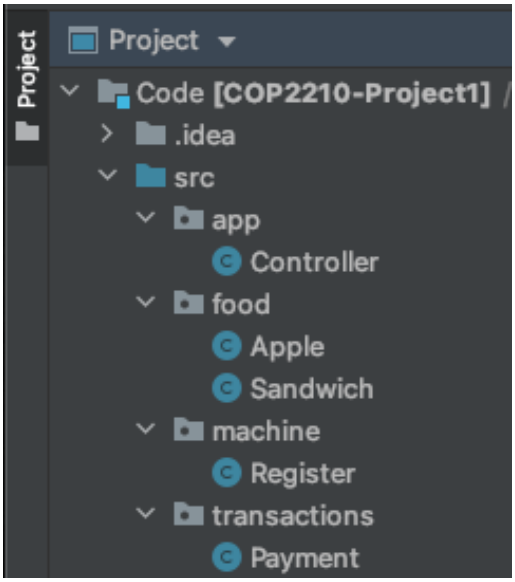
**Important:**
Do not copy someone's code. You may be randomly selected to explain your code. If you cannot explain your code, you will get a zero for the project grade. If you are stuck please ask me. I will gladly help you. Remember the goal is to learn.

**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. Following the order below and you will get the project done.

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




3) Create the Apple.java, Sandwich.java, Register.java and Payment.java files and make sure they are in the correct package.

4) Copy and paste the Payment.java code in the code section of this document into your Payment.java file. This is the entire code. There is no need to modify it.

5) Copy and paste the Register.java code into your Register.java file. Write the code for the Class variables, Instant variables and constructor. Check the UML diagram to do this.

6) Copy and paste the Controller.java code into your Controller.java file. 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.
//============================================================================
```

Write the code for 1 and 2 that are illustrated below. Make sure that you get this working and your output matches the code output. Alignment, $ symbols, and number of digits are important. I will be checking your code output format. Hint you will be going back and fore between the Register.java and Controller.java files. After this your Register's constructor and cashInfo method will be working.

```
package app;

public class Controller {


    public static void main(String[] args) {

        //enter you info here
        System.out.println("================================================");
        System.out.println("COP 2210 - Project 1 Output");
        System.out.println("Your Name here");
        System.out.println("YOUR ID here");
        System.out.println("Your Class Sections here ");
        System.out.println("================================================");
        System.out.println("\n");

        // create a Register varaible named register with
        // 15 dollars
        // 20 quarters
        // 10 dimes
        // 20 nickels
        // 50 pennies
        // YOUR CODE HERE

        // check register cash as a "Manager"
        // call the cashInfo method on the register object with input "Manager"
1 // YOUR CODE HERE

        // check register cash as a "Staff"
        // call the cashInfo method on the register object with input "Staff"
2 // YOUR CODE HERE

… more code
```

7) In your Controller.java file. Write the code for 1, 2, 3, 4, 5, and 6 that are illustrated below. Make sure that you get this working and your output matches the code output. Alignment, $ symbols, and number of digits are important. In 1 and 2 you need to fix the Apple.java file to make this work. Check the UML diagram. Code 3 and 4 you are just calling methods on a Payment object. Since the Payment.java is complete you need not to write any code within the Payment.java file. In 5 you need to write the code buyApple method within the Register.java file. Important the buyApple method can call the

```
private void giveChange(double price, Payment payment)
```

this is the hardest method to write. So, if you do not understand it just put some fill-in code

```
system.out.println( "running giveChange with payment and price")
```

for it until you figure it out. Check the program output to reverse engineer this code. In 6, this code has been written in Step 6.

```
      // create an Apple variable named grannySmith with
        // type = "Granny Smith"
        // weight = 140
        // pricePreUnitWeight = 1.51.   <- this is the correct numbers
1 // YOUR CODE HERE

        // call displayInfo() on the grannySmith object
2 // YOUR CODE HERE

        // create an Payment variable named applePayment1 with
        // 10 dollars
        // 0 quarters
        // 0 dimes
        // 0 nickels
        // 47 pennies
3 // YOUR CODE HERE


        // call displayInfo() on the applePayment1 object
4 // YOUR CODE HERE

        // call the buyApple method on the register object with
        // inputs grannySmith, applePayment1
5 // YOUR CODE HERE

        // check register cash as a "Manager"
        // call the cashInfo method on the register object with input "Manager"
6 // YOUR CODE HERE
```

8) In your Controller.java file. Write the code for 1, 2, 3, 4, 5, and 6 that are illustrated below. There is no new code that needs to be written in Apple.java, Register.java or Payment.java because was done in the earlier steps.

