

## Factory Pattern Assignment

### General Overview:

Factory Inc. is tasked with creating a streamlined system to keep track of orders being received by clients who wish to have a vehicle constructed. The program task is to be able to properly emulate this factory's laundry list of equipment in order to organize what different pieces the factory needs in order to construct in order to meet demands. For our example, the factory in question is only focused on.

### Deliverable:

Zipped eclipse project.

### Specifications:

Realistically, this system would be really automatic, with users who order vehicles automatically generating a report for the factory to be expected to produce. To keep things simple in this assignment, this process will be simulated by user input. The program will have to keep track of the different components expected by each vehicle ordered. The factory is in charge of creating three different types of vehicles: Airplane, Car, and Boat. The only components of the different vehicles we are concerned with are the amount of engines and wheels each needs (Assume that the same engine and wheel are used for each vehicle). The bottom table clarifies how many components the factory expects from each vehicle.

Type of Vehicles	Wheel Amount	Engine Amount
Airplane	12	8
Boat	0	4
Car	4	1

Using the factory design pattern, create a class system that produces these separate vehicle objects and, upon request, generates a report in the console. (No export needed) The main class which the code will be run on will be called *"InventoryTesting"*. This class will be responsible for retrieving user input and generating objects as a result of the input and produce a report. This report provides information on how many of each type of vehicle needs to be made as well as how many wheels and engines need to be in stock to satisfy the requests. Your code must call upon the object that is created for these values (wheel count, engine count) in order to do the math that will lead to the final report.

### NOTICE:

Code that attempts to skip these steps and generate the report through other means that does not involve the factory object creation and retrieval (getters/setters) will be receiving harsh grade penalties. As long as you produce the factory design within your class system and appropriately call the properties

from their respective objects, you will not be penalized. However you decide to take in the user input and format the final report is up to you (Keep it logical). Again, the assignment is not importing any input text file or asking for any exported text file. The program should be self-contained besides the user input that generates the objects.