# Desk Check: AsciiArt.java

The source code for the AsciiArt.java program is shown below. Scan the program looking for features you recognize to get the big picture, especially notice the use of String reference variables.

```
< 1>  /**
< 2>   * This program uses String objects to print an ASCII art message.
< 3>   *
< 4>   * ©FLVS 2007
< 5>   * @author B. Jordan
< 6>   * @version 03/05/07
< 7>   */
< 8>  public class AsciiArt
< 9>  {
<10>     public static void main(String [] args)
<11>     {
<12>        //local variables
<13>        String row1 = "  *   ***** ***** *****   ***** ***** ***** *  * *****  *";
<14>        String row2 = " * *  *    * *     *       *   * *   * *     * *  *      *";
<15>        String row3 = "***** ***** *     *****   ***** *   * *     **   *****  *";
<16>        String row4 = "*   * *     *         *   * *   *   * *     * *      *   ";
<17>        String row5 = "*   * *     ***** *****   *   *   ***** ***** *  * *****  *";
<18>
<19>        //print ASCII art
<20>        System.out.println(row1);
<21>        System.out.println(row2);
<22>        System.out.println(row3);
<23>        System.out.println(row4);
<24>        System.out.println(row5);
<25>     }
<26>  }
```

Once you have a clear overview of the program, start dissecting the code line-by-line.

Line(s)
==========

<1> through <7> are comments which are ignored by the computer.
<8> declares AsciiArt to be the name of the class.
<9> an opening curly brace marking the beginning of the class (matches up with Line <26>.
<10> the header for the **`main()`** method which is where program execution begins.
<11> the opening curly brace marking the beginning of the **`main()`** method (matches up with Line<25>).
<12> a comment indicating the variable declaration & initialization section.
<13> through <17> declare and initialize five string variables.
<18> white space to improve readability.
<19> a comment indicating the section of the program that will print output.
<20> through <24> print the String literals assigned to the row variables.
<25> a closing curly brace marking the end of the **`main()`** method (matches up with Line <11>.
<26> a closing curly brace marking the end of the class (matches up with Line <9>).

Make sure you understand each line of code before moving on to the assessment.