**COEN 359 Design patterns Summer2015**

**Assign1 (100 pts)** **Due: 7th July (10 am)**

*PartyOrders* is a small company that specializes in organizing parties and celebrations for small to medium sized gatherings. A **party setting** consists of several table settings. Each **table setting** consists of

* A table with 1 center-piece, 4 seats (chairs) and 4 place-settings.
* Each place-setting consists of plates, silver and napkins.

Currently, seats, place settings and center-pieces come in three styles: **vintage**, **contemporary** and **modern**.

**Pricing is determined as follows:**

Base Price for a chair: $50

Base Price for a place-setting: $20

Base Price for a center-piece: $30

Price for a Vintage chair: base price of chair + $50

Price for a Vintage place-setting: base price of a place-setting + $40

Price for a Vintage center-piece: base price of a center-piece + $35

Price for a Contemporary chair: base price of chair + $20

Price for a Contemporary place-setting: base price of a place-setting + $20

Price for a Contemporary center-piece: base price of a center-piece + $25

Price for a Modern chair: base price of chair + $100

Price for a Modern pace-setting: base price of a place-setting + $30

Price for a Modern center-piece: base price of a center-piece + $40

A client (who uses the services of the *PartyOrders*) supplies the following information:

* No. of tables required.
* Style of chairs (vintage, contemporary or modern), place-settings and center-piece.

**Note:** The client is required to choose the same style for chairs, place-settings and center-piece. For example, if the client chooses vintage for seating, then the place-settings and center-piece should also be vintage in style.

You are required to build an Object-Oriented model of a party setting for *PartyOrders* and implement your design in Java without using any design patterns.

**Note**: Use the basic concepts of class, object, inheritance, composition etc to build your model and implement it. Do not use any OO Design Patterns in your design or implementation.

1. Draw an UML diagram for the given description.
2. Implement the classes and relationships using Java. In the constructor of each class, for example, Place-Setting, include a print statement (System.out.println)
3. Define a class called **Party-Setting** (include this class in the UML diagram) with the following members:

**Data Members:**

**noOfTables**: an integer

**tables**: An array of Table-Setting (where each table-setting contains a Center-piece, place setting and a seat) in the chosen style.

**Methods:**

A method called **calculateCost() which calculates and returns the total cost for the party-setting (total cost of all the table-settings).**

A method called **displayPartySetting()** which displays the number of tables and what each table contains (place settings etc and the style).

Note: You are free to include any other data or methods of your choice.

From the main(), create an instance of Party\_Setting with data values of your choice (for the no. of tables, style etc). Call the methods, **displayPartySetting (**) and **calculateCost().**

**What to submit:**

1. **A zip file (your *firstinitialLastName*\_assign1.zip) with the following**

* A readme document with the UML diagram, the JDK version and IDE used. Include your name and course name on the document. **(30 pts)**
* Source code (include comments wherever necessary) **(50 pts)**
* A screen shot of the output of your program **(10 pts)**

1. **A reflections document (10 pts)**: helps to assess your progress as an individual learning the material through the assignments and it helps the instructor to formulate plans for the future. In your reflections document, address the following issues as they are applicable for the current assignment.

* What did you like about this assignment?
* What did you find difficult?
* Have you learned anything in this assignment that you did not know before?
* Have you used and found the references given useful?
* Any other comments that are relevant