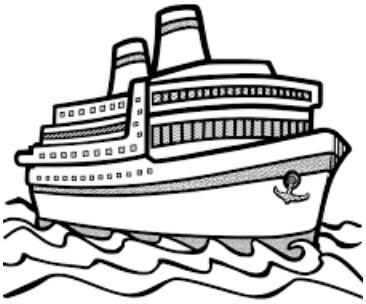

Luxury Vacation Cruise

System Specification Document

Project Overview

Our mission at Luxury Vacation Cruise is to provide the accurate and efficient ability to coordinate cruise bookings for our growing list of affiliate luxury cruise ship owners and cruise lines. We require a simple menu-based system to perform core functions that are essential to our daily operations.

The envisioned system will include three primary components: ships, cruises, and passengers. The Luxury Vacation Cruise company does not own ships; instead, it creates cruises using available ships. Once a ship is placed "in service," it is available for cruises. Passengers can be added to cruises. Here are the relationships of these components:

- Ships have unique names.
- Cruises have unique names and are assigned to ships.
- Ships can have multiple cruises.
- Each ship has a finite number of passenger cabins (Balcony, Ocean View, Suite, and Interior). Overbooking is not permitted.
- Passengers are assigned to cruises.

Your Assignment

Your task is to design the described system mentioned above. The company only requires a simple text-based menu system (naturally GUI is preferred) that should resemble the illustration below. Use all the knowledge you have learned in this course to design the system. You have learned the OOP design methodology and attained all the required skills and concepts of Java language features to write this program. Use your programming abilities and determine how you intend to design this software.

Your program is expected to be well documented. You are also required to submit the UML Diagram and generate your project JavaDoc (please watch the video lecture on how to generate JavaDoc).

Luxury Vacation Cruise
System Menu

[1] Add Ship	[A] Print Ship Names
[2] Edit Ship	[B] Print Ship In Service List
[3] Add Cruise	[C] Print Ship Full List
[4] Edit Cruise	[D] Print Cruise List
[5] Add Passenger	[E] Print Cruise Details
[6] Edit Passenger	[F] Print Passenger List
[x] Exit System	

Write a Learning Report Summary

Using Microsoft Word or any text editors, answer the following eight questions.

1. Did you complete your assignment and did it run without errors?
2. Did your program produce the correct result?
3. Did you test your program thoroughly?
4. How much time did you spend to complete your assignment?
5. Did you write the program yourself? Did you get any help from anyone?
6. When you encountered obstacles to complete your program, how did you resolve the issues? Did you use Google to get help? Describe how Google was abled or not able to assist you?
7. What did you learn from doing this assignment?
8. Any other information you would like to share with your instructor?

Submission Instructions:

- Submit all source code files.
- Submit the UML Diagram.
- Submit your JavaDoc project (zip all files).
- Submit your Learning Report Summary.
- Submit a 2 to 4 minutes video recording that demonstrates and proves of your program running without any error, and produces the correct results. Select at least five different options to test-run your program. Please make sure that you appear in your presentation for at least 3 to 5 seconds so that I know you are the person doing the presentation. You can use any available videorecording means to record your presentation. You can also use the Zoom virtual tool to record your demonstration.

Rubric (250pts)

Program logic and designed with Java class standard modules; Constructors, Accesor, Mutator, toString(), additional modules (30pts):

Designed classes using protected/private attributes (5pts):

Program implements menu driven and options (20pts):

Program documentation included (10pts):

UML diagram (10pts):

JavaDoc Feature Documentation (15pts):

Learning report summary submitted (10pts):

Program compile, run, and generate expected results (75pts):

Recorded video presentation of your program running (75pts):