```
/*A new automatic teller machine has been designed by Faculty of
Engineering and Technology for Mami Amara and a program is needed
to make the machine function properly. The machine sells fried eggs,
bananas, beverages, plantain chips and cookies.*/

/*Assumptions: 1- This Atm programme assumes only one currency in use, (FCFA)
/*                          2- This Atm programme assumes the user input an amount in the machine and
type exactly that amount (without errors)
*/
import java.util.Scanner;

public class ATM{

        private Keypad keypad;
        private Pay pay;
        private Dispenser product;
        private Screen screen;
        private MamiAmara mamiAmara;

        private static final int EGGS = 0;
        private static final int BANANAS = 1;
        private static final int BEVERAGES = 2;
        private static final int CHIPS = 3;
        private static final int COOKIES = 4;

        public ATM (){
                Screen screen = new Screen();
                Keypad keypad = new Keypad();
                Pay pay = new Pay();
                Dispenser product = new Dispenser();
        }
        public void run () {

                while(true){

                        int request, numProduct;
                        double price[] = {200, 100, 300, 200, 500};
                        String menu[] = {"fried_eggs", "bananas", "beverages",
"plantain_chips", "cookies" };

                        screen.displayMessageLine("Welcome");
                        screen.displayMessageLine("Select the number representing your
choice\n");
```

```
                    screen.displayMessageLine("1-> Fried eggs\t200F\n2->
Bananas\t100F\n3-> Beverages\t300F\n4-> Plantain chips\t200F\n5->
Cookies\t500F\n6-> MamiAmara queries...");
                    screen.displayMessage("Enter your choice:  ");

                    request = keypad.getInput();
                    while(request <= 5 && request >= 1){
                        //display the price and take amount
                        System.out.printf("The price of %s is %d",menu[request-1]
,price[request-1]);
                        System.out.printf("What amount of %s do you need? available
amount is %d.\n Type 0 to return. ", menu[request-1]);
                        numProduct = keypad.getInput();
                        if (product.isSufficientProductAvailable(numProduct, request)
&& numProduct > 0) {
                            screen.displayMessageLine("Product availabe waiting
for your cash...");
                            int cashIn = keypad.getInput();
                            product.dispenseProduct(price[request-1], cashIn,
numProduct, request-1);

                            System.out.printf("Thanks for choosing us and please
enjoy your %s",menu[request-1] );

                        }
                        else if(!product.isSufficientProductAvailable(numProduct,
request) && numProduct > 0)//if requestnum >item return to main menu
                            screen.displayMessageLine("The amount requested is
not availabe. ");
                        else if (numProduct == 6) {
                            screen.displayMessageLine("Welcome Mami Amara");
                            screen.displayMessageLine("please Enter your machine
code : ");
                            int code = keypad.getInput();
                            MamiAmara mamiAmara =  new MamiAmara(code);
                            mamiAmara.amaraMenu();
                            mamiAmara.amaraRun();
                        }
                    }
                }
        }
}
```

```
//this class will handle the point 6 on the main menu

public class MamiAmara{

	private Keypad keypad;
	private Screen screen;
	private Dispenser product;

	public MamiAmara(int code){
		Screen screen = new Screen();
		Keypad keypad = new Keypad();

		if (code == 1111) {
			amaraMenu();
		}
	}

	//display a menu for mami amara
	public void amaraMenu(){
		screen.displayMessageLine("1. See cash in machine\n2. See quantity of
product in machine\n3. Change machine code");
	}

	//this method execut the choosen option
	public void amaraRun(){
		int menu = keypad.getInput();
		switch(menu){
			case 1: screen.diplayMessageLine("The cash in machine is %d ",
amountInMachine);
				break;
			case 2: screen.displayMessageLine("%d fried_eggs\n%d Bananas\n
%d beverages\n%d Plantain_chips\n%d Cookies", cnt1, cnt2, cnt3, cnt4, cnt5);
				break;
			case 3:
				break;
		}
	}
}
```

//the definition of the main class

*/*A new automatic teller machine has been designed by Faculty of*
Engineering and Technology for Mami Amara and a program is needed
to make the machine function properly. The machine sells fried eggs,
bananas, beverages, plantain chips and cookies./*

/*Assumptions: 1- This Atm programme assumes only one currency in use, (FCFA)
/* 2- This Atm programme assumes the user input an amount in the machine and type exactly that amount (without errors)
****/***

```
public class Main{
	public static void main(String[] args) {
		ATM theATM = new ATM();//creat a new ATM object.
		theATM.run();//call run method
	}
}//End
```

/*defining the keypad class*/

```
import java.util.Scanner;

public class Keypad{
	private Scanner input;
//counstructor
	public Keypad(){
		input = new Scanner(System.in);
	}

	public int getInput(){
		return input.nextInt();
	}
}
```

