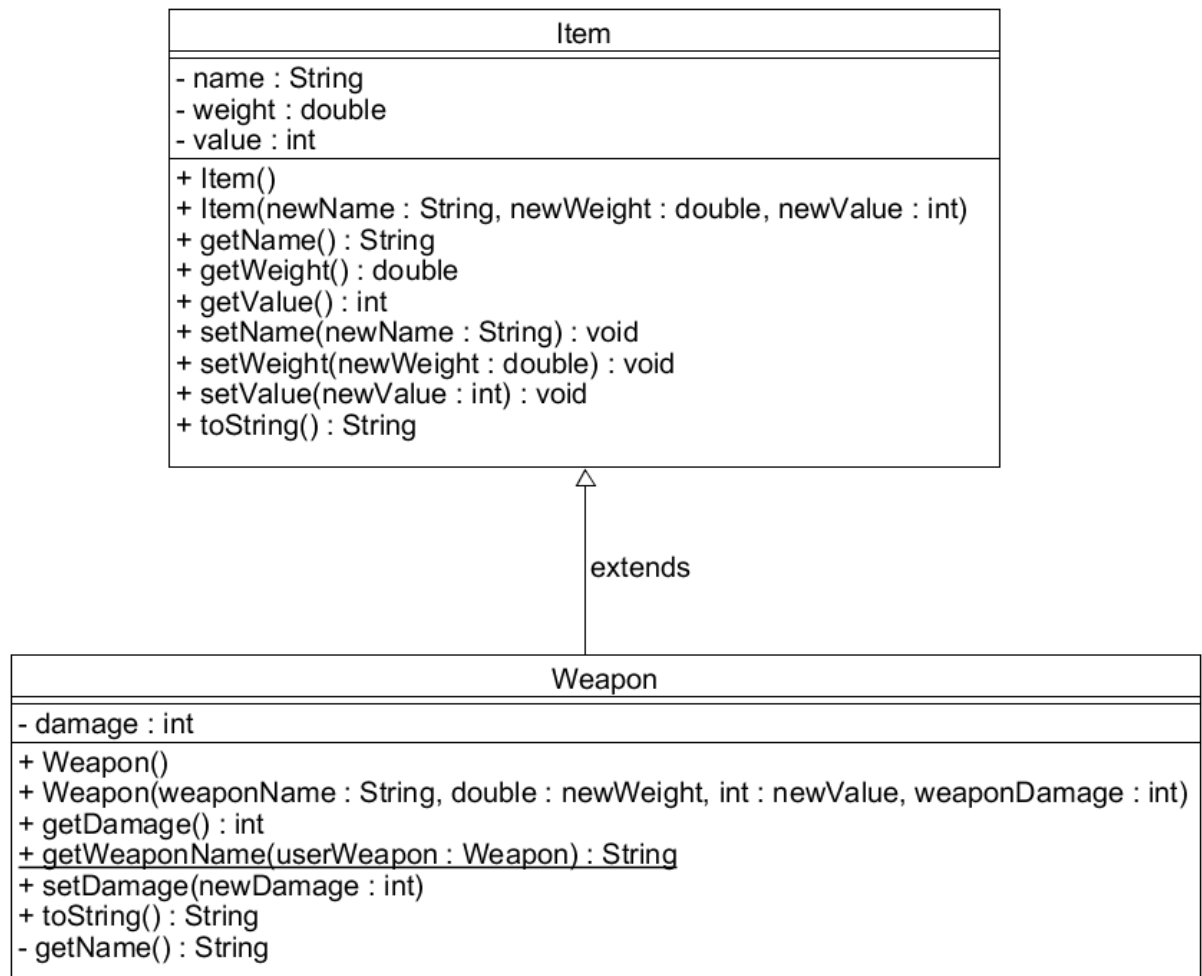


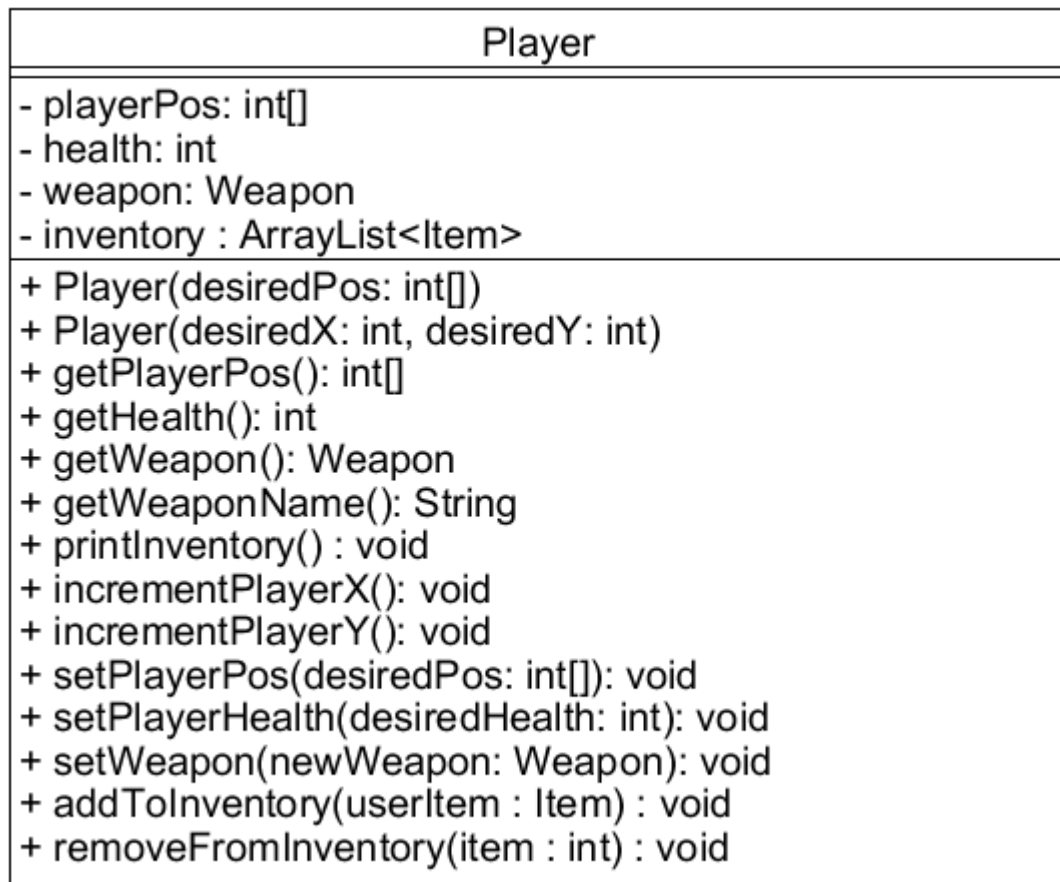
1. Given a superclass "Animal" and a subclass "Walrus", write the CODE for the class definition of "Walrus". It should contain NO data or methods; just show the class definition such that "Walrus" inherits from "Animal".

```
public class Walrus extends Animal {  
  
}
```

2. Given the code below, what is this an example of? **Animal a = new Walrus();**
  - a. Upcasting
3. Is the code in the previous question legal? Briefly explain your answer.
  - a. It is allowed because a walrus is a type of animal.
4. Design and draw the UML diagrams for the following classes: Also draw the inheritance relationship!



5. Modify the Player class and draw the new UML diagram for it (you do not need to draw any relationships here):



6. A child class inherits private methods from the superclass.  
a. True – but they are not accessible
7. In Java, a subclass may only extend ONE superclass.  
a. False
8. A protected field in the superclass is accessible by a subclass, EVEN if the subclass is in a DIFFERENT package than the superclass.  
a. False
9. What class is the ancestor of ALL other Java classes? (I.e., it is at the very top of the inheritance tree?)  
a. Java.lang
10. What Java operator allows me to check what class a given object is?  
a. instanceof
11. When is the no-args **super()** constructor **implicitly** called?  
a. Only if the current constructor does not explicitly make a call to another constructor or superclass constructor

12. To **override** a method, what must be the same?
- a. The signature