

CSE 102L Computer Programming Laboratory – Homework 6

Due Date: 23:59 Sunday May 9th

Disclosure

You will submit your file to an assignment that is given through MS teams. Your filename should be *HW06_yourStudentNumber.java*. Submissions made after the deadline will not be accepted, be sure to submit your work before the due date and make sure to click turn in button. Your code will be automatically controlled, so be sure to have the same class, method, variable names as described here. Failure to do so may result in you receiving 0 from this exercise. **All classes should be written to a single Java file. In a single java file there can only be single public class. Do NOT use Inner Classes. Be careful naming your file. If your editor inserts the file into a package, remove that line from the file but do NOT delete the import statements.**

Write set of classes according to the following specifications. Declare all attributes as **private** if not requested otherwise and use camelCase formatting for attributes.

Interfaces

1. Sellable
 - Methods:
 - getName(): String
 - getPrice(): double
2. Package <T>
 - Methods:
 - extract(): T – removes the item T (assigns null) and returns item **only** if it is not empty. Returns null if it is empty.
 - pack(T item): boolean – puts an item T in if it is empty
 - isEmpty(): boolean – checks if item T is null or not
3. Wrappable – child interface of Sellable. This interface is used to restrict which items can be put inside a Matroschka instance.

Classes

1. Product – an abstract class that implements Sellable
 - Attributes:
 - name: String
 - price: double
 - Methods:
 - Constructor that takes *name*, and *price*
 - Accessors
 - toString(): String – returns a String in the format of:
 - simple name of *this* class + “ (“ + name + “ , “ + price + “)”

2. Mirror – child of Product

- Attributes:
 - width: int
 - height: int
- Methods
 - Constructor that takes *width*, and *height*. Sends “*mirror*” and 2 to its parent’s constructor.
 - `getArea(): int` – returns the area of mirror
 - Override `getPrice` to scale with area
 - `reflect(T item): T` – displays the reflected item and returns it.

3. Paper – child of Product implements Wrappable

- Attributes:
 - note: String
- No-arg constructor, passes “Paper” and 0.5
- Accessor and Mutator for note

4. Matroschka <T extends Wrappable> - child of Product and implements Wrappable and Package with type T

- Attributes:
 - Item: T
- Methods
 - Constructor that takes *item* of type T packs it, and sends its parent’s constructor “Doll” and 5 + the price of item (item can be null, be careful)
 - `toString(): String` – returns a string in format:
 - `super’s toString + “{” + item + “}”`

5. Box <T extends Sellable> – implements Package with type T

- Attributes:
 - item: T
 - seal: boolean
- Methods:
 - No-Arg constructor that sets *item* to null
 - Constructor that takes *item* of type T packs it, sets seal to true
 - `extract(): T` – in addition to what it does (See related interface) sets *seal* to false.
 - `isSealBroken(): boolean`
 - `toString(): String` – returns a String in the format:
 - simple name of *this* class + “{” + item + “} Seal: “ + seal

