

Static and non-static Members

Due Date:

This assignment is due Friday, January 18 *at the beginning of lecture.*

Persons

- ⇒ Implement a `Person` class that has all of the methods shown in Figure 1.
- ⇒ Implement a `PersonTester` class whose public static `void main()` method uses the tests the `Person` class and simultaneously tells a story.

The `allSayHello` static member function should cause every currently living person to say hello. The tricky part of this is finding all of the currently living persons. To accomplish this use static member variable(s) to keep track of living people, and ensure that these variable(s) are updated by appropriate non-static functions. One technique is to use a member variable like

```
private static ArrayList<Person> thePeople;
```

Another is to use a class like

```
class PersonLink { private Person myPerson ; private PersonLink next ; ...
```

and a static member variable of type `PersonLink`. Choose whatever you are comfortable with.

All fields (member variables) **must be private**.

The public methods of the `Person` class must be exactly those described in Figure 1. You may add as many private methods and fields as you see fit.

Here are some things to check.

- Make sure that `die()` applied to a dead person doesn't cause the population to decrease.
- Make sure that a person's murderer's name prints correctly if the murderer herself is dead.
- Make sure that murdering a person causes them to die.

<i>Method</i>	<i>Meaning</i>
<i>attributes</i>	
<code>public Person(String n)</code>	Creates a living person with name n.
<code>public Person murderer()</code>	Returns the Person that murdered this person. Returns null if this person has not been murdered.
<code>public String name()</code>	Returns this person's name. Should end with ", deceased" if the person is dead.
<code>public boolean isAlive()</code>	<i>obvious.</i>
<i>actions</i>	
<code>public void die()</code>	Causes a person to die. Has no effect on someone already dead.
<code>public void murder(Person victim)</code>	Causes victim to die, and the murderer to be known to the victim.
<code>public void sayHello()</code>	causes this Person to print "Hello, I'm name." on System.out.
<i>class attributes</i>	
<code>public static int numberLiving()</code>	
<code>public static int numberDead()</code>	
<i>class actions</i>	
<code>public static void allSayHello()</code>	causes every living person to say hello as described above.

Figure 1: Properties and Actions of Persons