

15.05 Assignment Instructions

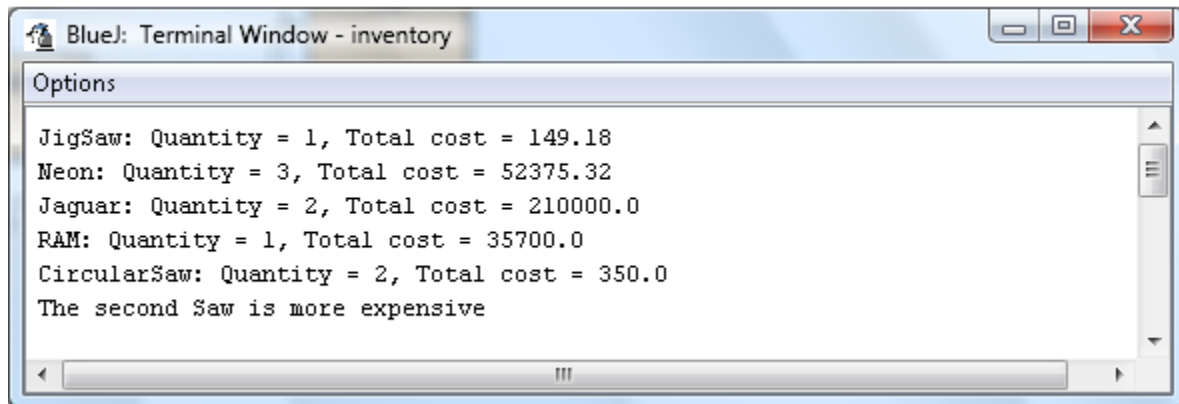
1. Create a folder called **15.05 Assignment** in your module 15 assignments folder.
2. Create an interface named **Product** .
 - a. Add a method called **getName()** that returns a string.
 - b. Add a method called **getCost()** that returns a double.
3. Create abstract class **Vehicle** that implements **Product**.
 - a. It should have string variable **name** and double **cost**, that are initialized in the constructor.
 - b. Add appropriate **getName()** and **getCost()** methods
4. Create classes **Car** and **Truck** that extend **Vehicle**.
 - a. No other methods are needed.
5. Create class **Tool** that implements **Product** and **Comparable<T>** .
 - a. It should have string variable **name** and double **cost** that are initialized in the constructor.
 - b. Add appropriate **getName()** and **getCost()** methods.
 - c. Add a **compareTo()** method that compares tools based upon **cost** .
6. Create class **InventoryDemo**.
 - a. Test your classes by using ArrayList **products** of following products (Remember to declare it properly using List):

Name	Cost
Jaguar	1000000.00
Neon	17000.00
JigSaw	149.18
Jaguar	110000.00
Neon	17500.00
Neon	17875.32
RAM	35700.00
CircularSaw	200.00
CircularSaw	150.00

- b. Create a static method **takeInventory** that, when passed the **name** of a product, will go through the list and print out <item name>: Quantity = <quantity>, Total cost = <totalcost>. <item name> is the **name** of the product, <quantity> and <totalcost> are the values you calculate by going through the list for the product with name that was passed to **takeInventory**.

- c. To test the **compareTo()** method, create two Tools, **saw1** , and **saw2** . Give them different prices and then test the **compareTo()** method you made, by displaying which one is more expensive.

Your output should be similar to:



The screenshot shows a BlueJ Terminal Window titled "BlueJ: Terminal Window - inventory". The window contains the following text:

```
Options
JigSaw: Quantity = 1, Total cost = 149.18
Neon: Quantity = 3, Total cost = 52375.32
Jaguar: Quantity = 2, Total cost = 210000.0
RAM: Quantity = 1, Total cost = 35700.0
CircularSaw: Quantity = 2, Total cost = 350.0
The second Saw is more expensive
```

The window has a standard Windows-style title bar with minimize, maximize, and close buttons. The text is displayed in a monospaced font, and there is a scrollbar on the right side of the text area.