```
        // create an Apple variable named macintosh with
        // type = "Macintosh"
        // weight = 1.70
        // pricePreUnitWeight = 150
1 // YOUR CODE HERE

        // call displayInfo() on the macintosh object
2 // YOUR CODE HERE

        // create an Payment variable named applePayment2 with
        // 0 dollars
        // 2 quarters
        // 0 dimes
        // 0 nickels
        // 0 pennies
 3 // YOUR CODE HERE

        // call displayInfo() on the applePayment2 object
 4 // YOUR CODE HERE

        // call the buyApple method on the register object with
        // inputs macintosh, applePayment2
 5 // YOUR CODE HERE

        // check register cash as a "Manager"
        // call the cashInfo method on the register object with input "Manager"
 6 // YOUR CODE HERE
```

9) Copy the Sandwich.java code into your Sandwich.java file. Write the needed code within your Sandwich.java file i.e. Instant variables, Constructor and displayInfo methods, etc. . Make sure your code matches the UML diagram.

In your sandwich constructor you need to handle all combination of sandwiches and set their prices to the following:

| Meat | Cheese | Veggies | | Price |
|---|---|---|---|---|
| T | T | T | | 7.99 |
| T | T | F | | 7.59 |
| T | F | T | | 7.09 |
| T | F | F | | 6.89 |
| F | T | T | | 5.99 |
| F | T | F | | 5.59 |
| F | F | T | | 5.09 |
| F | F | F | | 4.99 |

Hint: Use an if – else if statement for this.

In 4, you need to write the buySandwich code within your Register.java file. This code will be the same as the buyApple code except you adjust it for buying a sandwich.

```
      // create an Sandwich variable named sandwich with
       // meat = true
       // cheese = true
       // veggies = true
1 // YOUR CODE HERE

       // call displayInfo() on the sandwich object
       // YOUR CODE HERE

       // create an Payment variable named sandwichPayment1 with
       // 5 dollars
       // 2 quarters
       // 1 dimes
       // 1 nickels
       // 2 pennies
2 // YOUR CODE HERE

       // call displayInfo() on the sandwichPayment1 object
3 // YOUR CODE HERE

       // call the buySandwich method on the register object with
       // inputs sandwich, sandwichPayment1
4 // YOUR CODE HERE

