LAB # 07

1) Task # 01: Write the classes below containing the given instance variables and methods, following the inherited hierarchy:

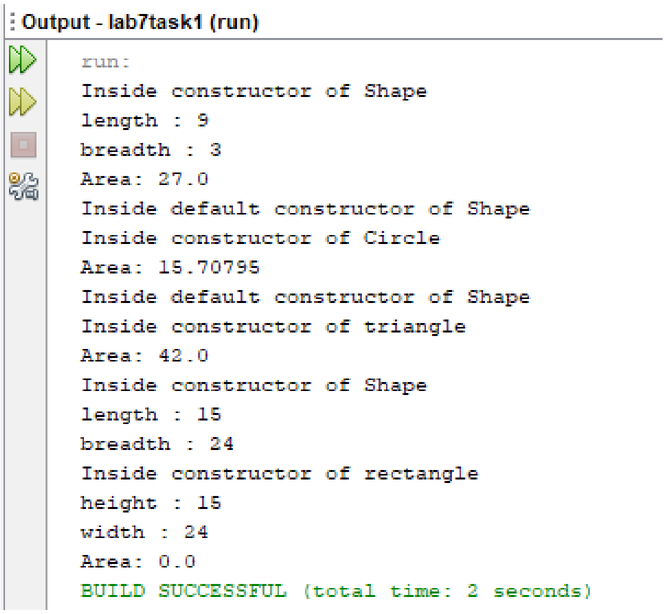
Solution:

```
package lab7task1;

/**

 *

 * @author Sameen Arshad

 */

public class Lab7task1 {

   public static void main(String[] args) {

        shapes obj = new shapes(9 ,3);

        obj.onAreaChange();


        circle obj2 = new circle(5 ,3.14159);

        obj2.onAreaChange();


        triangle obj3 = new triangle(6,7);

        obj3.onAreaChange();


        rectangle obj4 = new rectangle(15,24);

        obj4.onAreaChange();

   }


}


package lab7task1;

/**
```

```
 *
 * @author Sameen Arshad
 */
public class shapes {
   private double area;
   private int length;
   private int breadth;

   public int getBreadth() {
    return breadth;
  }
   public int getLength() {
    return length;
  }
   public void setBreadth(int i) {
    breadth = i;
  }
  public void setLength(int i) {
    length = i;
  }
   shapes(){
      length = 0;
      breadth = 0;
    System.out.println("Inside default constructor of Shape ");
   }
   shapes(int len , int brdth){
      length = len;
     breadth = brdth;
```

```
       System.out.println("Inside constructor of Shape ");

       System.out.println("length : " + length);

       System.out.println("breadth : " + breadth);


    }

    double getArea(){

        return breadth *length;

    }

    protected void onAreaChange(){

        double area = length *breadth;

        System.out.println("Area: " + area);

    }



}



package lab7task1;

/**

 *

 * @author Sameen Arshad

 */

public class circle extends shapes {

    private double radius;

    private double pi = 3.14159;


    circle(double radius, double pi){

         System.out.println("Inside constructor of Circle ");
        this.radius = radius;

        this.pi = pi;


    }

    public void setRadius(double radius){

        this.radius = radius;

    }

    public void onAreaChange(){

        double area = radius *pi;

        System.out.println("Area: " + area);

    }

}

package lab7task1;

/**

 *

 * @author Sameen Arshad

 */

public class triangle extends shapes{

    private double base;

    private double height;

    private double setBase;




    triangle(double base , double height){

        System.out.println("Inside constructor of triangle ");

        this.base = base;

        this.height = height;

    }
    public void setBase(double base){
        this.base = base;
    }
    public void setHeight(double height){
        this.height = height;
    }
     public void onAreaChange(){
         double area = base * height;
         System.out.println("Area: " + area);
     }

}

package lab7task1;

/**
 *
 * @author Sameen Arshad
 */
public class rectangle extends shapes {

    private double height;
    private double width;

    rectangle(int h, int w) {
        super(h, w);
        System.out.println("Inside constructor of rectangle ");
        System.out.println("height : " + h);
        System.out.println("width : " + w);


    }
    rectangle() {
        //System.out.println("Inside default constructor of rectangle ");


    }
    public void setHeight(double height) {
        this.height = height;
    }
    public void setWidth(double width) {
        this.width = width;
    }


    public void onAreaChange() {
        double area = height *width;
        System.out.println("Area: " + area);
    }


}
```

