```
import java.util.Scanner;

public class Product {

    String productName = null;//get combobox and set
    double productPrise = 0.00;//get database and set
    final double includeQuantity = 80.00;//get database
    final double includeQuantityMix = 25.00;//get database
    int orderQuantity = 0; //get textbox and set
    double stockProduct = 500000.00;//get database
    double stockLow = 0; //calculate and set all add items add list
    String mixProNumArr[] = new String[3];//mix order name list
    int[] mixProQuaArr = new int[3];//mix order quantity
    double[] mixProPrise = new double[3]; //mix order prise
    int arrIndex = 0;//three mix array index set
    int multyArr = 3; //3D array first index
    String allProductSales[][] = new String[multyArr][50]; // big array used story one custermer order store
    int allArrIndex = 0; //used 3D array value set

    void setProductNameAndQuantity(String flaverName, String quantity, String priseProduct) {
        /* database prise combox and textbox data set action text box focus lost*/
        try {
```

```
        if (flaverName.isEmpty() || quantity == null || priseProduct == null) {
          System.out.println("Enter Product details");
        } else {
          int quntityInt = Integer.parseInt(quantity);
          double priseDouble = Double.parseDouble(priseProduct);
          this.productName = flaverName;
          this.orderQuantity = quntityInt;
          this.productPrise = priseDouble;
        }

      } catch (NumberFormatException e) {
        System.out.println("Invalid integer input");
      } catch (Exception ee) {
        System.out.println("set product err");
      }

    }

    public void addIceCream() {
      /* click adding ice orderlist */
      if (productName != null && orderQuantity != 0 && productPrise != 0.00) {
        double singleTotal = orderQuantity * productPrise;
        System.out.println(productName + " " + orderQuantity + " " + singleTotal);
```

```
     stockLow += (includeQuantity * orderQuantity);
     stockProduct -= (includeQuantity * orderQuantity);
     System.out.println("stock : " + stockProduct + " sales product stock " + stockLow);

     allProductSales[0][allArrIndex] = productName;
     allProductSales[1][allArrIndex] = "" + orderQuantity;
     allProductSales[2][allArrIndex] = "" + singleTotal;
     allArrIndex += 1;

     allDataClear();
   } else if (mixProNumArr.length > 0 && mixProQuaArr.length <= 3) {
     String proNamCancat = "";
     int countOrder = 0;
     double totalPrise = 0;
     double includeTotal = 0;
     for (int i = 0; i < mixProPrise.length; i++) {

       proNamCancat += mixProNumArr[i] + "_";
       countOrder += mixProQuaArr[i];
       totalPrise += mixProQuaArr[i] * mixProPrise[i];

       allProductSales[0][allArrIndex] = mixProNumArr[i];
       allProductSales[1][allArrIndex] = "" + mixProQuaArr[i];
```

```
      allProductSales[2][allArrIndex] = "" + mixProPrise[i];
      allArrIndex += 1;

    }

    System.out.println(proNamCancat + " " + countOrder + " " + totalPrise);
    stockLow += (includeQuantityMix * countOrder);
    stockProduct -= (includeQuantityMix * countOrder);
    System.out.println("stock : " + stockProduct + " sales product stock " + stockLow);

    allDataClear();
  } else {
    System.out.println("enter product details");
  }
}

public void mixIceCream() {

  if (productName != null && orderQuantity != 0 || productPrise != 0.00) {
    mixProNumArr[arrIndex] = productName;
    mixProQuaArr[arrIndex] = orderQuantity;

    if (arrIndex == 2) {
      System.out.println("Cant add anyMore product");

    }

    allDataClear();
  } else {
    System.out.println("enter product details");
  }

}

public void removeIceCream() {
  if (productName != null && orderQuantity != 0 && productPrise != 0.00) {
    double singleTotal = orderQuantity * productPrise;
    System.out.println(productName + " " + orderQuantity + "  -" + singleTotal);
    stockLow -= (includeQuantity * orderQuantity);
    stockProduct += (includeQuantity * orderQuantity);
    System.out.println("stock : " + stockProduct + " sales product stock " + stockLow);

    allProductSales[0][allArrIndex] = productName;
    allProductSales[1][allArrIndex] = "" + orderQuantity;
    allProductSales[2][allArrIndex] = "-" + singleTotal;
    allArrIndex += 1;

    allDataClear();
  } else if (mixProNumArr.length > 0 && mixProQuaArr.length <= 3) {
    String proNamCancat = "";
    int countOrder = 0;
    double totalPrise = 0;
    double includeTotal = 0;
    for (int i = 0; i < mixProPrise.length; i++) {

      proNamCancat += mixProNumArr[i] + "_";
      countOrder += mixProQuaArr[i];
      totalPrise += mixProQuaArr[i] * mixProPrise[i];

      allProductSales[0][allArrIndex] = mixProNumArr[i];
      allProductSales[1][allArrIndex] = "" + mixProQuaArr[i];
      allProductSales[2][allArrIndex] = "" + mixProPrise[i];
      allArrIndex += 1;

    }

    System.out.println(proNamCancat + " " + countOrder + "  -" + totalPrise);
    stockLow -= (includeQuantityMix * countOrder);
    stockProduct += (includeQuantityMix * countOrder);
    System.out.println("stock : " + stockProduct + " sales product stock " + stockLow);

    allDataClear();
  } else {
    System.out.println("enter product details");
  }
}

public void buyProduct() { //add all order

  System.out.println("");
  System.out.println("");
  System.out.println("");
  System.out.println("Buy Items");

  for (int x = 0; x < multyArr; x++) {

    for (int y = 0; y < allProductSales[x].length; y++) {

      System.out.print("<< : " + allProductSales[x++][y] + " ");
      System.out.print(allProductSales[x++][y] + " ");
      System.out.print(allProductSales[x++][y] + " : >> ");
```