       // check register cash as a "Manager"
       // call the cashInfo method on the register object with input "Manager"
5 // YOUR CODE HERE
```

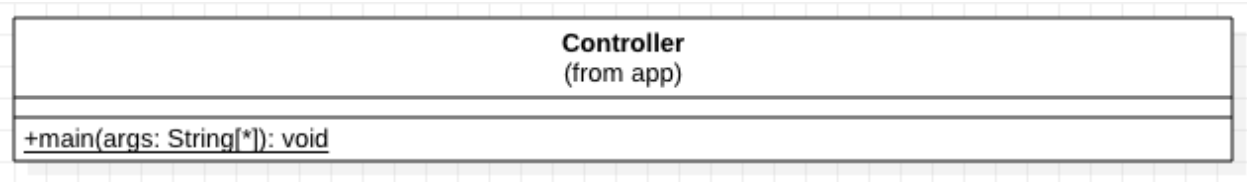
10) You now have all the code in your files to do a large order of sandwiches. Here a person wishes to order every combination of sandwich:

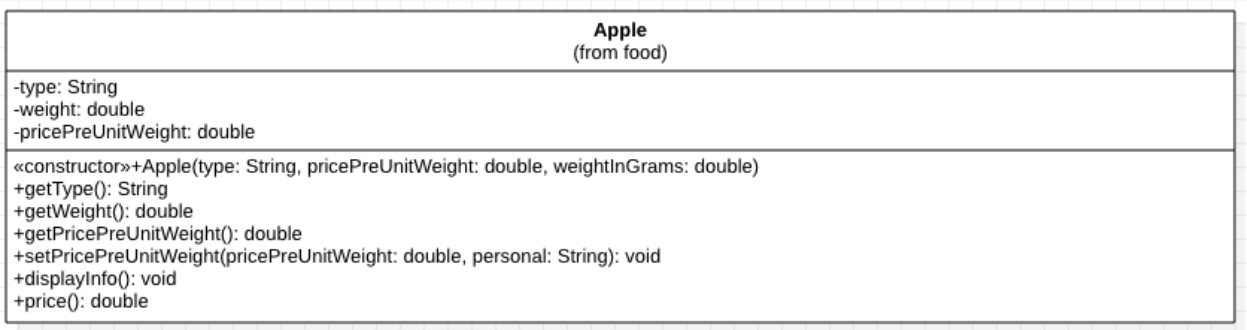
| Meat | Cheese | Veggies | | Price |
|---|---|---|---|---|
| T | T | T | | 7.99 |
| T | T | F | | 7.59 |
| T | F | T | | 7.09 |
| T | F | F | | 6.89 |
| F | T | T | | 5.99 |
| F | T | F | | 5.59 |
| F | F | T | | 5.09 |
| F | F | F | | 4.99 |

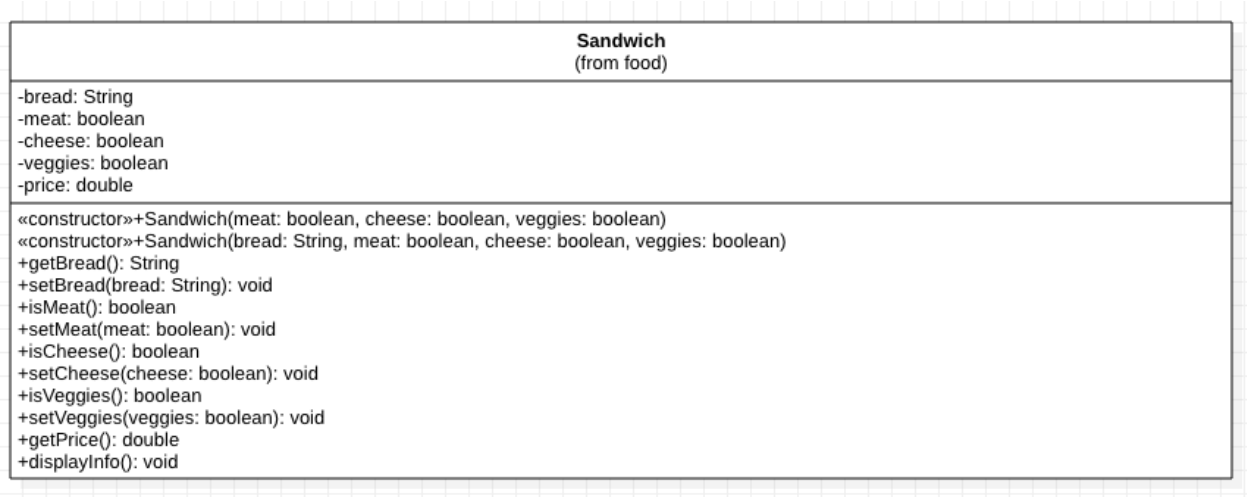
and pay with 10 one dollar bills. In 1, you are the written a for loop that use if- else if statement to create the sandwich, create the payment, buy the sandwich and output the register info. Check the console output at the end of this document. Check your alignment, $ symbols, and number of digits are important. I will be checking your code output format.

```
      // create booleans
        boolean meat = true;
        boolean cheese = true;
        boolean veggies = true;

        System.out.println("");

        // Create all type sandwiches combinations
        // then buy each sandwich by creating a payment with
        // 10 dollars
        // 0 quarters
        // 0 dimes
        // 0 nickels
        // 0 pennies

        // YOU MUST USE A FOR LOOP AND IF ELSE STATMENTS
1 // YOUR CODE HERE
```

## UML Diagrams



**Register**
(from machine)

-registerCounter: int = 0
-REGISTERCODE: String = "US-FL-032020-" {readOnly}
-DOLLARVALUE: double = 1.00 {readOnly}
-QUATERVALUE: double = 0.25 {readOnly}
-DIMEVALUE: double = 0.10 {readOnly}
-NICKELVALUE: double = 0.05 {readOnly}
-PENNYVALUE: double = 0.01 {readOnly}
-registerID: String
-numberOfOneDollarBills: int
-numberOfQuarters: int
-numberOfDimes: int
-numberOfNickels: int
-numberOfPennies: int
-currentTotal: double

«constructor»+Register(numberOfOneDollarBills: int, numberOfQuarters: int, numberOfDimes: int, numberOfNickels: int, numberOfPennies: int)
-cashValue(): double
+cashInfo(personal: String): void
+buyApple(apple: Apple, payment: Payment): void
+buySandwich(sandwich: Sandwich, payment: Payment): void
-giveChange(price: double, payment: Payment): void

**Payment**
(from transactions)

-DOLLARVALUE: double = 1.00 {readOnly}
-QUATERVALUE: double = 0.25 {readOnly}
-DIMEVALUE: double = 0.10 {readOnly}
-NICKELVALUE: double = 0.05 {readOnly}
-PENNYVALUE: double = 0.01 {readOnly}
-numberOfOneDollarBills: int
-numberOfQuarters: int
-numberOfDimes: int
-numberOfNickels: int
-numberOfPennies: int

«constructor»+Payment(numberOfOneDollarBills: int, numberOfQuarters: int, numberOfDimes: int, numberOfNickels: int, numberOfPennies: int)
+getNumberOfOneDollarBills(): int
+getNumberOfQuarters(): int
+getNumberOfDimes(): int
+getNumberOfNickels(): int
+getNumberOfPennies(): int
+displayInfo(): void
+paymentValue(): double

## Controller.java code