/*defining the Dispenser class*/

```
public class Dispenser{
	public final static int INITIAL_COUNT = 100;
	public int amountInMachine;//number of hundred francs coins
	//cnt1, 2, 3, 4, 5 holds the number of available items for each product
	public int cnt1;
	public int cnt2;
	public int cnt3;
	public int cnt4;
	public int cnt5;

	public Dispenser(){
		amountInMachine = INITIAL_COUNT;
		cnt1 = 10;
		cnt2 = 10;
		cnt3 = 10;
		cnt4 = 10;
		cnt5 = 10;
	}

	public void dispenseProduct(double price, double amountSentIn, int
amountProduct, int product){
		if (isSufficientProductAvailable(amountProduct, product)) {
			switch(product){
			case 1: cnt1-=amountProduct; break;//update the amount of product
left
			case 2: cnt2-=amountProduct; break;
			case 3: cnt3-=amountProduct; break;
			case 4: cnt4-=amountProduct; break;
			case 5: cnt5-=amountProduct; break;
			}
			dispenseChange(price, amountProduct, amountSentIn);
		}

	}

	public void dispenseChange(double price, int amountProduct, double
amountSentIn){
		double totalPrice = price *amountProduct;//compute exact customer price
		double dispenseChange = amountSentIn - totalPrice;//compute the change
		if (isSufficientCashAvailable(price, amountProduct, amountSentIn)) {
```

```
                amountInMachine -= dispenseChange/100;
                amountInMachine += amountSentIn;// reset count to hold the remaining
amount after giving the change
            }
        }

        public boolean isSufficientCashAvailable(double price, int amountProduct, double
amountSentIn){
            double totalPrice = price *amountProduct;//compute exact customer price
            double dispenseChange = amountSentIn - totalPrice;//compute the change

            if (amountInMachine >= dispenseChange)
                return true;//enough cash available
            else
                return false;//unsuficient cash cant give change
        }

        public boolean isSufficientProductAvailable(int amountProduct, int product){
            switch(product){
                case 1: if(cnt1>amountProduct)return true; else return false; break;
                case 2: if(cnt2>amountProduct)return true; else return false; break;
                case 3: if(cnt3>amountProduct)return true; else return false;break;
                case 4: if(cnt4>amountProduct)return true; else return false; break;
                case 5: if(cnt5>amountProduct)return true; else return false; break;
            }
            return false;
        }
}
```

/*The screen class*/

```
public class Screen{
	public Screen(){}

	//display message on screen
	public void displayMessage(String message){
		System.out.print(message);
	}

	//display message with new line
	public void displayMessageLine(String message){
		System.out.println(message);
	}

	//display price amount
	public void displayPrice(double price){
		System.out.printf("%.2f", price);
	}
}
```

Requirements

==========

1- A new automatic teller machine has been designed by Faculty of Engineering and Technology for Mami Amara and a program is needed to make the machine function properly. The machine sells fried eggs, bananas, beverages, plantain chips and cookies. You will write a program in java for the Mami Amara machine.

The requirements for the program are:

a) show the customer the different products sold by the Amara machine

b) let the customer make the selection

c) show the customer the cost of the item selected

d) Accept the money from the customer

e) Release the item

The Amara machine has three main components: a built-in cash register, several dispensers to hold and release the products, and the Amara machine itself. The register has some cash on hand, it accepts the amount from the customer, and if the amount entered is more than the cost of the item, then—if possible—it returns the change. The cash register should also be able to show the Amara machine's owner the amount of money in the register at any given time. The dispenser releases the selected item if it is not empty. It should show the number of items in the dispenser and the cost of the item.

When the program executes, it must do the following:

a) show the different products sold by the Amara machine

b) get the selection

c) if the selection is valid and the dispenser corresponding to the selection is not empty, sell the product.

Furthermore, these instructions must be displayed after processing each selection (except when exiting the program), so that the user need not remember what to do if he or she wants to buy additional items. Once the user makes the appropriate selection, the Amara machine must act accordingly. If the user opts to buy an available product, the Amara machine should show the cost of the product and ask the user to deposit the money. If the money deposited is at least the cost of the item, the Amara machine should sell the item and display an appropriate message

Analysis

==========

Input:

---

1. Choosen Product

2. Amount of product
3. Customer Money

Output:

---

1. Products for sale
2. Price of selected product
3. Products sold
4. Balance if any

Design

==========

step1:

---

*Display a menu
*Prompt the user to enter his selection
*Store the selection and display the price of the selected product
*Prompt the user to enter the amount of the product he needs

Step2:

