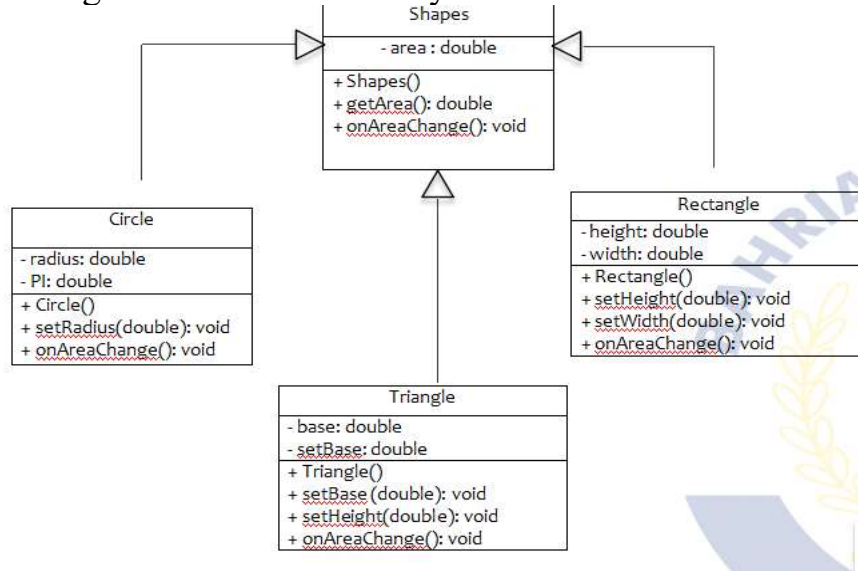


LAB # 07

Task No 01: Write the classes below containing the given instance variables and methods, following the inherited hierarchy:

**Code:****Main:**

```

package lab07task01;

public class Lab07task01 {

    public static void main(String[] args) {

        System.out.println("          Area of Circle          ");
        Circle c1 = new Circle();
        c1.setradius(4.2);
        c1.areaonchange();
        System.out.println("          Area of Rectangular          ");
        Rectangle r1 = new Rectangle();
        r1.setheight(20);
        r1.setwidth(19);
        r1.areaonchange();
        System.out.println("          Area of Triangle          ");
        Triangle t1 = new Triangle();
        t1.setbase(23);
        t1.setheight(40);
        t1.areaonchange();
    }
}
  
```

Shapes (Parent):

```

package lab07task01;

public class Shapes {
    double area;
  
```

```
public Shapes() {  
    area = 0;  
}  
public double getarea(double area) {  
    this.area = area;  
    return area;  
}  
public void areaonchange() {  
    System.out.println("area is = " + area);  
}  
}
```

Circle (Child):

```
package lab07task01;  
  
public class Circle extends Shapes {  
    private double radius;  
    private double pi;  
    public Circle() {  
        radius = 0;  
        pi = 3.142;  
    }  
    public void setradius(double radius) {  
        this.radius = radius;  
    }  
    public double getradius() {  
        return radius;  
    }  
    public void areaonchange() {  
        System.out.println("radius = " + radius);  
        area = pi * (radius * radius);  
        super.areaonchange();  
    }  
}
```

Rectangle (Child):

```
package lab07task01;  
  
public class Rectangle extends Shapes {  
    private double height;  
    private double width;  
    public Rectangle() {  
        height = 0;  
        width = 0;  
    }  
    public void setheight(double height) {  
        this.height = height;  
    }  
    public double getheight() {  
        return height;  
    }  
    public void setwidth(double width) {  
        this.width = width;  
    }  
    public double getwidth() {  
        return width;  
    }  
}
```

```

    }
    public void areaonchange() {
        System.out.println("Height is = " + height);
        System.out.println("Width is = " + width);
        area = height * width;
        super.areaonchange();
    }
}