Output:



```
run:
Inside constructor of Shape
length : 9
breadth : 3
Area: 27.0
Inside default constructor of Shape
Inside constructor of Circle
Area: 15.70795
Inside default constructor of Shape
Inside constructor of triangle
Area: 42.0
Inside constructor of Shape
length : 15
breadth : 24
Inside constructor of rectangle
height : 15
width : 24
Area: 0.0
BUILD SUCCESSFUL (total time: 2 seconds)
```

Task # 02: Write a program that inherits a class named Alien and Pirates from a parent class Human. The human class has its own features like, Human can sleep, walk, talk etc. the Alien and Pirates class inheriting these functionalities as well as they have their characteristics, thus explaining the concepts of inheritance.

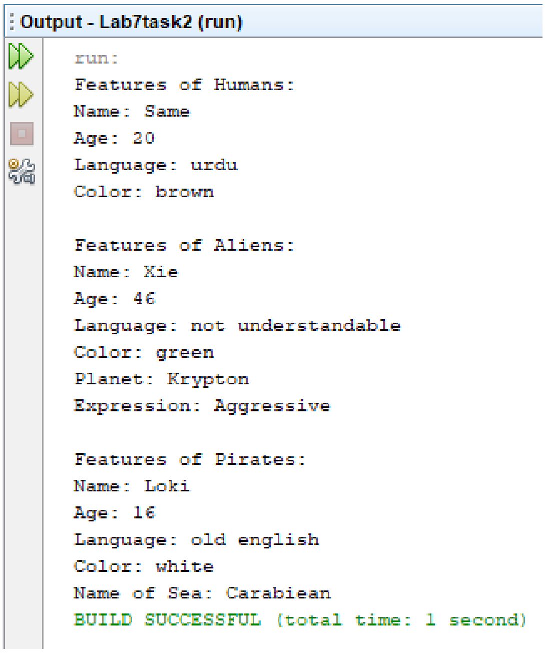
Solution:

```
package lab7task2;


/**
 *
 * @author Sameen Arshad
 */
public class Lab7task2 {


  public static void main(String[] args) {
    Human h = new Human("Same", 20, "urdu", "brown");
    h.Features();
    System.out.println("");
```

```
        aliens a1 = new aliens("Krypton", "Aggressive");
        a1.name = "Xie";
        a1.age = 46;
        a1.skin_colour = "green";
        a1.language = "not understandable";
        a1.alienFeatures();
        System.out.println("");

        pirates p1 = new pirates("Carabiean");
        p1.name = "Loki";
        p1.age = 16;
        p1.skin_colour = "white";
        p1.language = "old english";
        p1.piratesFeatures();

    }

}
package lab7task2;

/**
 *
 * @author Sameen Arshad
 */
public class Human {

    protected String name;
    protected int age;
    protected String language;

    protected String skin_colour;


    public Human() {

        this.name = "";

        this.age = 0;

        this.skin_colour = "";

        this.language = "";

    }


    public Human(String n, int a, String l, String sc) {

        this.name = n;

        this.age = a;

        this.language = l;

        this.skin_colour = sc;

    }


    public void Features() {

        System.out.println("Features of Humans:");

        System.out.println("Name: " + name);

        System.out.println("Age: " + age);

        System.out.println("Language: " + language);

        System.out.println("Color: " + skin_colour);


    }


}

package lab7task2;
```

```
/**
 *
 * @author Sameen Arshad
 */
public class aliens extends Human {

    private String planet;
    private String expression;

    public aliens() {
        this.planet = "";
        this.expression = "";
    }

    public aliens(String p, String e) {
        this.planet = p;
        this.expression = e;
    }

    public void alienFeatures() {
        System.out.println("Features of Aliens:");
        System.out.println("Name: " + name);
        System.out.println("Age: " + age);
        System.out.println("Language: " + language);
        System.out.println("Color: " + skin_colour);
        System.out.println("Planet: " + planet);
        System.out.println("Expression: " + expression);
    }


}
package lab7task2;


/**
 *
 * @author Sameen Arshad
 */
public class pirates extends Human {


    private String sea_name;


    public pirates(String sn) {
        this.sea_name = sn;
    }


    public void piratesFeatures() {
        System.out.println("Features of Pirates:");
        System.out.println("Name: " + name);
        System.out.println("Age: " + age);
        System.out.println("Language: " + language);
        System.out.println("Color: " + skin_colour);
        System.out.println("Name of Sea: " + sea_name);


    }
}
```

