

# CS 2263

## Lab 07 – Refactor It!

Assigned: February 7, 2019

Due: February 7, 2019

For each of the code snippets below, there is a new feature that needs to be added. But the code will be ungainly if not refactored before adding that feature. So, for each snippet,

Identify which refactoring will help accomodate the **new** feature;

Show the code resulting from your refactoring in a. performed on it.

Explain why it will now be easier to add the **new** feature (don't add it, just explain why).

Empress Cruise Lines allows their customers to specify many accommodation preferences, and they would like to allow even more preferences, including queen vs double bed, and smoking or non-smoking room. The following method declaration appears on the Booking interface.

```
public void requestBooking(
    Customer cust, Date start, Date end,
    MealType meal, boolean isPreferredCustomer,
    boolean isViewRequired);
```

Shop-even-more.com provides a client shopping cart library for interacting with its servers. Many years ago the protocol was upgraded and they used the code structure below to keep support for code using the legacy handler. They are soon going to release a major version upgrade (with breaking changes allowed to interfaces) and want to drop support for this legacy protocol.

```
public interface CartHandler {
    public void addToCart(String item);
    public void login(String user, String pass);
    public void checkout();
}
public class LegacyCartHandler implements CartHandler { ... }
public class NewCartHandler implements CartHandler { ... }
```

Jack Hammer University wants to now allow non-student affiliates (such as employees) to take classes at JHU. Currently, this would require copying a lot of logic from the Student class to the Employee class.

```
public class Person { ... }
public class Employee extends Person { ... }
```

```

public class Student extends Person {
    private Schedule schedule;
    ...
    public void enroll(Course course) {
        if (!schedule.isFree(course.getTimeSlot())) {
            throw new ScheduleException();
        } else if (schedule.getCourseCount() >= schedule.getMaxCourses()) {
            throw new FullException();
        } else {
            schedule.getAllCourses().add(course);
        }
    }
}

```

The eBay.com online auction site permits three types of accounts: buyer accounts, seller accounts, and administrator accounts. It is often the case that a customer is both a buyer and a seller, but this currently requires two different accounts. eBay would instead like to think of users as fitting into certain roles and would like to expand the number of roles in the future.

```

public class Account { ... }
public class SellerAccount extends Account { ... }
public class BuyerAccount extends Account { ... }
public class AdministratorAccount extends Account { ... }

```

The Cutthroat Fashion computer game consists of characters competing for certain looks. Player characters can acquire accessories, but adding new accessories to the game is becoming difficult due to how it was originally coded and subsequently patched on to.

```

public int computePunkness(Accessory accessory) {
    if (accessory.getName().equals("shawl")) {
        return 3;
    } else if (accessory.getName().equals("lampshade hat")) {
        return 2;
    } else if (accessory.getName().equals("bum flap") &&
        accessory.getColors().equals(
            Arrays.asList("black", "grey"))) {
        return 32;
    } else if (accessory.getName().equals("army boots")) {
        return 20;
    }
    return 0;
}

```