

Files and Folders

A dedicated operating system controlling a mystery system has code which generates and handles folders and files¹. Some of the code related to this system will be introduced in this sample practice question.

Every File has 10 Lines of text. Here is the File class:

```
public class File {
    private Line[] textLines;
    private static int numFiles = 0;

    public File() {
        textLines = new Line[10];
        generateLines();
        numFiles = numFiles + 1;
    }

    private void generateLines() {
        //to be implemented
    }

    public static int getNumFiles() {
        return numFiles;
    }
}
```

¹ As well as being able to manage input and output devices and manage memory, operating systems are also expert at file management.

Line objects store a single line of text:

```
public class Line {  
    private String text;  
  
    public Line(String text) {  
        this.text = text;  
    }  
  
    public String getText() {  
        return this.text;  
    }  
}
```

1. Complete the `generateLines` method of the `File` class. This method assigns a `Line` object to every element in the array `textLines`. For simplicity, every `Line` in the array will store the text **“Hello”**

Write/Paste your solution here

[3]

2. Design an accessor method, `getLines`. This will return a reference to the `textLines` array.

[2]

A Folder HAS Files. Here is the Folder class:

```
public class Folder {
    private File[] files;

    public Folder() {
        files = new File[5];
        generateFiles();
    }
    private void generateFiles() {
        //to be implemented

    }

    public int countHello() {
        //to be implemented
    }
}
```

3. The modifier used in the signature of the `generateFiles` method is *private*. Explain the consequence of using this modifier in the signature of the method.

[2]

4. The `generateFiles` method will populate the `files` array with 5 files. Implement this method.

[3]

Write/Paste your solution here

5. The `countHello` method will visit each `File` object in the array `files` and determine a count of the number of instances of the word "Hello" in all of the `File` objects. It will return the count.

Implement the `countHello` method.

[7]

Write/Paste your solution here

6. A `HelloFolder` IS-A `Folder`. However, as well as having an array of `File` objects, it also has a boolean property which is `true` if the hello count is greater than 20 and `false` otherwise. When a `HelloFolder` is first constructed, it generates the array of `File` objects as ordinary `Folders` do. It then assigns a value to the boolean indicator depending on the total number of “Hello” instances in the `Files`.

Design the full `HelloFolder` class, including any new properties and accessor/mutator methods for any new properties.

[5]

Write/Paste your solution here

Cont↓

7. Draw UML diagrams describing the relationship between all classes used in this project. Fully describe the Line class and only include the title of each class in the other UML diagrams, as well as the relationships between classes.

[5]

Draw/Paste your solution here

8. In a main() program, write lines of code to instantiate a HelloFolder object and display messages which show the status of the object's boolean indicator.

Eg:

[3]

"This HelloFolder does not have enough instances of Hello!"

Write/Paste your solution here