```
// PUT YOUR HEADER HERE

package app;

public class Controller {


    public static void main(String[] args) {

        //enter you info here
        System.out.println("=============================================");
        System.out.println("COP 2210 - Project 2 Output");
        System.out.println("Your Name here");
        System.out.println("YOUR ID here");
        System.out.println("Your Class Sections here ");
        System.out.println("=============================================");
        System.out.println("\n");

        // create a Register varaible named register with
        // 15 dollars
        // 20 quarters
        // 10 dimes
        // 20 nickels
        // 50 pennies
        // YOUR CODE HERE

        // check register cash as a "Manager"
        // call the cashInfo method on the register object with input "Manager"
        // YOUR CODE HERE

        // check register cash as a "Staff"
        // call the cashInfo method on the register object with input "Staff"
        // YOUR CODE HERE

        // create an Apple variable named grannySmith with
        // type = "Granny Smith"
        // weight = 1.51
        // pricePreUnitWeight = 140
        // YOUR CODE HERE

        // call displayInfo() on the grannySmith object
        // YOUR CODE HERE

        // create an Payment variable named applePayment1 with
        // 10 dollars
        // 0 quarters
        // 0 dimes
        // 0 nickels
        // 47 pennies
        // YOUR CODE HERE


        // call displayInfo() on the applePayment1 object
        // YOUR CODE HERE

        // call the buyApple method on the register object with
        // inputs grannySmith, applePayment1
        // YOUR CODE HERE

        // check register cash as a "Manager"
        // call the cashInfo method on the register object with input "Manager"
        // YOUR CODE HERE

        // create an Apple variable named macintosh with
        // type = "Macintosh"
        // weight = 1.70
        // pricePreUnitWeight = 150
        // YOUR CODE HERE
```

## Controller.java code continue….

```
    // call displayInfo() on the macintosh object
    // YOUR CODE HERE

    // create a Payment variable named applePayment2 with
    // 0 dollars
    // 2 quarters
    // 0 dimes
    // 0 nickels
    // 0 pennies
    // YOUR CODE HERE

    // call displayInfo() on the applePayment2 object
    // YOUR CODE HERE

    // call the buyApple method on the register object with
    // inputs macintosh, applePayment2
    // YOUR CODE HERE

    // check register cash as a "Manager"
    // call the cashInfo method on the register object with input "Manager"
    // YOUR CODE HERE

    // create an Sandwich variable named sandwich with
    // meat = true
    // cheese = true
    // veggies = true
    // YOUR CODE HERE

    // call displayInfo() on the sandwich object
    // YOUR CODE HERE

    // create an Payment variable named sandwichPayment1 with
    // 5 dollars
    // 2 quarters
    // 1 dimes
    // 1 nickels
    // 2 pennies
    // YOUR CODE HERE

    // call displayInfo() on the sandwichPayment1 object
    // YOUR CODE HERE

    // call the buySandwich method on the register object with
    // inputs sandwich, sandwichPayment1
    // YOUR CODE HERE

    // check register cash as a "Manager"
    // call the cashInfo method on the register object with input "Manager"
    // YOUR CODE HERE


    // create booleans
    boolean meat = true;
    boolean cheese = true;
    boolean veggies = true;

    System.out.println("");

    // Create all type sandwichs combinations
    // then buy each sandwich by creating a payment with
    // 10 dollars
    // 0 quarters
    // 0 dimes
    // 0 nickels
    // 0 pennies

