

Disclaimer: the following are provided for your reference only, and the inclusion of information here does not guarantee it will be used on the exam.

Methods from the `java.io.File` class:

<code>File(String filename)</code>	Creates a <code>File</code> object for the given file name
<code>boolean exists()</code>	Returns <code>true</code> if the specified file already exists
<code>boolean canRead()</code>	Returns <code>true</code> if the specified file can be opened for reading
<code>boolean canWrite()</code>	Returns <code>true</code> if the specified file can modified (written to)

Methods from the `java.util.Scanner` class:

<code>Scanner(String s)</code>	Creates a <code>Scanner</code> to read the <code>String s</code>
<code>Scanner(System.in)</code>	Creates a <code>Scanner</code> that reads from the keyboard.
<code>Scanner(File fn)</code> throws <code>FileNotFoundException</code>	Creates a <code>Scanner</code> to read from file
<code>void close()</code> throws <code>IOException</code>	Closes the stream and any associated file
<code>boolean hasNext()</code>	Returns <code>true</code> if there's another token of input.
<code>boolean hasNextInt()</code>	Returns <code>true</code> if the next input is an int value.
<code>boolean hasNextDouble()</code>	Returns <code>true</code> if the next input is a double value.
<code>boolean hasNextLine()</code>	Returns <code>true</code> if there's another line of input.
<code>String next()</code>	Returns the next word only, as a <code>String</code> .
<code>int nextInt()</code>	Returns the next word only, as an integer.
<code>double nextDouble()</code>	Returns the next word only, as a double.
<code>String nextLine()</code>	Returns the next line as a <code>String</code> .

Questions

1. Write a method named `fileEcho` that accepts a filename as a `String` and displays each line to the screen with a line number.

2. Write code that reads a grade file and computes the percentage for each student. The first line is the grade item name. The second line is the percentage of the final grade for each grade item. The third line is the max score possible for each grade item. The remaining lines are scores for specific students.

Example grade file:

```
ITEM,e1,e2,p1,p2,LABS
WEIGHT,10,15,10,10,15
MAX,100,100,50,50,15
s1,89,93,49,50,15
s2,70,52,47,48,15
s3,88,79,42,38,15
s4,76,80,38,45,15
s5,75,74,40,43,15
```