

LABMID

NAME FAHAD FARMAN

REGNO:FA20-BSE-044

DECORATOR:

CODE

```
public interface MealComponent {  
    string getDescription();  
    double price();  
  
    public double getPrice();  
  
    public static class string {  
  
        public string() {  
            }  
        }  
    }  
abstract class MainDish extends MealDecorator {  
    public MainDish(MealComponent meal) {  
        super(meal);  
    }  
  
    public String getDescription() {  
        return super.getDescription() + ", Main Dish";  
    }  
}
```

```
    public double getPrice() {  
        return super.getPrice() + 5.0; // Additional cost for a main dish  
    }  
}
```

```
}
```

```
class Dish implements MealComponent {
```

```
    private String name;
```

```
    private double price;
```

```
    public Dish(String name, double price) {
```

```
        this.name = name;
```

```
        this.price = price;
```

```
    }
```

```
    public String getDescription() {
```

```
        return name;
```

```
    }
```

```
    public double getPrice() {
```

```
        return price;
```

```
    }
```

```

@Override

public double price() {

    throw new UnsupportedOperationException("Not supported yet."); // Generated from
nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody

}

}

class Burger extends MealDecorator {

    public Burger(MealComponent meal) {

        super(meal);

    }

    public String getDescription() {

        return super.getDescription() + " , Burger";

    }

    public double getPrice() {

        return super.getPrice() + 3.0; // Additional cost for a dessert

    }

}

abstract class Starter extends MealDecorator {

    public Starter(MealComponent meal) {

        super(meal);

    }

```

```
Starter(MealComponent meal) {

    throw new UnsupportedOperationException("Not supported yet."); // Generated from
nbfs://nbhost/SystemFileSystem/Templates/Classes/Code/GeneratedMethodBody

}

public String getDescription() {

    return super.getDescription() + ", Starter";

}

public double getPrice() {

    return super.getPrice() + 2.0; // Additional cost for a starter

}

}

public class MessManagement {

    public static void main(String[] args) {

        MealComponent meal = new Dish("Pasta", 10.0);

        meal = new Starter(meal);

        meal = new MainDish(meal);

        meal = new Burger(meal);
```

```
System.out.println("Meal Description: " + meal.getDescription());
```

```
System.out.println("Total Price: $" + meal.getPrice());
```

```
}
```

```
}
```