    // YOU MUST USE A FOR LOOP AND IF ELSE STATMENTS
    // YOUR CODE HERE

   }//end main
}//end class
```

## Register.java code

```
package machine;

import food.Apple;
import food.Sandwich;
import transactions.Payment;

public class Register {

    //-------------------------------------------------------
    // Class variables
    // Check UML Diagram
    //-------------------------------------------------------
    private static int registerCounter = 0;
    private static final  String REGISTERCODE =  "US-FL-032020-";
    // YOUR CODE HERE

    //-------------------------------------------------------
    // Instant variables
    // Check UML Diagram
    //-------------------------------------------------------
    // YOUR CODE HERE

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

    public Register(int numberOfOneDollarBills,
                    int numberOfQuarters,
                    int numberOfDimes,
                    int numberOfNickels,
                    int numberOfPennies) {

        // increment registerCounter by one
        // YOUR CODE HERE


        // set registerID to REGISTERCODE + registerCounter
        // YOUR CODE HERE


        currentTotal = 0.0;

       // set the constructor inputs values to the register instant variables
       // hint code:
       // this.numberOfOneDollarBills = numberOfOneDollarBills;
       // YOUR CODE HERE
   }

    //-------------------------------------------------------
    // Utility methods
    // Check UML Diagram
    //-------------------------------------------------------

    private double cashValue(){

        double total = numberOfOneDollarBills * DOLLARVALUE +
                       numberOfQuarters * QUATERVALUE +
                       numberOfDimes * DIMEVALUE +
                       numberOfNickels * NICKELVALUE +
                       numberOfPennies * PENNYVALUE;

        return total;

    }

    public void cashInfo(String personal){

        // if the personal is a Manager
        // then output the cashValue of the register
        // hint code:
        /*
        System.out.println("========================================");
           System.out.println("Register Cash Info");
           System.out.println("========================================");
```

```
        System.out.println("Access Level:\t\t Valid");
        System.out.printf("Cash in the Register:\t $%-15.2f\n", cashValue());
        System.out.printf("Dollars:\t\t %-15d\n", numberOfOneDollarBills);
        System.out.printf("Quarters:\t\t %-15d\n", numberOfQuarters);
    .....
    */

    // else the personal is not a Manager the denied access
    // hint code:
    /*
    System.out.println("=========================================");
       System.out.println("Register Cash Info");
       System.out.println("=========================================");
       System.out.println("Access Level:\t\t Not Valid by " + personal);
       System.out.println("");
    */
   // hint use an if else statement
   // YOUR CODE HERE

    }
}

public void buyApple(Apple apple, Payment payment){
    System.out.println("=========================================");
    System.out.println("Register Buy Apple");
    System.out.println("=========================================");
    System.out.printf("Apple Price:\t\t $%-15.2f\n" , apple.price() );
    System.out.printf("Payment:\t\t $%-15.2f\n" , payment.paymentValue() );

    // check if you have enough payment to buy the apple
    // if your payment is less the apple price calculate the amount short
    // and output to the console
    // Sorry but you do not have enough money to buy the Apple
    // hint code:
    /*
        System.out.printf("You need:\t\t $%-15.2f\n",shortAmount );
        System.out.println("");
        System.out.println("Sorry but you do not have enough money to buy the Apple");
        System.out.println("=========================================");
        System.out.println("\n");
    */
    // else you have enough payment then give change to buyer
    // hence call the giveChange method with the apple price and payment
    // hint: use an if else statement
    // YOUR CODE HERE

}//end buyApple()


public void buySandwich(Sandwich sandwich, Payment payment){
    System.out.println("=========================================");
    System.out.println("Register Buy Sandwich");
    System.out.println("=========================================");

    // check if you have enough payment to buy the sandwich
    // if your payment is less the sandwich price calculate the amount short
    // and output to the console
    // Sorry but you do not have enough money to buy the Sandwich
    // hint code:
    /*
      System.out.printf("Sandwich Price:\t\t $%-15.2f\n" , sandwich.getPrice() );
       System.out.printf("Payment:\t\t $%-15.2f\n" , payment.paymentValue() );
       …
        System.out.println("");
        System.out.println("Sorry but you do not have enough money to buy the Sandwich");
        System.out.println("=========================================");
        System.out.println("\n");
    */
    // else you have enough payment then give change to buyer
    // hence call the giveChange method with the sandwich price and payment
    // hint: use an if else statement
    // YOUR CODE HERE