```

Triangle (Child):

```

package lab07task01;

public class Triangle extends Shapes {
    private double base;
    private double height;
    public Triangle() {
        base = 0;
        height = 0;
    }
    public void setbase(double base) {
        this.base = base;
    }
    public void setheight(double height) {
        this.height = height;
    }
    public double getbase() {
        return base;
    }
    public double getheight() {
        return height;
    }
    public void areaonchange() {
        System.out.println("Base = " + base);
        System.out.println("Hieght = " + height);
        area = 0.5 * height * base;
        super.areaonchange();
    }
}

```

Output:

```

--- exec-maven-plugin:3.1.0:exec (default-cli) @ Lab07task01 ---
        Area of Circle
radius = 4.2
area is = 55.42488
        Area of Rectangular
Height is = 20.0
Width is = 19.0
area is = 380.0
        Area of Triangle
Base = 23.0
Hieght = 40.0
area is = 460.0
-----
BUILD SUCCESS
-----

```

Task No 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.

Code:**Main:**

```
package lab07task02;

public class Lab07task02 {

    public static void main(String[] args) {

        Human human = new Human("John");
        human.sleep();
        human.walk();
        human.talk();

        Alien alien = new Alien("Zorg", "Xenon");
        alien.sleep();
        alien.walk();
        alien.talk();
        alien.teleport();

        Pirate pirate = new Pirate("Blackbeard", "Queen Anne's Revenge");
        pirate.sleep();
        pirate.walk();
        pirate.talk();
        pirate.plunder();
    }
}
```

Human (Parent):

```
package lab07task02;

public class Human {

    private String name;

    public String getName() {
        return name;
    }
    public Human(String name) {
        this.name = name;
    }
    public void sleep() {
        System.out.println(name + " is sleeping");
    }
    public void walk() {
        System.out.println(name + " is walking");
    }
    public void talk() {
        System.out.println(name + " is talking");
    }
}
```

}

Alien (Child):

package lab07task02;

public class Alien extends Human {

private String planet;

public Alien(String name, String planet) {

super(name);

this.planet = planet;

}

public void teleport() {

System.out.println(getName() + " is teleporting from planet " + planet);

}

}

Pirate (Child):

package lab07task02;

public class Pirate extends Human {

private String ship;

public Pirate(String name, String ship) {

super(name);

this.ship = ship;

}

public void plunder() {

System.out.println(getName() + " is plundering on ship " + ship);

}

}

Output:

--- exec-maven-plugin:3.1.0:exec (default-cli) @ Lab07task02 ---

John is sleeping

John is walking

John is talking

Zorg is sleeping

Zorg is walking

Zorg is talking

Zorg is teleporting from planet Xenon

Blackbeard is sleeping

Blackbeard is walking

Blackbeard is talking

Blackbeard is plundering on ship Queen Anne's Revenge

BUILD SUCCESS

Total time: 0.934 s

Finished at: 2023-04-04T23:16:42+05:00

Task No 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. Child classes like Produce, can have their own child classes i.e., Frozen and Fresh.

Code:**Main:**

```
package lab07task03;

public class Lab07task03 {

    public static void main(String[] args) {
        Fresh Banana = new Fresh("mango", 60, "Fruit", "29/04/2023");
        Frozen Icecream = new Frozen("XYZ", 15, "Ice Cream", "29/04/2023");
        Cosmetics cosmetics = new Cosmetics("cream", 3000, "XYZ");
        Electronics Laptop = new Electronics("Dell_Latitude", 48000, "Dell");
        Pharmacy Painkiller = new Pharmacy("Panadol", 50, "GSK");
        Cloth cloth = new Cloth("Dress Coat", 8000, "Cotton");
        System.out.println("Top Products:");
        System.out.println("Frozen Product: "+Icecream.getname()+" is of
"+Icecream.getCategory()+" category. It costs "+Icecream.getprice()+" and it expires
on "+Icecream.getexpirydate());
        System.out.println("Fresh Product: "+Banana.getname()+" is of
"+Banana.getCategory()+" category. It costs "+Banana.getprice()+" and it expires on
"+Banana.getexpirydate());
        System.out.println("Cosmetics item: "+cosmetics.getbrand()+"
"+cosmetics.getname()+" costs "+cosmetics.getprice());
        System.out.println("Electronics item: "+Laptop.getname()+" manufactured by
"+Laptop.getmanufacturer()+" costs "+Laptop.getprice());
        System.out.println("Pharmaceutical item: "+Painkiller.getname()+"
manufactured by "+Painkiller.getmanufacturer()+" costs "+Painkiller.getprice());
        System.out.println("Clothes: "+cloth.getname()+" made of
"+cloth.getmaterial()+" costs "+cloth.getprice());
    }
}
```

