Computer Science 3620
Assignment 1 - Due Friday February 10 @11:55pm

For each of these problems you must provide a working IntelliJ project with the solution and a UML diagram describing the class design that you implemented.

1) Strategy Pattern

You are developing a new “Rock Legends” game and need a configuration system for your players. The game has three different rock legend characters (The Edge, Jimi Page, and David Gilmour). Each character can play one of three different guitars (Gibson SG, Fender Stratocaster or Gibson Les Paul). Each character can also perform a unique solo act (Smash the Guitar, Set the Guitar on Fire, Jump off the Stage).

Implement a player configuration system in Java using the Strategy design pattern. You should have separate classes for each of the components specified.   Below is a ‘driver’ class (with main) to get you started - it is not exhaustive with respect to your assignment: you should add more code showing how the Strategy Pattern can be used in this problem.

```
public class RockLegends {
   public static void main(String[] args) {
      GameCharacter player1 = new GameCharacterEdge();
      GameCharacter player2 = new GameCharacterPage();
      player1.playGuitar();
      player2.playGuitar();
      player1.playSolo();
      player2.playSolo();
   }
}
```

(see submission guidelines below)

2) Decorator Pattern

A car dealer sells three vehicle types: a sedan, a sports car and an SUV. Each of these can be purchased with add-on options. Options include a premium sound system, alloy wheels, a moon roof, and navigation system. Each car has a base price (sedan = 20,000, sports car = 30,000, SUV = 40,000) and each option has a fixed price (sound = 1000, wheels = 2000, moon roof = 1500, navigation = 1800). Write a program that uses the Decorator Pattern to create any vehicle type with any mix of options. Write a test class to build the following vehicles, displaying the vehicle type and final price.

- bare sedan (no options)
- sedan with moon roof
- sports car with sound, wheels and navigation
- SUV with wheels and moon roof
- SUV loaded

Submission Guidelines:

Submit your assignment as a single zip file through the moodle link provided. Use the following naming format for your zip file:

lastname-as1.zip

The zip file should contain two folders, “problem1” and “problem2”, corresponding to the two problems above. Each of these folders should contain an IntelliJ project with the solution. **The folder must also include a UML class diagram describing the design solution**.