Layout Reading Questions

Use Eck's Textbook, Section 6.7, for this reading. You can skip sections 6.7.2, 6.7.3, 6.7.4, and 6.7.6.

1. Containers (circle all that apply)
   a) Include the content pane of JFrame   c) Might be JPanels
   b) Are components   d) Come with default layouts

2. Layout considers s______ and p__________________.

3. Each container can specify a different layout manager by using the method set____________________().

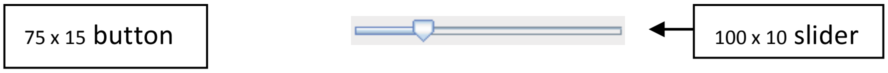
4. Layout managers are in what class that you would import? java. a_______

5. Next to each container, write the default layout for it.
   a) JPanel's default layout is _______________________________________.
   b) JFrame's default layout is _______________________________________.

6. We used setContentPane() to add a Panel to a Frame. We will continue to do that. For other components, they are placed on a container by using the method add(). If the container (a JPanel) is called holdIt and the component is called button, write the one line of code that would place the button on the container.

7. To set the size of a container, you can use the method ____().
   Otherwise, the layout manager may set the size of the container.

For the following problems, consider the components shown below. They are shown in their default preferred sizes (but sizes may not be fixed for all layouts). Redraw the components in the layouts being described for each of the following questions, in the 200 x 100 JPanels (boxes) shown below. If an object is repeated in a list of components, assume the indicated number of instances can exist in your JPanel.



8. Code is setLayout(new FlowLayout(FlowLayout.CENTER, 50, 20)). The following are then added to the JPanel in order: button, slider, button. Show what the JPanel would finally look like. Draw the slider and button proportionally.

9. Code is setLayout(new BorderLayout(10, 20). The following are then added to the JPanel with the associated positions indicated as shown: button (BorderLayout.SOUTH), button(BorderLayout.WEST), slider(BorderLayout.CENTER). Show what the JPanel would finally look like.
   *Note that these are added in the same order the layout will be normally established – North/South gets handled before East/West, then Center last (taking up remaining space). Use the box on the next page.

10. Code is setLayout(new GridLayout(2, 4, 10, 5). The following are then added to the JPanel with the in order: slider, button, slider, slider, button, button, button, slider. Show what the JPanel would finally look like. Note that the components will resize to fill each of their spaces completely.

11. Now consider Null Layout, which we will tend to avoid (the reading explains why). To the left or below of each line of pseudocode, add the line of code to do what the pseudocode describes.

// set container holdIt to have a null layout

// set button onto holdIt (don't worry about size and placement)

//make button be 100 x 200 pixels placed with its top left corner at (0,50) on holdIt. (Width is 100.)

12. The final layout required in class is Card Layout. Eck doesn't have a tutorial on it, so refer to Oracle's tutorial at http://docs.oracle.com/javase/tutorial/uiswing/layout/card.html (if you didn't know, Oracle owns the Java language now). In the example in this tutorial, a different method is used to establish the layout of cards, the container JPanel that holds two other panels (card1 and card2). Next to each line of pseudocode, write the code found in the tutorial. Part of two lines are written in a slightly modified way for you to make answering the last part of this question easier.

__________ = ______ ________________ (_____ ______________()); // establishes layout of cards JPanel

___________ . _________ (_______________ , "Card with JButtons"); // puts card1 JPanel onto cards

___________ . _________ (_______________ , "Card with JTextField"); // puts card2 JPanel onto cards

If card1 is meant to be shown, then fill in the line of code below to do this

cardLayoutInstance.show (____________, "_____________________________JButtons")

13. Write a list of four other methods you can use to display different card panels (other than show()). (* In case you were wondering, the first card is the first one added when the card stack is generated.)