Items (Parent):

```
package lab07task03;

public class Items {
    private String name;
    private double price;
    protected Items(String name, double price) {
        this.name = name;
        this.price = price;
    }
    protected void setname(String name) {
        this.name = name;
    }
    protected String getname() {
        return name;
    }
}
```

```
    }  
    protected void setprice(double price) {  
        this.price = price;  
    }  
    protected double getprice() {  
        return price;  
    }  
}
```

Product (Child of Items):

```
package lab07task03;  
  
public class Product extends Items {  
    private String category;  
    protected Product(String name, double price, String category) {  
        super(name, price);  
        this.category = category;  
    }  
    protected String getCategory() {  
        return category;  
    }  
    protected void setCategory(String category) {  
        this.category = category;  
    }  
}
```

Fresh (Child of Product):

```
package lab07task03;  
  
public class Fresh extends Product {  
    private String expirydate;  
    public Fresh(String name, double price, String category, String expirydate) {  
        super(name, price, category);  
        this.expirydate = expirydate;  
    }  
    protected void setexpirydate(String brand) {  
        this.expirydate = expirydate;  
    }  
    protected String getexpirydate() {  
        return expirydate;  
    }  
}
```

Frozen (Child of Product):

```
package lab07task03;  
  
public class Frozen extends Product {  
    private String expirydate;  
    public Frozen(String name, double price, String category, String expirydate) {  
        super(name, price, category);  
        this.expirydate = expirydate;  
    }  
    protected void setexpirydate(String brand) {  
        this.expirydate = expirydate;  
    }  
    protected String getexpirydate() {  
        return expirydate;  
    }  
}
```

```
}  
}
```

Cosmetics (Child of Items):

```
package lab07task03;  
  
public class Cosmetics extends Items {  
    private String brand;  
    public Cosmetics(String name, double price, String brand) {  
        super(name, price);  
        this.brand = brand;  
    }  
    protected void setbrand(String brand) {  
        this.brand = brand;  
    }  
    protected String getbrand() {  
        return brand;  
    }  
}
```

Electronics (Child of Items):

```
package lab07task03;  
  
public class Electronics extends Items {  
    private String manufacturer;  
    protected Electronics(String name, double price, String manufacturer) {  
        super(name, price);  
        this.manufacturer = manufacturer;  
    }  
    protected void setmanufacturer(String manufacturer) {  
        this.manufacturer = manufacturer;  
    }  
    protected String getmanufacturer() {  
        return manufacturer;  
    }  
}
```

Pharmacy (Child of Items):

```
package lab07task03;  
  
public class Pharmacy extends Items {  
    private String manufacturer;  
    public Pharmacy(String name, double price, String manufacturer) {  
        super(name, price);  
        this.manufacturer = manufacturer;  
    }  
    protected void setmanufacturer(String manufacturer) {  
        this.manufacturer = manufacturer;  
    }  
    protected String getmanufacturer() {  
        return manufacturer;  
    }  
}
```

Cloth (Child of Items):

```
package lab07task03;  
  
public class Cloth extends Items {
```



```

private String material;
public Cloth(String name, double price, String material) {
    super(name, price);
    this.material = material;
}
protected void setmaterial(String material) {
    this.material = material;
}
protected String getmaterial() {
    return material;
}
}