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

        }

    }
    allSalseClearBigArr();

}

public void allDataClear() {
    arrIndex += 1;
    productName = null;
    orderQuantity = 0;
    productPrise = 0.00;
    arrIndex = 0;

    for (int a = 0; mixProNumArr.length > a; a++) {
        mixProNumArr[a] = null;
        mixProPrise[a] = 0.00;
        mixProQuaArr[a] = 0;
    }
}
```

```
public void allSalseClearBigArr() {
    allArrIndex = 0;
    for (int x = 0; x < multyArr; x++) {

        for (int y = 0; y < allProductSales[x].length; y++) {

            allProductSales[x++][y] = null;
            allProductSales[x++][y] = null;
            allProductSales[x++][y] = null;

        }

    }
}

public static void main(String[] args) {

    Product mainClassOne=new Product();
    Scanner sr = new Scanner(System.in);
    while (true) {
        System.out.println("0=set ( flaverName,quantity,priseProduct )");
        System.out.println("1=add ");
        System.out.println("2=addMix ");
        System.out.println("3=remove ");
        System.out.println("4=buy ");


        System.out.println("");
        System.out.print("Give 0-4 Number : ");
        int inSw = sr.nextInt();
        switch (inSw) {


            case 0:


              System.out.print("Flaver Name : ");
              String nn = sr.next();
              System.out.print("Product quantity : ");
              String qq = sr.next();
              System.out.println("Product prise : ");
              String pp = sr.next();


              //setProductNameAndQuantity(nn, qq, pp);
              break;
            case 1:
              //addIceCream();
              break;
            case 2:
```

```
                    //mixIceCream();
                    break;
                case 3:
                    //removeIceCream();
                    break;
                case 4:
                   // buyProduct();
                    break;
                default:
                    System.out.print("invalid index 0-4 give");
                    System.out.println("");
                    System.out.println("");

            }
        }
    }
}
```