---

*Check amount of product available if sufficient

Step3:

---

*If step3 is true, prompt the user to enter Cash
*If cash require change, check for sufficient ballance, if ok, give balance to user
*Dispence product

Step4:

---

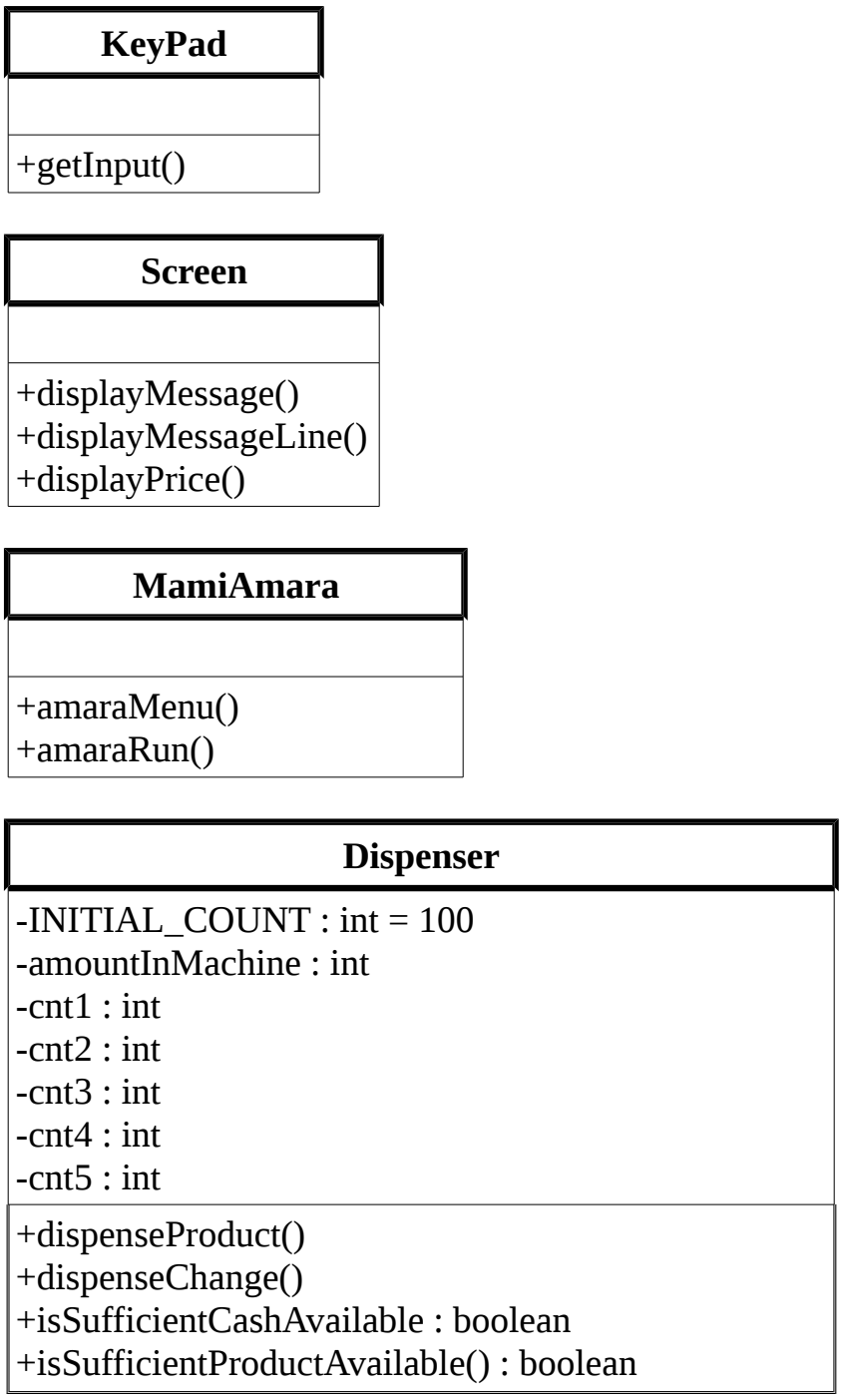
*Go back to main menu

Implementation

==========

UML diagram with attributes and operations

ATM
+run()


KeyPad
+getInput()
Screen
+displayMessage()
+displayMessageLine()
+displayPrice()
MamiAmara
+amaraMenu()
+amaraRun()
Dispenser
-INITIAL_COUNT : int = 100
-amountInMachine : int
-cnt1 : int
-cnt2 : int
-cnt3 : int
-cnt4 : int
-cnt5 : int
+dispenseProduct()
+dispenseChange()
+isSufficientCashAvailable : boolean
+isSufficientProductAvailable() : boolean