

READ ALL INSTRUCTIONS BEFORE STARTING YOUR SOLUTION.

Added Topics: analysis, structured, top-down **design**, expressions, variables, loops, class constants

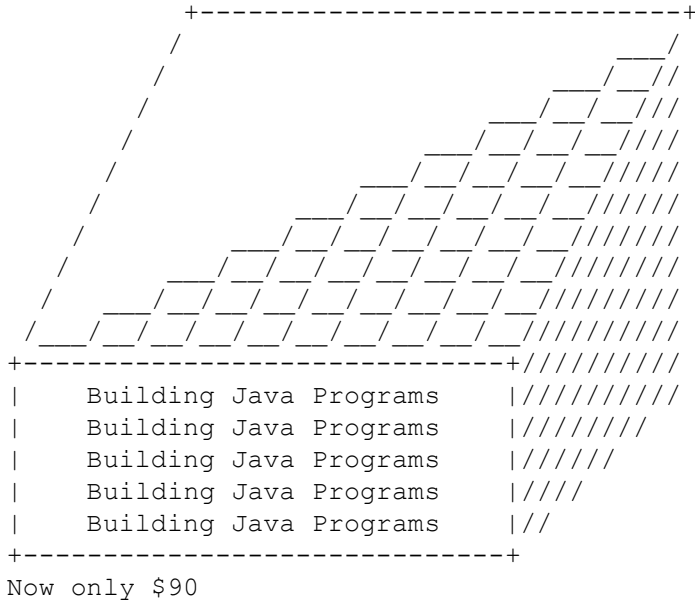
Specifications:

1. You must follow all the programming style requirements (available on the web site) to earn full credit on all programs. That document also describes the scoring process used for your programs and some design hints.
2. Write a Java class to produce a text figure that looks similar to the first edition of our textbook, *Building Java Programs*, sitting on its side. Your output must include the title line with **your name**, not Ada's

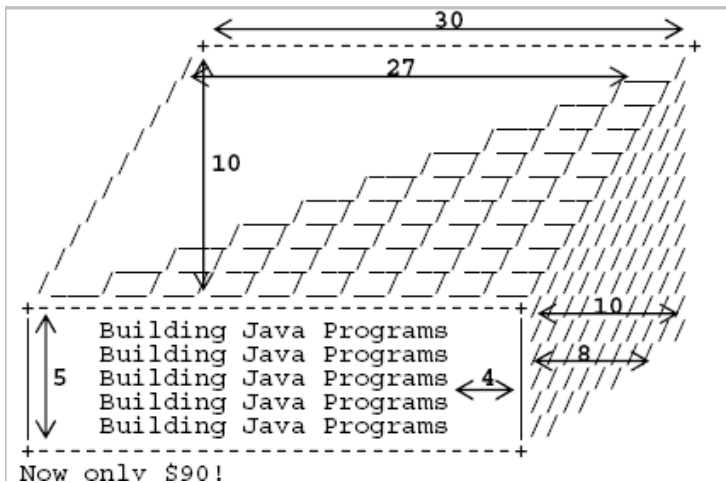
```
... >java BookEdHyde
```

That's what I could type to run EdHyde's program.

Book Program by Ada Byron, Countess of Lovelace



3. You must create a Class constant for the figure's height/size: to scale the size of the various pieces of the drawing. **Initially, use 10 as the value of this constant.** Later we will change its value to enlarge or shrink the drawing. The height and width of the drawing are proportional; that is the height determines the width. All other class constants (such as a width constant) must be based on the height scale constant.



4. For a height scale size = 10. Your code must exactly reproduce the output above. Your output must have identical characters and spacing.

5. One might try to draw this figure with a `System.out.println` statement for each line of the figure. That is not acceptable; this assignment should demonstrate that you understand analysis, design and loops. Use loops to produce lines that have repeated patterns of characters that vary in number from line to line, represent the lines and character patterns with nested for loops as done in the Chapter 2 case study. Examine the image, analyze the repetitions and make tables to understand the patterns in the output; then write pseudo-code for your design.
6. On any one execution your program will produce just one version of the figure. You must refer to the class constant(s) throughout your code – no "magic numbers" can appear in your code. If I change only your height scaling constant's value and recompile your program it must produce a proportional figure of a different size. Your program must work correctly for any even constant value of 8 or greater--I will not test your code with an odd value or a value less than 8.
7. This assignment is best completed in stages. At first, do not worry about the constant. Design an initial version of the program without a constant. Focus on each section of the figure individually. Use **loop tables** and **pseudo code**, to help you deduce the patterns in the output. After your figure looks correct at the default size, make a backup copy of your solution in a separate folder. See Chapter 2's case study example using a constant to draw a figure. You will change the constant and recompile each time you want to see a size change.

Note that my current SCC email server rejects any message with an attachment that contains executable code even if the executable code is zipped. Please email typos and suggestions for this document to me.