Output:



```
run:
Features of Humans:
Name: Same
Age: 20
Language: urdu
Color: brown

Features of Aliens:
Name: Xie
Age: 46
Language: not understandable
Color: green
Planet: Krypton
Expression: Aggressive

Features of Pirates:
Name: Loki
Age: 16
Language: old english
Color: white
Name of Sea: Carabiean
BUILD SUCCESSFUL (total time: 1 second)
```

Task # 03: Write a program that inherits a class named Produce, Cosmetics, Pharmacy, electronic Item and Cloth from a parent class Item. The Item class has its own features like, name and price etc. the Child classes inheriting these functionalities as well as they have their characteristics, thus explaining the concepts of inheritance. Chile classes like Produce, can have their own child classes i.e., Frozen and Fresh.

Solution:

```
package lab7task3;

/**

 *

 * @author Sameen Arshad

 */

public class Lab7task3 {

    public static void main(String[] args) {

        System.out.println("Cosmetics Collections");

       Cosmetics c1 = new Cosmetics("VIP", "red", "Semi_liquid", "10g");
```

```
c1.name="Lipstick";

c1.brand="lexus";

c1.made_in="Pakistan";

c1.TM_number=1430;

c1.price=50;

c1.display();

System.out.println("");

System.out.println("Pharmacy Info");

Pharmacy p1=new Pharmacy("Antibiotics", "25mg", "intoxication");

p1.name="Penadol";

p1.brand="epixu";

p1.made_in="England";

p1.price=10;

p1.TM_number=88815;

p1.display();

System.out.println("");

System.out.println("Electronic Equipments");

Electronics e1 = new Electronics("automatic", "yes", 150);

e1.name="Washing machine";

e1.price=25000;

e1.brand="Dawlance";

e1.made_in="Ireland";

e1.TM_number=5255;

e1.display();

System.out.println("");

System.out.println("Cloths collections");

clothes C2 = new clothes("Lawn", "Summer", "upto you");

C2.name="New COllection";

C2.price=2500;

C2.brand="Ethnic";
```

