

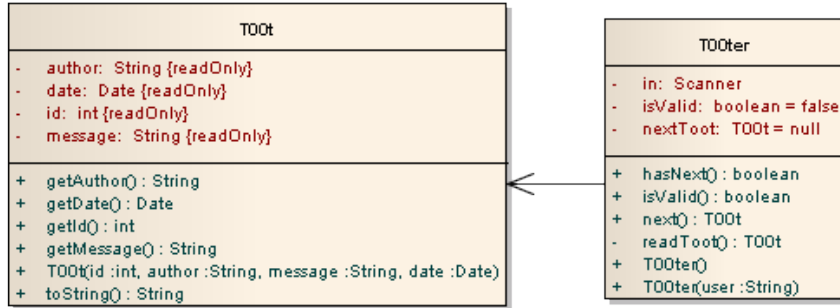
Show all of your work clearly in the space provided or on the additional page at the end of the exam. Be sure to **read each problem carefully**. Note that the exam is double sided. You are not required to provide comments in your code or include `import` statements.

1. (10 points) Describe when to use an abstract class. Your answer should be concise and complete.

2. (10 points) What would be displayed if the following code snippet were executed? Note the ASCII value of 'A' is 65.

```
char c = 'A';
int i = 3;
double x = 2;
System.out.println(i+x);
System.out.println(++i);
System.out.println(i);
System.out.println(i++);
System.out.println(i);
System.out.println(i+(int)x);
System.out.println(c);
System.out.println(c+i);
System.out.println((char)(c+i));
```

3. The classes shown in the UML diagram were required to complete the lab 1 assignment.



Provide example Java code that does each of the following:

(a) (10 points) Display the contents of the most recent toot on the www.t00ter.com website.

(b) (10 points) Display the username for the second most recent toot posted on the website.

(c) (10 points) Assume that user “caleb” has fewer than 8 toots total. Display the number of toots this user has made.

4.

(a) (10 points) Create an interface called **Physical** that provides the following methods that describe a physical object's mass and volume: `getMass()` and `getVolume()`. No arguments should be passed to the methods and they should each return a numeric value. Provide the complete contents for the `Physical.java` file.

**(b)** (15 points) Create a class called **Stone** that implements the **Physical** interface. In addition, the class should provide a **getColor()** method along with one constructor that allows the user to initialize the mass, volume, and color of the object when it is created. If you have any doubt about what a method should do, please ask. Provide the complete contents for the **Stone.java** file.

(c) (15 points) Create a class called **Gem** that is a subclass of **Stone**. The class should have one additional method: `getPrice()` which returns a random price between \$45.00 and \$3000.00. Note that the `Math.random()` method returns a random value between 0.0 and 1.0. Provide the complete contents for the `Gem.java` file.

**(d)** (10 points) Draw the UML class diagram for the **Physical**, **Stone**, and **Gem** interface/classes. Include all appropriate connectors and show all methods that are available to objects of a class even if the method is not actually implemented in the class, i.e., include inherited methods.