

If your only tool is a hammer, everything looks like a nail.

Pseudo Object-Oriented Programming Considered Harmful

Conrad Weisert
Information Disciplines, Inc., Chicago

A recent piece in ACM *Communications*¹ urged us to do more to indoctrinate our students into object-oriented programming. The author proposed to rewrite the popular "Hello, World" first complete program example "to include a user-created object". He believes this Java version to be object-oriented:

```
class HelloWorld {
    public static void printHello()
    {System.out.println
      ("Hello, world");
    }

    public class UseHello {
        public static void
            main(String args[])
        {HelloWorld myHello
          = new HelloWorld();
          myHello.printHello();
        }
    }
}
```

Experienced OOP practitioners know that that's no more object-oriented than the original pure procedural version.

First, we note that the class `HelloWorld` contains neither member data nor instance methods. There's no point, then, in ever instantiating an object of that class. We conclude that `HelloWorld` is a *pseudo-class* whose only role is to serve as a packaging artifice for a single function.

Sometimes a student turns in work containing such a pseudo class. I ask him or her: "What is the purpose of this class? What does an object of this class represent?" Of course, the student has no answer, having missed the basic concepts of classes and objects.

Second, even if we accept the `HelloWorld` class the invocation of its `static` function doesn't

call for an object. A Java programmer just invokes it like this:

```
public class UseHello {
    public static void
        main(String args[])
    {HelloWorld.printHello();
    }
}
```

So the empty pseudo-object `myHello` served no purpose at all and only added overhead for unnecessary dynamic memory allocation.

Third, the name of the other class, `UseHello`, reveals its lack of any cohesive purpose. Even a beginning programmer ought to form strong doubts about a paradigm or a language that requires one clumsy construct just to get at another clumsy construct.

A student who turned in such an assignment, even after just one class session, would earn, at most, a grade of C, along with a few stern comments from this instructor.

If we look hard we may find object-oriented opportunities in the "Hello, World" program. A `Message` class might encapsulate not only the text, but also the severity-level, the destination, and perhaps the language of a message object. A `UserConsole` class might provide the flexibility to display output in a GUI or applet window rather than just the command-line interface. Either or both would yield a more flexible and more widely usable program.

OOP is for modeling *objects*, not for obfuscating simple logic.

CONRAD WEISERT (cweisert@acm.org) is a consultant specializing in software development methodologies. He currently teaches object-oriented programming at Illinois Institute of Technology.

¹ "Hello World Considered Harmful", Technical Opinion, October, 2001, p. 129.