}//end buySandwich()
```

## Register.java code continued …

```
    private void giveChange(double price, Payment payment){

         // add payment to register
        // hint code:
        //  numberOfOneDollarBills += payment.getNumberOfOneDollarBills();
        // YOUR CODE HERE


        // calculate needed change
        // YOUR CODE HERE

        // rounded to whole number so you can use the % operator for the change
       // example 9.65 becomes 965
        int neededChangeWhole = (int)Math.round(neededChange * 100);

        System.out.printf("Change:\t\t $%-15.2f\n", neededChange);


        // figure out the dollar to give back
        // hint: 965 /100 = 9 because of the int/ int
        // so you have 9 dollars
        // update the remaining change to give back
        // 965 – 900 = 65 this is the cents you need to give back

        // figure out the quarters to give back
        // YOUR CODE HERE

        // update the remaining change to give back
        // YOUR CODE HERE

        // figure out the dimes to give back
        // YOUR CODE HERE

        // update the remaining change to give back
        // YOUR CODE HERE

        // figure out the nickels to give back
        // YOUR CODE HERE

        // update the remaining change to give back
        // YOUR CODE HERE

        // figure out the pennies to give back
        // YOUR CODE HERE


        // give the change back
        // remove the dollars, quarters, dimes, nickels, pennies
        // from the register
        // Hint code:
        // numberOfOneDollarBills -= changeDollars;
        // YOUR CODE HERE


        // output to the console the change:
        // dollars, quarters, dimes, nickels, pennies
        // Hint code:
        // System.out.printf("Dollars:\t\t %-15d\n", changeDollars);
        // System.out.printf("Quaters:\t\t %-15d\n", changeQuaters);
        // YOUR CODE HERE


    }

}//end class
```

## Apple.java code

```
package ?????;

public class Apple {
    //-------------------------------------------------------------
    // Instant variables
    //-------------------------------------------------------------
    // YOUR CODE HERE


    //-------------------------------------------------------------
    // Constructor
    //-------------------------------------------------------------
    public Apple(String type, double pricePreUnitWeight, double weightInGrams) {

        // convert gram into lbs
        weight =  0.00220 * weightInGrams;

        this.type = type;
        this.pricePreUnitWeight = pricePreUnitWeight;
    }

    //-------------------------------------------------------------
    // Setters and Getters
    // IMPORTANT: match the UML Diagram
    //-------------------------------------------------------------


    //-------------------------------------------------------------
    // Utility methods
    // Check UML Diagram
    //-------------------------------------------------------------

    // setPricePreUnitWeight method
    // YOUR CODE HERE


    // displayInfo method
    // use printf and println method see below
    // Hint code:
    /*
    System.out.println("========================================");
    System.out.println("Apple Info");
    System.out.println("========================================");
    System.out.printf("Type:\t\t\t %-15s\n", type);
    System.out.printf("Weight:\t\t\t %-6.4f lbs\n", weight);
    System.out.printf("Price Pre Unit:\t\t %-10.2f\n", pricePreUnitWeight)
    */
    // YOUR CODE HERE

    // price method
    // done for you :)
    public double price(){
        double price = weight * pricePreUnitWeight;
        return price;
    }

}//end of class
```

## Sandwich.java code

```
package ?????;

public class Sandwich {

    //-----------------------------------------------------------
    // Instant variables
    //-----------------------------------------------------------
    // YOUR CODE HERE


    //-----------------------------------------------------------
    // Constructor
    //-----------------------------------------------------------
    public Sandwich(boolean meat, boolean cheese, boolean veggies) {

        bread = "White";

        // if else for all combinations of the sandwichs
        // set the price of each conbination
        // see project document for prices
        // YOUR CODE HERE


        this.meat = meat;
        this.cheese = cheese;
        this.veggies = veggies;

    }

    public Sandwich(String bread,
                    boolean meat,
                    boolean cheese,
                    boolean veggies) {

        this(meat, cheese, veggies);
        this.bread = bread;
    }

    //-----------------------------------------------------------
    // Setters and Getters
    // IMPORTANT: match the UML Diagram
    //-----------------------------------------------------------
    // YOUR CODE HERE



    //-----------------------------------------------------------
    // Utility methods
    // Check UML Diagram
    //-----------------------------------------------------------
    // displayInfo method
    // Hint: code
    /*
        System.out.println("========================================");
        System.out.println("Sandwich Info");
        System.out.println("========================================");
        System.out.printf("Bread:\t\t\t %-15s\n", bread);
        System.out.printf("Meat:\t\t\t %-15b \n", meat)
    */