```

Output:

```

--- exec-maven-plugin:3.1.0:exec (default-cli) @ Lab07task03 ---
Top Products:
Frozen Product: XYZ is of Ice Cream category. It costs 15.0 and it expires on 29/04/2023
Fresh Product: mango is of Fruit category. It costs 60.0 and it expires on 29/04/2023
Cosmetics item: XYZ cream costs 3000.0
Electronics item: Dell_Latitude manufactured by Dell costs 48000.0
Pharmaceutical item: Panadol manufactured by GSK costs 50.0
Clothes: Dress Coat made of Cotton costs 8000.0
-----
BUILD SUCCESS
-----

```

Task No 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.

Code:**Main:**

```

package lab07task04;

public class Lab07task04 {

    public static void main(String[] args) {

        System.out.println("PAKISTANI: ");
        Biryani biryani = new Biryani();
        biryani.setname("Biryani");
        biryani.setquantity(2);
        biryani.setprice(700.0);
        biryani.settype("Spicy");
        biryani.setmeat("Chicken");
        biryani.Display();
        Karahi karahi = new Karahi();
        karahi.setname("Mutton Karahi");
        karahi.setquantity(4);
        karahi.setprice(1500.0);
        karahi.settype("Mild");
        karahi.setspiceLevel("Medium");
    }
}

```

```

        karahi.Display();
        System.out.println("BBQ:");
        BBQ bbq = new BBQ();
        bbq.setname("Beef Kabab");
        bbq.setquantity(6);
        bbq.setprice(800.0);
        bbq.setSauce("Mint Chutney");
        bbq.Display();
        System.out.println("CHINESE:");
        Chinese chinese = new Chinese();
        chinese.setname("Sweet and Sour Chicken");
        chinese.setquantity(4);
        chinese.setprice(1200.0);
        chinese.setStyle("Cantonese");
        chinese.Display();
    }
}

```

Cuisine:

```

package lab07task04;

public class Cuisine {
    private String name, style;
    private int quantity;
    private double price;
    protected void setname(String name) {
        this.name = name;
    }
    protected void setquantity(int quantity) {
        this.quantity = quantity;
    }
    protected void setStyle(String style) {
        this.style = style;
    }
    protected void setprice(double price) {
        this.price = price;
    }
    protected void display() {
        System.out.println("Name: " + name);
        System.out.println("Quantity: " + quantity);
        System.out.println("Price: " + price);
        System.out.println("Style: " + style);
    }
}

```

Pakistani:

```

package lab07task04;

public class Pakistani extends Cuisine {
    private String type;
    protected void settype(String type) {
        this.type = type;
    }
    public void Display() {
        super.display();
        System.out.println("Type: " + type);
    }
}

```

```
    }  
}
```

Biryani:

```
package lab07task04;  
  
public class Biryani extends Pakistani {  
    private String meat;  
    protected void setmeat(String meat) {  
        this.meat = meat;  
    }  
    public void Display() {  
        super.display();  
        System.out.println("Meat Type: " + meat);  
    }  
}
```

Karahi:

```
package lab07task04;  
  
public class Karahi extends Pakistani {  
    String spiceLevel;  
    protected void setspiceLevel(String spiceLevel) {  
        this.spiceLevel = spiceLevel;  
    }  
    public void Display() {  
        super.display();  
        System.out.println("Spice Level: " + spiceLevel);  
    }  
}
```

Chinese:

```
package lab07task04;  
  
public class Chinese extends Cuisine {  
    private String sauce;  
    protected void setSauce(String sauce) {  
        this.sauce = sauce;  
    }  
    public void Display() {  
        super.display();  
        System.out.println("Sauce: " + sauce);  
    }  
}
```