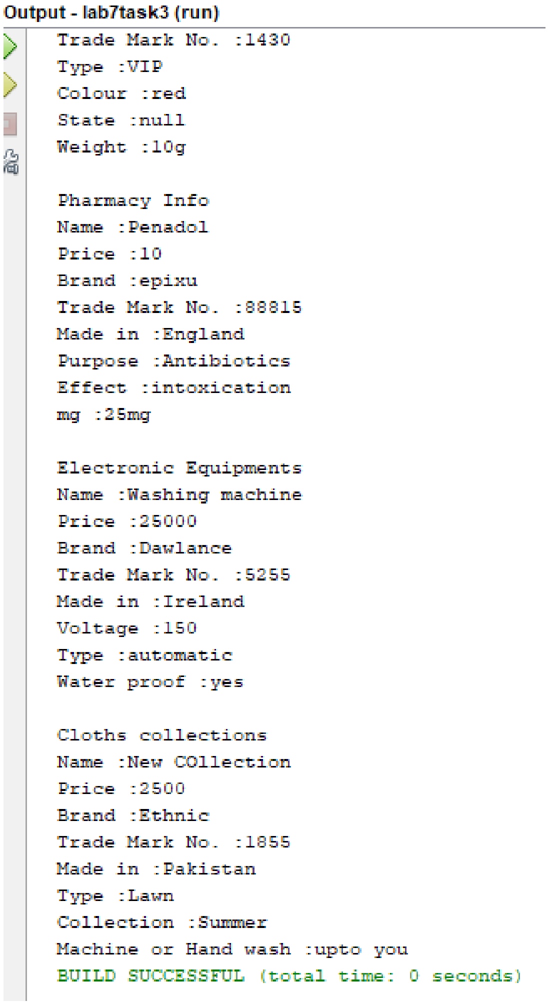
```
      C2.TM_number=1855;

      C2.made_in="Pakistan";

      C2.display();

   }


}

package lab7task3;


/**

 *

 * @author Sameen Arshad

 */

public class items {


   protected String name;

   protected int price;

   protected String brand;

   protected int TM_number;

   protected String made_in;


   items() {

   }


   public void display() {

      System.out.println("Name :" + name);

      System.out.println("Price :" + price);

      System.out.println("Brand :" + brand);

      System.out.println("Trade Mark No. :" + TM_number);

      System.out.println("Made in :" + made_in);

   }
```

```
}
package lab7task3;
/**
 *
 * @author Sameen Arshad
 */
public class Cosmetics extends items{
    private String type;
    private String colour;
    private String state;
    private String weight;
    Cosmetics() {}
    public Cosmetics(String type,String col,String State,String W)
    {
        this.type=type;
        this.colour=col;
        this.state=state;
        this.weight=W;
    }
     public void display()
    {
        System.out.println("Name :"+name);
        System.out.println("Price :"+price);
        System.out.println("Brand :"+brand);
        System.out.println("Trade Mark No. :"+TM_number);
        System.out.println("Type :"+type);
        System.out.println("Colour :"+colour);
        System.out.println("State :"+state);
        System.out.println("Weight :"+weight);
    }


}
package lab7task3;
/**
 *
 * @author Sameen Arshad
 */
public class Pharmacy extends items {
    private String purpose;
    private String mg;
    private String effect;
    Pharmacy(){}
    public Pharmacy(String purpose,String mg,String effect)
    {
        this.purpose=purpose;
        this.effect=effect;
        this.mg=mg;
    }
    public void display()
    {
        System.out.println("Name :"+name);
        System.out.println("Price :"+price);
        System.out.println("Brand :"+brand);
        System.out.println("Trade Mark No. :"+TM_number);
        System.out.println("Made in :"+made_in);
        System.out.println("Purpose :"+purpose);
        System.out.println("Effect :"+effect);
        System.out.println("mg :"+mg);
    }
```

```
}
package lab7task3;
/**
 *
 * @author Sameen Arshad
 */
public class Electronics extends items {
    private int volts;
        private String automatic_or_manual;
        private String water_res;
        public Electronics(){


        }
        public Electronics(String type,String waterproof,int volts)
        {
            this.volts=volts;
            this.water_res=waterproof;
            this.automatic_or_manual=type;
        }
     public void display()
    {
        System.out.println("Name :"+name);
        System.out.println("Price :"+price);
        System.out.println("Brand :"+brand);
        System.out.println("Trade Mark No. :"+TM_number);
        System.out.println("Made in :"+made_in);
        System.out.println("Voltage :"+volts);
        System.out.println("Type :"+automatic_or_manual);
        System.out.println("Water proof :"+water_res);
    }
```

}

Output:



Output - lab7task3 (run)

```
Trade Mark No. :1430
Type :VIP
Colour :red
State :null
Weight :10g

Pharmacy Info
Name :Penadol
Price :10
Brand :epixu
Trade Mark No. :88815
Made in :England
Purpose :Antibiotics
Effect :intoxication
mg :25mg

Electronic Equipments
Name :Washing machine
Price :25000
Brand :Dawlance
Trade Mark No. :5255
Made in :Ireland
Voltage :150
Type :automatic
Water proof :yes

Cloths collections
Name :New COllection
Price :2500
Brand :Ethnic
Trade Mark No. :1855
Made in :Pakistan
Type :Lawn
Collection :Summer
Machine or Hand wash :upto you
BUILD SUCCESSFUL (total time: 0 seconds)
```

Task # 04: Write a program that inherits a class named Pakistani, BBQ, Chines, Fast Food and Beverages etc. from a parent class Cuisines. The Cuisines class has its own features like, name, quantity and price etc. the Child classes inheriting these functionalities as well as they have their characteristics, thus explaining the concepts of inheritance. Child classes can have their own child classes.

Solution:

