COSC 3P91

Lab 1

In this lab you are going to implement an existing design of parts of a game in Java. The design is provided by a UML class diagram below. The following additional comments might help during implementation:

1. The method `computeHitPoints()` in the enum type `WeaponClass` will return
   a. D6+6 for a bow,
   b. 2D6+4 for a staff,
   c. D6+8 for a sword,
   where D6 is a roll of a six-sided dice (2D6 = roll of two six-sided dices). Such a roll can be computed by `(int)(6*Math.random()+1)`.
2. The `Wizard` is a magic user, the `Fighter` and the `Ranger` are not.
3. The method `inventoryWeight()` in the class `Inventory` returns the sum of the weights of all items in the inventory.
4. The method `reduceArmorValue(amount : int)` in the `Equipment` class reduces the armor value of the equipment by amount. The armor value cannot become negative. i.e., if the armor value is 2 and amount is 3, then the new armor value will be 0 (and **not** -1).
5. The `toString()` method in the class `Item` will first call the method `getDetails()` in order to print the details of the specific item and then add the general information stored in `Item` to that string.
6. The `toString()` method in the `Hero` class is supposed to print all available information about the hero, i.e., name, class, and the inventory. Below you find two examples as a guideline:
   ```
   This is the hero Draxen Bloodfist.
   He is a Fighter.
   His inventory contains the following items:
   A Sword with range 1.
   Price: 100
   Weight: 10
   A Shield with armor value of 4.
   Price: 50
   Weight: 8
   ```

```
This is the hero Morgana Moonstone.
She is a Wizard.
Her inventory contains the following items:
A Staff with range 5.
Price: 30
Weight: 12
A Healing Potion with armor value of 0.
Price: 88
Weight: 1
```

7. Delegation/Forwarding is indicated by the fact that the corresponding methods in the sender and receiver class have the same name.

Hero

- name : String {readonly}
- male : boolean {readonly}
- cClass : CharacterClass {readonly}
- inventory : Inventory {readonly}

+ Hero(name : String, male : Boolean,
        cClass : CharacterClass)
+ addItem(item : Item)
+ dropItem(item : Item)
+ toString() : String

1 —— **1**

Inventory

- items : List<Item> {readonly}

+ Inventory()
+ addItem(item : Item)
+ dropItem(item : Item)
+ numItems() : int
+ inventoryWeight() : int
+ toString() : String

1
0..*

<<enumeration>>
CharacterClass

Fighter
Ranger
Wizard
+ magicUser : Boolean {readonly}

Item

- weight : int {readonly}
- price : int

+ Item(weight : int, price : int)
+ getWeight() : int
+ getPrice() : int
+ setNewPrice(newPrice : int)
+ toString() : String
 # getDetails() : String

Weapon

- wClass : WeaponClass {readonly}
- range : int {readonly}

+ Weapon(weight : int, price : int,
      wClass : WeaponClass, range : int)
+ computeHitPoints() : int
# getDetails() : String

Equipment

- description : String {readonly}
- armor : int

+ Equipment(weight : int, price : int,
        description : String, armor : int)
+ reduceArmorValue(amount : int)
# getDetails() : String

<<enumeration>>
WeaponClass

Bow
Staff
Sword

+ computeHitPoints() : int