File system example

```java
import java.io.*;

class fs {

public static void main (String[] arguments)
    throws java.io.IOException  {
    PrintWriter filehandler = new PrintWriter(new FileWriter("hello.txt"));
    filehandler.println(arguments[0]);
    filehandler.close();
    }
}
```

```java
class fs {

  public static void main (String[] arguments)  {
    try {
      PrintWriter filehandler = new PrintWriter(new FileWriter("hello.txt"));
      filehandler.println(arguments[0]);
      filehandler.close();
    }
    catch (IOException e)  {
      System.out.println("Ooop, promblem with the filesystem:" + e);
    }

  }
}
```

Java's implicit constructor invocation

Quoting from Section 8.8.5 of the Java Language Specification:

blah blah ... *If a constructor body does not begin with an explicit constructor invocation and the constructor being declared is not part of the primordial class Object, then the constructor body is implicitly assumed by the compiler to begin with a superclass constructor invocation "super();", an invocation of the constructor of its direct superclass that takes no arguments.*

test-like questions inheritance

- What is the difference between superclass & subclass?

- Which keyword is used to inherit a class?

- When can subclasses not access superclass members?

- Which class does begin Java class hierarchy?

- Object class is a superclass of all other classes? (True/False)

test-like questions inheritance

- Java supports multiple inheritance? (True/False)

- Which method is used to call the constructors of the superclass from the subclass?

- Which is used to execute any method of the superclass from the subclass?

- Does a class inherit the constructors of it's super class?