|  | Initials |
| --- | --- |
| Exercise 1.1 Create another circle. Then create a square. |  |
| Exercise 1.2 What happens if you call **moveDown** twice? Or three times? What happens if you call **makeInvisible** twice? |  |
| Exercise 1.3 Try invoking the **moveVertical, slowMoveVertical,** and **changeSize** methods before you read on. Find out how you can use **moveHorizontal** to move the circle 70 pixels to the left. |  |
| Exercise 1.4 Invoke the **changeColor** method on one of your circle objects and enter the string **"red".** This should change the color of the circle. Try other colors. |  |
| Exercise 1.5 This is a very simple example, and not many colors are supported. See what happens when you specify a color that is not known. |  |
| Exercise 1.6 Invoke the **changeColor** method, and write the color into the parameter field *without* the quotes. What happens? |  |
| Exercise 1.7 Create several circle objects on the object bench. You can do so by select- ing **new Circle()** from the pop-up menu of the **Circle** class. Make them visible, then move them around on the screen using the “move” methods. Make one big and yellow; make another one small and green. Try the other shapes too: create a few triangles, squares, and persons. Change their positions, sizes, and colors. |  |
| Exercise 1.8 Make sure you have several objects on the object bench, and then inspect each of them in turn. Try changing the state of an object (for example, by calling the **moveLeft** method) while the object inspector is open. You should see the values in the object inspector change. |  |
| Exercise 1.9 Figure 1.8 shows two different images. Choose one of these images and rec- reate it using the shapes from the *figures* project. While you are doing this, write down what you have to do to achieve this. Could it be done in different ways? |  |
| Exercise 1.10 Select *Show Terminal* from the *View* menu. This shows another window that BlueJ uses for text output. Then select *Record method calls* from the terminal’s *Options* menu. This function will cause all our method calls (in their textual form) to be written to the terminal. Now create a few objects, call some of their methods, and observe the output in the terminal window. |  |