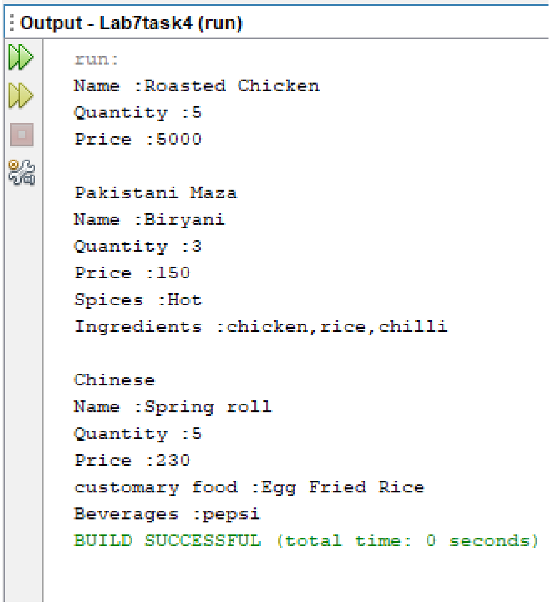
```
package lab7task4;


/**
 *
 * @author Sameen Arshad
 */
public class Lab7task4 {


    public static void main(String[] args) {
        Cuisines cu1 = new Cuisines("Roasted Chicken", 5, 5000);
        cu1.display();
        System.out.println("");
        System.out.println("Pakistani Maza");
        Pakistani p1 = new Pakistani("Hot", "chicken,rice,chilli");
        p1.name = "Biryani";
        p1.price = 150;
        p1.quantity = 3;
        p1.display();
        System.out.println("");
        System.out.println("Chinese");
        Chinese c1 = new Chinese("Egg Fried Rice", "pepsi");
        c1.name = "Spring roll";
        c1.price = 230;
        c1.quantity = 5;
        c1.display();
    }


}
package lab7task4;
```

```
/**
 *
 * @author Sameen Arshad
 */
public class Cuisines {

    protected String name;
    protected int quantity;
    protected int price;

    public Cuisines() {
    }

    public Cuisines(String name, int quant, int pric) {
        this.name = name;
        this.price = pric;
        this.quantity = quant;
    }

    public void display() {
        System.out.println("Name :" + name);
        System.out.println("Quantity :" + quantity);
        System.out.println("Price :" + price);

    }
}
package lab7task4;

/**
```

```
 *
 * @author Sameen Arshad
 */
public class Pakistani extends Cuisines {

    String spices;
    String ingridients;

    public Pakistani() {
    }

    public Pakistani(String spicies, String ingred) {
        this.spices = spicies;
        this.ingridients = ingred;
    }

    public void display() {
        System.out.println("Name :" + name);
        System.out.println("Quantity :" + quantity);
        System.out.println("Price :" + price);
        System.out.println("Spices :" + spices);
        System.out.println("Ingredients :" + ingridients);
    }
}
package lab7task4;

/**
 *
 * @author Sameen Arshad
 */
```

```
public class Chinese extends Cuisines {


   String customary_food;

   String beverages;


   Chinese() {

   }


   Chinese(String cutom, String bever) {

      this.beverages = bever;

      this.customary_food = cutom;

   }


   public void display() {

      System.out.println("Name :" + name);

      System.out.println("Quantity :" + quantity);

      System.out.println("Price :" + price);

      System.out.println("customary food :" + customary_food);

      System.out.println("Beverages :" + beverages);

   }

}
```

Output:



```
run:
Name :Roasted Chicken
Quantity :5
Price :5000

Pakistani Maza
Name :Biryani
Quantity :3
Price :150
Spices :Hot
Ingredients :chicken,rice,chilli

Chinese
Name :Spring roll
Quantity :5
Price :230
customary food :Egg Fried Rice
Beverages :pepsi
BUILD SUCCESSFUL (total time: 0 seconds)
```

Task # 05: Write code according to given guide.You must draw a class diagram first to start writing your code.

Consider a superclass Items which models customer's purchases. This class has:

- **Two private instance variables name (String) and unitPrice (double).**
- **One constructor to initialize the instance variables.**
- **A default constructor to initialize name to "no item", and unitPrice to 0. use this()**
- **A method getPrice that returns the unitPrice.**
- **Accessor and mutator methods.**
- **A toString method to return the name of the item followed by @ symbol, then the unitPrice.**

Consider two subclasses WeighedItem and CountedItem. WeighedItem has an additional instance variable weight (double) in Kg while CountedItem has an additional variable quantity (int) both private.

- **Write an appropriate constructor for each of the classes making use of the constructor of the superclass in defining those of the subclasses.**
- **Override getPrice method that returns the price of the purchasedItem based on its unit price and weight (WeighedItem), or quantity (CountedItem). Make use of getPrice of the superclass**

- **Override also toString method for each class making use of the toString method of the superclass in defining those of the subclasses.**

toString should return something that can be printed on the receipt.