    // YOUR CODE HERE

    }

}// end of class
```

## Payment.java code this is the entire code just copy and paste

```
package transactions;

public class Payment {

    private static final double DOLLARVALUE = 1.00;
    private static final double QUATERVALUE = 0.25;
    private static final double DIMEVALUE = 0.10;
    private static final double NICKELVALUE = 0.05;
    private static final double PENNYVALUE = 0.01;

    private int numberOfOneDollarBills;
    private int numberOfQuarters;
    private int numberOfDimes;
    private int numberOfNickels;
    private int numberOfPennies;

    public Payment(int numberOfOneDollarBills,
                   int numberOfQuarters,
                   int numberOfDimes,
                   int numberOfNickels,
                   int numberOfPennies) {

        this.numberOfOneDollarBills = numberOfOneDollarBills;
        this.numberOfQuarters = numberOfQuarters;
        this.numberOfDimes = numberOfDimes;
        this.numberOfNickels = numberOfNickels;
        this.numberOfPennies = numberOfPennies;
    }

    public int getNumberOfOneDollarBills() {
        return numberOfOneDollarBills;
    }

    public int getNumberOfQuarters() {
        return numberOfQuarters;
    }

    public int getNumberOfDimes() {
        return numberOfDimes;
    }

    public int getNumberOfNickels() {
        return numberOfNickels;
    }

    public int getNumberOfPennies() {
        return numberOfPennies;
    }

    public void displayInfo(){
        System.out.println("========================================");
        System.out.println("Payment Info");
        System.out.println("========================================");
        System.out.printf("Number of Dollar:\t %-15d\n", numberOfOneDollarBills);
        System.out.printf("Number of Quarters:\t %-15d \n", numberOfQuarters);
        System.out.printf("Number of Dimes:\t %-15d\n", numberOfDimes);
        System.out.printf("Number of Nickels:\t %-15d\n", numberOfNickels );
        System.out.printf("Number of Pennies:\t %-15d\n", numberOfPennies );
        System.out.printf("Total Payment:\t\t $%-15.2f\n", paymentValue() );
        System.out.println("");
    }

    public double paymentValue(){
            double total = numberOfOneDollarBills * DOLLARVALUE +
                       numberOfQuarters * QUATERVALUE +
                       numberOfDimes * DIMEVALUE +
                       numberOfNickels * NICKELVALUE +
                       numberOfPennies * PENNYVALUE;

        return total;
    }
}// end of class
```

## Project Output:

```
================================================
COP 2210 - Project 1 Output
Your Name here
YOUR ID here
Your Class Sections here
================================================



========================================
Register Cash Info
========================================
Access Level:               Valid
Cash in the Register:       $22.50
Dollars:                    15
Quarters:                   20
Dimes:                      10
Nickels:                    20
Pennies:                    50


========================================
Register Cash Info
========================================
Access Level:                Not Valid by Staff



========================================
Apple Info
========================================
Type:                        Granny Smith
Weight:                      0.3080 lbs
Price Pre Unit:              1.51
Price:                       $0.47

========================================
Payment Info
========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          47
Total Payment:              $10.47

========================================
Register Buy Apple
========================================
Apple Price:                $0.47
Payment:                    $10.47
Needed Change:              $10.00
Dollars:                    10
Quarters:                   0
Dimes:                      0
Nickels:                    0
Pennies:                    0

========================================
Register Cash Info
========================================
Access Level:               Valid
Cash in the Register:       $22.97
Dollars:                    15
Quarters:                   20
Dimes:                      10
Nickels:                    20
Pennies:                    97
```

```
==========================================
Apple Info
==========================================
Type:                        Macintosh
Weight:                      0.3300 lbs
Price Pre Unit:              1.70
Price:                       $0.56

==========================================
Payment Info
==========================================
Number of Dollar:            0
Number of Quarters:          2
Number of Dimes:             0
Number of Nickels:           0
Number of Pennies:           0
Total Payment:               $0.50

==========================================
Register Buy Apple
==========================================
Apple Price:                 $0.56
Payment:                     $0.50
You need:                    $0.06

Sorry but you do not have enough money to buy the Apple
==========================================

==========================================
Register Cash Info
==========================================
Access Level:                Valid
Cash in the Register:        $22.97
Dollars:                     15
Quarters:                    20
Dimes:                       10
Nickels:                     20
Pennies:                     97



