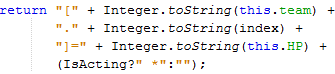
Advanced Programming Using Java - **acp3 discussion** - Lesson 6 (Chapter 3 Part B)

GOAL: learn about when NOT to use abstract class

**Do not actually do these instructions.** It would be a bad idea to make an abstract class so that each game object (“Entity”) is interchangeable, because it introduces a standard that is incompatible with the world unit list:

* File, New File, Java Class, Next
* Set Class Name to Entity, Package acp, Finish
* Change “public” to “abstract”
* Write a new function that should be implemented in every entity, which may include either creatures or inanimate objects (an abstract function ends in a semicolon instead of curly braces, since the subclass is responsible for implementation):  
  
* Make the Entity a subclass of the abstract class: add “extends Entity” after the name of the Unit class in the Unit.java file
* Click the bulb by that line, then click “Implement all abstract methods”
* Find the new method that was added, erase the throw statement, then add the following code (or paste from MainForm then change to match this):  
  
* Now make the MainForm use the new method so that each subclass of Entity can have its own method of display:
  + Open MainForm
  + Find where the old version of the code above was used (Ctrl F and type “Integer.toString(i” without quotes)
  + Delete the String msg variable, i==unit\_index, and the first parameter of setValueAt, and change the call to look like the code below:   
    