For example

Banana @ 3.00   1.37 Kg   4.11 PKR (in case of WeighedItem class)

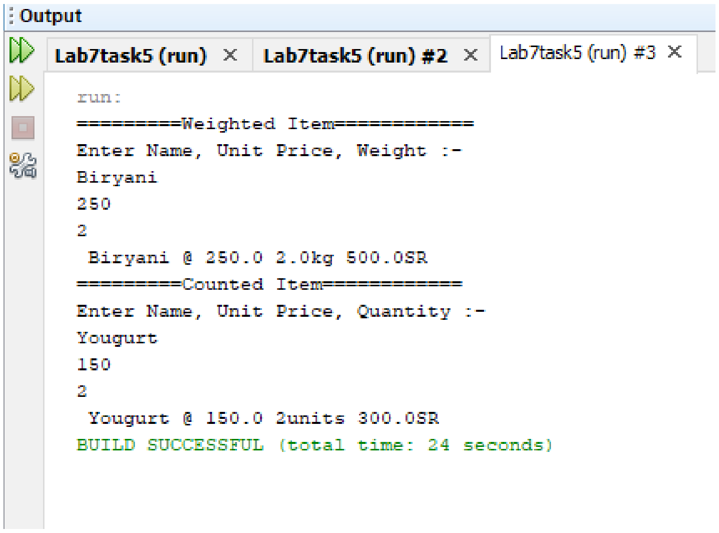
Pens @ 4.5   10 units   45 PKR (in case of CountedItem class)

Solution:

```
package lab7task5;
import java.util.Scanner;
/**
 *
 * @author Sameen Arshad
 */

public class Lab7task5 {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("=========Weighted Item===========");
        System.out.println("Enter Name, Unit Price, Weight :-");
        WeightedItem w = new WeightedItem();
        w.setData(scan.next(), scan.nextDouble());
        w.setWeight(scan.nextDouble());
        System.out.println(" " + w.toString());
        System.out.println("=========Counted Item===========");
        System.out.println("Enter Name, Unit Price, Quantity :-");
        CountedItem c = new CountedItem();

        c.setData(scan.next(), scan.nextDouble());

        c.setQuantity(scan.nextInt());

        System.out.println(" " + c.toString());


    }

}

package lab7task5;


/**

 *

 * @author Sameen Arshad

 */

public class Items {


    private String name;

    private double unitPrice;


    public Items() {

        this.name = "No Item";

        this.unitPrice = 0;

    }


    public Items(String name, double unitPrice) {

        this.name = name;

        this.unitPrice = unitPrice;

    }
    public void getData() {
        System.out.println("Name :- " + name);
        System.out.println("Unit Price :- " + unitPrice);
    }

    public void setData(String name, double unitprice) {
        this.name = name;
        this.unitPrice = unitprice;
    }

    public double getPrice() {
        return unitPrice;
    }

    public String toString() {
        return name + " @ " + unitPrice;
    }

}
package lab7task5;
/**
 *
 * @author Sameen Arshad
 */
public class WeightedItem extends Items {

    private double weight;
    public WeightedItem() {
    }

    public WeightedItem(double weight) {
        this.weight = weight;
    }

    public double getWeight() {
        return weight;
    }

    public void setWeight(double weight) {
        this.weight = weight;
    }

    public double getPrice() {
        return weight * super.getPrice();
    }

    public String toString() {
        return super.toString() + " " + weight + "kg " + getPrice() + "SR";
    }
}
package lab7task5;

/**
 *
 * @author Sameen Arshad
 */
public class CountedItem extends Items {

    private int quantity;

    public CountedItem() {
    }

    public CountedItem(int quantity) {
        this.quantity = quantity;
    }

    public int getQuantity() {
        return quantity;
    }

    public void setQuantity(int quantity) {
        this.quantity = quantity;
    }

    public double getPrice() {
        return quantity * super.getPrice();
    }

    public String toString() {
        return super.toString() + " " + quantity + "units " + getPrice() + "SR";
    }
```

}

Output:



```
run:
=========Weighted Item============
Enter Name, Unit Price, Weight :-
Biryani
250
2
 Biryani @ 250.0 2.0kg 500.0SR
=========Counted Item============
Enter Name, Unit Price, Quantity :-
Yougurt
150
2
 Yougurt @ 150.0 2units 300.0SR
BUILD SUCCESSFUL (total time: 24 seconds)
```