

Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class Order {
7     private static int orderNo = 0;
8     private double totalPrice;
9     private final List<FoodItem> orderItems = new ArrayList<>();
10
11     public Order() {
12         orderNo++;
13     }
14
15     public int addItem(FoodItem item) {
16         this.totalPrice += item.getPrice();
17         orderItems.add(item);
18         return orderItems.size();
19     }
20
21     public double getTotalPrice() {
22         return this.totalPrice;
23     }
24
25     public String getOrderDetails() {
26         String itemDivider = "-----\n";
27         String orderDivider = "-----\n";
28         StringBuilder details = new StringBuilder();
29         details.append(orderDivider);
30         details.append("Order No: #").append(orderNo).append("\n");
31         details.append(orderDivider);
32         for (FoodItem item : this.orderItems) {
33             details.append(item.getDescription());
34             details.append(itemDivider);
35         }
36         details.append(Formatter.format("Total", this.getTotalPrice()));
37         details.append(orderDivider);
38         return details.toString();
39     }
40 }
41
```

Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800;
2
3 import edu.cpp.cs5800.addons.CheeseDecorator;
4 import edu.cpp.cs5800.addons.LettuceDecorator;
5 import edu.cpp.cs5800.addons.PicklesDecorator;
6 import edu.cpp.cs5800.addons.TomatoDecorator;
7 import edu.cpp.cs5800.customer.Customer;
8 import edu.cpp.cs5800.customer.GoldCustomer;
9 import edu.cpp.cs5800.customer.PlatinumCustomer;
10 import edu.cpp.cs5800.customer.SilverCustomer;
11
12 public class Driver {
13     public static void main(String[] args) {
14         Order o1 = new Order();
15         Customer c1 = new PlatinumCustomer();
16         o1.addItem(new TomatoDecorator(new CheeseDecorator(new ChickenBurger())));
17         o1.addItem(new PicklesDecorator(new VeggieBurger()));
18         System.out.print(o1.getOrderDetails());
19         System.out.println(c1.applyLoyaltyDiscount(o1.getTotalPrice()));
20
21         Order o2 = new Order();
22         Customer c2 = new GoldCustomer();
23         o2.addItem(new LettuceDecorator(new CheeseDecorator(new TomatoDecorator(new
FishBurger()))));
24         System.out.print(o2.getOrderDetails());
25         System.out.println(c2.applyLoyaltyDiscount(o2.getTotalPrice()));
26
27         Order o3 = new Order();
28         Customer c3 = new SilverCustomer();
29         o3.addItem(new ChickenBurger());
30         System.out.print(o3.getOrderDetails());
31         System.out.println(c3.applyLoyaltyDiscount(o3.getTotalPrice()));
32     }
33 }
34
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800;  
2  
3 public interface FoodItem {  
4     double getPrice();  
5     String getDescription();  
6 }  
7
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800;
2
3 public class Formatter {
4     public static String format(String name, double price) {
5         int totalWidth = 25;
6         String strPrice = String.format("%.2f", price);
7
8         int dashCount = totalWidth - name.length() - strPrice.length();
9         String fill = new String(new char[dashCount]).replace("\0", " ");
10
11         return name + fill + strPrice + "\n";
12     }
13 }
14
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800;  
2  
3 public class FishBurger implements FoodItem {  
4  
5     private static final double price = 6.0;  
6  
7     @Override  
8     public double getPrice() {  
9         return price;  
10    }  
11  
12    @Override  
13    public String getDescription() {  
14        return Formatter.format("Fish Burger", price);  
15    }  
16 }  
17
```

Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800;  
2  
3 public class VeggieBurger implements FoodItem {  
4  
5     private static final double price = 4.5;  
6  
7     @Override  
8     public double getPrice() {  
9         return price;  
10    }  
11  
12    @Override  
13    public String getDescription() {  
14        return Formatter.format("Veggie Burger", price);  
15    }  
16 }  
17
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800;  
2  
3 public class ChickenBurger implements FoodItem {  
4  
5     private static final double price = 5.5;  
6  
7     @Override  
8     public double getPrice() {  
9         return price;  
10    }  
11  
12    @Override  
13    public String getDescription() {  
14        return Formatter.format("Chicken Burger", price);  
15    }  
16 }  
17
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800.addons;
2
3 import edu.cpp.cs5800.FoodItem;
4
5 public abstract class AddOnDecorator implements FoodItem {
6     protected FoodItem foodItem;
7
8     public AddOnDecorator(FoodItem foodItem) {
9         this.foodItem = foodItem;
10    }
11 }
12
```


Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800.addons;
2
3 import edu.cpp.cs5800.FoodItem;
4 import edu.cpp.cs5800.Formatter;
5
6 public class CheeseDecorator extends AddOnDecorator {
7
8     private static final double price = 1.5;
9
10    public CheeseDecorator(FoodItem foodItem) {
11        super(foodItem);
12    }
13
14    @Override
15    public double getPrice() {
16        return this.foodItem.getPrice() + price;
17    }
18
19    @Override
20    public String getDescription() {
21        return this.foodItem.getDescription() + Formatter.format("+ Cheese", price);
22    }
23 }
24
```

Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800.addons;
2
3 import edu.cpp.cs5800.FoodItem;
4 import edu.cpp.cs5800.Formatter;
5
6 public class TomatoDecorator extends AddOnDecorator {
7
8     private static final double price = 1.0;
9
10    public TomatoDecorator(FoodItem foodItem) {
11        super(foodItem);
12    }
13
14    @Override
15    public double getPrice() {
16        return this.foodItem.getPrice() + price;
17    }
18
19    @Override
20    public String getDescription() {
21        return this.foodItem.getDescription() + Formatter.format("+ Tomato", price);
22    }
23 }
24
```

Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800.addons;
2
3 import edu.cpp.cs5800.FoodItem;
4 import edu.cpp.cs5800.Formatter;
5
6 public class LettuceDecorator extends AddOnDecorator {
7
8     private static final double price = 0.5;
9
10    public LettuceDecorator(FoodItem foodItem) {
11        super(foodItem);
12    }
13
14    @Override
15    public double getPrice() {
16        return this.foodItem.getPrice() + price;
17    }
18
19    @Override
20    public String getDescription() {
21        return this.foodItem.getDescription() + Formatter.format("+ Lettuce", price);
22    }
23 }
24
```

Decorator (Assignment 5 Implementation)

```
1 package edu.cpp.cs5800.addons;
2
3 import edu.cpp.cs5800.FoodItem;
4 import edu.cpp.cs5800.Formatter;
5
6 public class PicklesDecorator extends AddOnDecorator {
7
8     private static final double price = 1.2;
9
10    public PicklesDecorator(FoodItem foodItem) {
11        super(foodItem);
12    }
13
14    @Override
15    public double getPrice() {
16        return this.foodItem.getPrice() + price;
17    }
18
19    @Override
20    public String getDescription() {
21        return this.foodItem.getDescription() + Formatter.format("+ Pickles", price);
22    }
23 }
24
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800.customer;
2
3 public interface Loyalty {
4     int getDiscountPercentage();
5     String applyLoyaltyDiscount(double amount);
6 }
7
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800.customer;
2
3 import edu.cpp.cs5800.Formatter;
4
5 public class Customer implements Loyalty {
6
7     @Override
8     public int getDiscountPercentage() {
9         return 0;
10    }
11
12    @Override
13    public String applyLoyaltyDiscount(double amount) {
14        return Formatter.format("After " + this.getDiscountPercentage() + "% Discount:", ((
double) (100 - this.getDiscountPercentage()) / 100 ) * amount);
15    }
16 }
17
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800.customer;  
2  
3 public class GoldCustomer extends Customer {  
4     @Override  
5     public int getDiscountPercentage() {  
6         return 15;  
7     }  
8 }  
9
```

Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800.customer;
2
3 public class SilverCustomer extends Customer {
4     @Override
5     public int getDiscountPercentage() {
6         return 10;
7     }
8 }
9
```


Decorator (Assignment 5 Implmentation)

```
1 package edu.cpp.cs5800.customer;
2
3 public class PlatinumCustomer extends Customer {
4     @Override
5     public int getDiscountPercentage() {
6         return 20;
7     }
8 }
9
```