BBQ:

```
package lab07task04;  
  
public class BBQ extends Cuisine {  
    private String sauce;  
    protected void setSauce(String sauce) {  
        this.sauce = sauce;  
    }  
    public void Display() {  
        super.display();  
        System.out.println("Sauce: " + sauce);  
    }  
}
```

Beverages:

```
package lab07task04;

public class Beverages extends Cuisine{
    private String brand;
    public void setBrand(String brand) {
        this.brand = brand;
    }
    public void Display() {
        super.display();
        System.out.println("Brand: " + brand);
    }
}
```

Fast Food:

```
package lab07task04;

public class FastFood extends Cuisine {
    private String category;
    protected void setCategory(String category) {
        this.category = category;
    }
    public void Display() {
        super.display();
        System.out.println("Category: " + category);
    }
}
```

Output:

```
--- exec-maven-plugin:3.1.0:exec (default-cli) @ Lab07task04 ---
PAKISTANI:
Name: Biryani
Quantity: 2
Price: 700.0
Style: null
Meat Type: Chicken
Name: Mutton Karahi
Quantity: 4
Price: 1500.0
Style: null
Spice Level: Medium
BBQ:
Name: Beef Kabab
Quantity: 6
Price: 800.0
Style: null
Sauce: Mint Chutney
CHINESE:
Name: Sweet and Sour Chicken
Quantity: 4
Price: 1200.0
Style: Cantonese
Sauce: null
-----
BUILD SUCCESS
-----
```

Task No 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)

Code:

Main:

```
package lab07task05;

public class Lab07task05 {

    public static void main(String[] args) {

        System.out.println("Fruits and vegetable shop:\n");
        System.out.println("-----Weighed Item-----");
        WeightedItems w1 = new WeightedItems("Mangoes",200,3.5);
        System.out.println("Price of "+w1.getname()+" is "+w1.getPrice());
        System.out.println(w1.toString());
        System.out.println("-----Counted Item-----");
        CountedItems c1 = new CountedItems("Banana",64,6);
        System.out.println("Price of "+c1.getname()+" is "+c1.getPrice());
        System.out.println(c1.toString());
    }
}
```

Items:

```
package lab07task05;

public class Items {
    private String name;
    private double unitprice;
    public Items(String name, double unitprice) {
        this.name = name;
        this.unitprice = unitprice;
    }
    public Items() {
        this("no items", 0.0);
    }
    protected double getprice() {
        return unitprice;
    }
    protected String getname() {
        return name;
    }
    protected void setname(String name) {
        this.name = name;
    }
    protected void setunitprice(double price) {
        this.unitprice = price;
    }
    protected double getunitprice() {
        return unitprice;
    }
    public String toString() {
        return name + " @ Unitprice:" + unitprice;
    }
}
```

Weighted Items:

```
package lab07task05;

public class WeightedItems extends Items {
    private double weight;
    public WeightedItems(String name, double unitprice, double weight) {
        super(name, unitprice);
        this.weight = weight;
    }
    public double getprice() {
        return super.getprice() * weight;
    }
    public String toString() {
        return super.toString() + "(Weight " + weight + " kg)";
    }
}
```

Counted Items:

```
package lab07task05;

public class CountedItems extends Items{
    private int quantity;
    public CountedItems(String name, double unitprice, int quantity) {
```

```
        super(name, unitprice);
        this.quantity = quantity;
    }
    public double getPrice() {
        return super.getPrice() * quantity;
    }
    public String toString() {
        return super.toString() + " (Quantity: " + quantity + ")";
    }
}
```

Output:

```
--- exec-maven-plugin:3.1.0:exec (default-cli) @ Lab07task05 ---
Fruits and vegetable shop:

-----Weighed Item-----
Price of Mangoes is 700.0
Mangoes @ Unitprice:200.0(Weight 3.5 kg)
-----Counted Item-----
Price of Banana is 384.0
Banana @ Unitprice:64.0 (Quantity: 6)
-----
BUILD SUCCESS
-----
```