

1. What design principles does this code violate?
2. Refactor the code to improve its design.

### Person.java

```
public class Person {
    public String last;
    public String first;
    public String middle;

    public Person(String last, String first, String middle) {
        this.last = last;
        this.first = first;
        this.middle = middle;
    }
}
```

### PersonClient.java

// The clients are in one file for convenience;  
// imagine them as non-test methods in separate client classes.

```
import junit.framework.TestCase;
```

```
import java.io.*;
```

```
public class PersonClient extends TestCase {
    public PersonClient(String name) {super(name);}
}
```

```
    public void client1(Writer out, Person person) throws IOException {
        out.write(person.first);
        out.write(" ");
        if (person.middle != null) {
            out.write(person.middle);
            out.write(" ");
        }
        out.write(person.last);
    }
}
```

```
    public String client2(Person person) {
        String result = person.last + ", " + person.first;
        if (person.middle != null)
            result += " " + person.middle;
        return result;
    }
}
```

```
    public void client3(Writer out, Person person) throws IOException {
        out.write(person.last);
        out.write(", ");
        out.write(person.first);
        if (person.middle != null) {
            out.write(" ");
            out.write(person.middle);
        }
    }
}
```

```

public String client4(Person person) {
    return person.last + ", " +
        person.first +
            ((person.middle == null) ? "" : " " + person.middle);
}

public void testClients() throws IOException {
    Person bobSmith = new Person("Smith", "Bob", null);
    Person jennyJJones = new Person("Jones", "Jenny", "J");
    StringWriter out = new StringWriter();
    client1(out, bobSmith);
    assertEquals("Bob Smith", out.toString());

    out = new StringWriter();
    client1(out, jennyJJones);
    assertEquals("Jenny J Jones", out.toString());

    assertEquals("Smith, Bob", client2(bobSmith));
    assertEquals("Jones, Jenny J", client2(jennyJJones));

    out = new StringWriter();
    client3(out, bobSmith);
    assertEquals("Smith, Bob", out.toString());

    out = new StringWriter();
    client3(out, jennyJJones);
    assertEquals("Jones, Jenny J", out.toString());

    assertEquals("Smith, Bob", client4(bobSmith));
    assertEquals("Jones, Jenny J", client4(jennyJJones));
}
}

```