==========================================
Sandwich Info
==========================================
Bread:                       White
Meat:                        true
Cheese:                      true
Veggies:                     true
Price:                       $7.99

==========================================
Payment Info
==========================================
Number of Dollar:            5
Number of Quarters:          2
Number of Dimes:             1
Number of Nickels:           1
Number of Pennies:           2
Total Payment:               $5.67

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:              $7.99
Payment:                     $5.67
You need:                    $2.32

Sorry but you do not have enough money to buy the Sandwich
==========================================


==========================================
Register Cash Info
==========================================
Access Level:                Valid
Cash in the Register:        $22.97
Dollars:                     15
Quarters:                    20
Dimes:                       10
Nickels:                     20
Pennies:                     97
```

```
============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       true
Cheese:                     true
Veggies:                    true
Price:                      $7.99

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $7.99
Payment:                    $10.00
Change:                     $2.01
Dollars:                    2
Quarters:                   0
Dimes:                      0
Nickels:                    0
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $30.96
Dollars:                    23
Quarters:                   20
Dimes:                      10
Nickels:                    20
Pennies:                    96

============================================================
============================================================


============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       true
Cheese:                     true
Veggies:                    false
Price:                      $7.59

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $7.59
Payment:                    $10.00
Change:                     $2.41
Dollars:                    2
Quarters:                   1
Dimes:                      1
Nickels:                    1
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $38.55
Dollars:                    31
Quarters:                   19
Dimes:                      9
Nickels:                    19
Pennies:                    95

============================================================
============================================================
```

```
============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       true
Cheese:                     false
Veggies:                    true
Price:                      $7.09

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $7.09
Payment:                    $10.00
Change:                     $2.91
Dollars:                    2
Quarters:                   3
Dimes:                      1
Nickels:                    1
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $45.64
Dollars:                    39
Quarters:                   16
Dimes:                      8
Nickels:                    18
Pennies:                    94

============================================================
============================================================


============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       true
Cheese:                     false
Veggies:                    false
Price:                      $6.89

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $6.89
Payment:                    $10.00
Change:                     $3.11
Dollars:                    3
Quarters:                   0
Dimes:                      1
Nickels:                    0
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $52.53
Dollars:                    46
Quarters:                   16
Dimes:                      7
Nickels:                    18
Pennies:                    93

============================================================
============================================================
```

```
============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       false
Cheese:                     true
Veggies:                    true
Price:                      $5.99

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $5.99
Payment:                    $10.00
Change:                     $4.01
Dollars:                    4
Quarters:                   0
Dimes:                      0
Nickels:                    0
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $58.52
Dollars:                    52
Quarters:                   16
Dimes:                      7
Nickels:                    18
Pennies:                    92

============================================================
============================================================


============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       false
Cheese:                     true
Veggies:                    false
Price:                      $5.59

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $5.59
Payment:                    $10.00
Change:                     $4.41
Dollars:                    4
Quarters:                   1
Dimes:                      1
Nickels:                    1
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $64.11
Dollars:                    58
Quarters:                   15
Dimes:                      6
Nickels:                    17
Pennies:                    91

============================================================
============================================================
```

```
============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       false
Cheese:                     false
Veggies:                    true
Price:                      $5.09

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $5.09
Payment:                    $10.00
Change:                     $4.91
Dollars:                    4
Quarters:                   3
Dimes:                      1
Nickels:                    1
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $69.20
Dollars:                    64
Quarters:                   12
Dimes:                      5
Nickels:                    16
Pennies:                    90

============================================================
============================================================

============================================================
============================================================
==========================================
Sandwich Info
==========================================
Bread:                      White
Meat:                       false
Cheese:                     false
Veggies:                    false
Price:                      $4.99

==========================================
Payment Info
==========================================
Number of Dollar:           10
Number of Quarters:         0
Number of Dimes:            0
Number of Nickels:          0
Number of Pennies:          0
Total Payment:              $10.00

==========================================
Register Buy Sandwich
==========================================
Sandwich Price:             $4.99
Payment:                    $10.00
Change:                     $5.01
Dollars:                    5
Quarters:                   0
Dimes:                      0
Nickels:                    0
Pennies:                    1

==========================================
Register Cash Info
==========================================
Access Level:               Valid
Cash in the Register:       $74.19
Dollars:                    69
Quarters:                   12
Dimes:                      5
Nickels:                    16
Pennies:                    89

============================================================
============================================================
```