

1. Complete and correct this code for the Coin class. There are 8 errors (not including the blank constructor)

```
/** (Javadoc comments start with /**)
 * a coin A coin with a value. (a complete sentence)
 * @author Bill Gates (must have tag name after @)
 */
public class Coin {
    private static double value; // Each coin has its own value.
                                // So value can't be static!
    public Coin(double value) {
        this.value = value;
    }

    /**
     * Get the value of this coin.
     * @return value of this coin
     */
    public int double getValue(int value) { //NO PARAM on accessor method
        return value;
    }

    public String toString() {
        return System.out.print( value+"-Baht coin" );
    }
    // toString() should not print anything.
}
```

2. Complete this code to sum an array of coins and return the sum. Use a "for-each" loop if you know how.

```
public static double sum( Coin[] coins ) {
    // This method has to be completely correct for credit.
    // This is prerequisite skill that you learned in Programming 1.
    double sum = 0.0;
    for( Coin c: coins ) sum += c.getValue();
    return sum;
}
```

3. This code sometimes returns the correct sum and sometimes not. What is wrong?

```
int sum = 0;
public int sum( int[ ] nums ) {
    for(int k=0; k<nums.length; k++) sum = sum + nums[k];
    return sum;
}
```

Describe why the error occurs and show how to fix it. Write your correction on the code.

In this code, sum is an attribute so its only initialized to zero once!

It should be a local variable so it is initialized to zero each time the method is invoked.

4. Suppose you try to run Java in a Terminal or Command window and you get this output:

```
cmd> java Main
Command not found: java
```

What does it mean?

- a) The java command isn't installed on your computer.
- b) The directory containing "java" or "java.exe" isn't on your Command or Terminal's search path.
- c) You typed it wrong. Should be "Java Main.class". *This is not correct.*
- d) Either a or b.
- e) Either a, b, or c.

5. Name these characteristics of object-oriented programming and of objects.

Encapsulation	An object contains both attributes (data) and the methods that operate on the attributes. The class may choose to make the data private to prevent direct access.
Polymorphism	We can use an object reference to invoke a method, even though the method may be performed by different kinds of objects. We don't need to know which class will actually perform the method. An example is <code>x.toString()</code> . This invokes the <code>toString()</code> of whatever kind of object that <code>x</code> refers to.
Inheritance	One class can extend another class (the parent class) so that it automatically has all the methods of the parent class. This lets us reuse code.
State	(A characteristic of objects.) The condition of an object or what the object knows (the values of its attributes).
Class	A blueprint or definition for a kind of object.

6. What happens when we assign one object reference (variable) to another one?

Use this code:

```
String[] s = {"a", "b", "c" }; // array of 3 strings s[0]="a", s[2]="c"
String[] t = {"", "", "", ""}; // array of 4 empty strings.

// What does this do?
t = s;
System.out.println("Length of t is " + t.length); // 5.1

t[0] = "confusing";
System.out.println( s[0] ); // 5.2
```

6.1 What does the first println statement print? **Length of t is 3** _____

6.2 What does the second println statement print? **confusing** _____

6.3 Explain what "t = s;" does.

It makes t refer to whatever object that s refers to.