

Answer on Question #59370, Programming & Computer Science, Java, JSP, JSF

Condition

2-We must calculate the banana price by supplying both how many KGs of bananas we are adding to the cart, and the price per KG in the constructor. The `getCost()` method will return the total cost based on these two values.

3- Lemon: This class also inherits from Fruit. You must supply a price per lemon and how many lemons we are adding through the constructor. The `getCost()` method will just return the cost of all the lemons.

Code

Task.java

```
public class Task {  
    public static void main(String[] args) {  
        Apple apple = new Apple(3);  
        System.out.println("Apple: " + apple.getCost());  
  
        Banana banana = new Banana(3, 12.5);  
        System.out.println("Banana: " + banana.getCost());  
  
        Lemon lemon = new Lemon(6, 1.5);  
        System.out.println("Lemon: " + lemon.getCost());  
    }  
}
```

Fruit.java

```
public abstract class Fruit {  
    public abstract double getCost();  
}
```

Consts.java

```
public class Consts {  
    public static final double applePrice = 24.0; // per dozen  
}
```

Apple.java

```

public class Apple extends Fruit {
    private int applesNumber;

    public Apple(int number) {
        this.applesNumber = number;
    }

    @Override
    public double getCost() {
        return (Consts.applePrice / 12.0) * applesNumber;
    }
}

```

Banana.java

```

public class Banana extends Fruit {
    private double bananasWeight;
    private double bananaPrice;

    public Banana(double weight, double price) {
        this.bananasWeight = weight;
        this.bananaPrice = price;
    }

    @Override
    public double getCost() {
        return bananasWeight * bananaPrice;
    }
}

```

Lemon.java

```

public class Lemon extends Fruit {
    private int lemonsNumber;
    private double lemonPrice;

    public Lemon(int number, double price) {
        this.lemonsNumber = number;
        this.lemonPrice = price;
    }

    @Override
    public double getCost() {
        return lemonPrice * lemonsNumber;
    }
}

```

Output

Apple: 6.0

Banana: 37.5

Lemon: 9.0