Chapter 9 – Gui using AWT

Answers

1. A,B,E

   The List constructor used here accepts two parameters. The first one is an int value, representing the number of visible rows to display in the List. And the second parameter is a boolean value, indicating whether or not the list allows for multiple selections. Here a 10 is passed for the number of rows, and “true” is passed allowing for multiple selections.

2. B

   Since the checkbox has no font set for it, it will use the font set for the container that directly holds it, which is a panel. If the panel had no font set for it, then the checkbox would use the font of the applet, since the panel would have used that as well.

3. B

   A menu cannot contain a Checkbox item, but can contain a CheckboxMenuItem, which represents a checkbox that can be used in a menu.

4. A

   For each component, if certain attributes were set directly for the component, such as background color and font, those are the attributes that would be used for the component. If any attributes were not set for a component, then the component uses the attributes of the container component holding it, and so on up the hierarchy.

5. B,C,D

   In this example a TextArea is being created, and the constructor is being passed the number of rows and columns, 8 and 24, respectively. The number of rows defines the height of the TextArea, which is also the number of possible rows that can be displayed in the TextArea. There can be more that 8 lines of text in this TextArea, and if this occurs here a vertical scrollbar will appear.

The width of a column in a TextArea is defined by the width of a character of the current font. A width defined for a TextArea defines the number of columns that are visible to the user. So here the TextArea will have a width that is 24 characters wide. The TextArea can hold more characters than the width displayed in this example, and if this occurs a horizontal scrollbar would be visible.

6. B

The CheckboxGroup class actually extends directly from the Object class and not from the Component class.

7. B

A menu cannot contain a Checkbox item, but can contain a CheckboxMenuItem, which represents a checkbox that can be used in a menu.

8. B

Once a text field is assigned a String value, its width does not change after that, even if the amount of text gets longer. You then would need to use the right and left arrow keys to scroll right and left within the text field.

9. B

A MenuBar object can only be put onto a Frame object, and not onto a Panel object.

10. B

An Applet cannot contain a MenuBar object, since the Applet class extends from the Panel class, and MenuBar objects can only be placed on Frame objects.

11. B

A MenuBar object can only be put onto a Frame object, and not onto a Panel object.

12. B

The size and look of different components varies depending on the computer's platform.

13. B

The number of lines to appear in a container that has a GridLayout is determined either when the GridLayout constructor is called, or later can be set with specific methods. The number of items added to the container has no bearing on the lines.

14. A

When using the GridLayout Manager on a container, all of the components placed on this container will have the same width and height.

15. A

The most practical and popular place to put a horizontal scrollbar is at the bottom of the container that can be scrolled. With the BorderLayout, when putting a scrollbar in the South position, it will take up the whole width of the container that it is on, and that is the preferred way to use a scrollbar.

16. E

Each layout manager has its own setBounds() method. If a component calls the setBounds() method, then the layout manager's version of this method will be called afterwards, and therefore this second version will determine the display on the screen. So in this example the setBounds() method has no effect on the component. If the layout manager were to be set to be null, then the setBounds() method would work for components.

17. A

When adding a component to a container that has the BorderLayout defined for it, if no location is defined for the component, but default it will be placed in the Center position.

18. A, B,C,D

The Label component just displays text, and is not used to catch user input. The default alignment for the text in a label is left, and the text on a label can be changed as often as needed with the method “setText(String txt)”. A List component holds a column of text.

19. A

A column’s width in a text area is determined by the approximate average character’s width for a given font on a given platform.

20. A,B,C

All of these statements are true.

21. A

A Menu is a component that is placed onto a MenuBar, which is located on a Frame. A Menu can hold all of the 4 items listed here.

22. A,B,C

All of these are true. One of the constructors to create a Font is: new Font(String name, int style, int size)

23. C,D,E

The method “drawRect” accepts 4 int parameters. The first two, x and y, represent the upper left corner. And the second two represent the width and the height of the rectangle.

24. A

A polygon is always a closed shape. You pass to the constructor of a polygon 2 arrays, one of x points and one of y points, and you also pass it the number of points to use. In this case, the 2 arrays define 4 x and y points, and the constructor is passed a 4 as the number of points to use. If you sketch out these points on paper, you will see that it draws a polygon shape. With a polygon, a line is always drawn from the last point to the first point as well, to close it.

25. B

The parameters “0,0” passed to the drawstring method determine the point at the left baseline where the text is to begin. Since all of the letters in the word “ISRAEL” are written above the baseline, and here the baseline is the top left corner of the applet, the word will not be displayed on the screen.

26. A

When using the Grid Layout, one, but not both of the rows and cols values can be set to 0. If the columns are set to 0 then all of the items will be displayed in one row, and if the rows are set to 0 then all of the items will be displayed in one column.

27. A,C,D

This program can act either as an applet or as an application. If it is used as an applet the init() method will be called, and if it is used as an application, the main() method will be called.

28. A, B, C, D, E

When using the GridLayoutManager, if the number of rows specified is greater than 0, then that will be the number of rows displayed, regardless of the number of columns passed. This layout displays 3 rows and 4 columns. Since the answers A, B, C, D and E all define the rows to be 3, the value in the columns is irrelevant here. Since there are 12 items added, there will always be 4 columns in this case.

29. A

With the FlowLayout Manager components are added to the screen left to right as they appear in the code. In addition all of the components are centered on the screen. If one row is filled and more components are added, they will be added on the next row in the same manner.

30. A

With the BorderLayout Manager one component can be added to each of the 5 available positions, which are: Center, North, South, East, and West. This is what is happening in this example.

31. A

With the BorderLayout Manager only one component can be viewed in each of the 5 available positions, which are: Center, North, South, East, and West. In order to overcome this restriction, a panel can be added to any of these positions, and that panel can contain many components, which gives the effect of the BorderLayout position itself containing more than 1 component. In this example the East position has one panel on it, and that panel contains 3 components. And the West position has a panel on it that contains 2 components.

32. A

For a Frame the default layout manager is the BorderLayout. In this program, the “SomeGui2” class extends Frame, and no layout manager is set for it, so the BorderLayout is used by default. Then 3 panels are created, each holding their own Button components. Then one panel is placed on the West of the frame, another on the East of the frame, and the third on the Center of the frame, which creates this appearance. Items placed in the Center position in a Border Layout use up all of the space remaining after the other 4 positions have been filled (or not filled). In this case the North and South positions were not filled on the frame.

33. A

Only 2 text fields are displayed on the frame here, and not 3. This is because the first 2 text fields are added to the frame, and then the line “frame.setVisible(true)” appears, which causes the frame be displayed, The third text field is added to the frame after this line, so it is not displayed.

34. B

In this example 3 text fields are created and added to the frame, and then the line “frame.setVisible(true)” appears, so all of the text fields are displayed.

35. B

In this example 2 text fields are added to the frame, and then the line “frame.setVisible(true)” appears, and then the third text field is added to the frame. After this third text field is added to the frame another line of code appears: “frame.validate()” . This validate method causes the container, the frame, to lay out all of its components again. Therefore all 3 text fields are displayed on the screen.

36. A

In this example 2 TextFields were added to the frame and then the setVisible(true) method was called, so these 2 components are displayed. The third TextField is added after the call to setVisible, so it is not displayed on the frame. The call to repaint() will call update() and then paint(), but does not add the new TextField.

37. A

In this example, since the LayoutManager was set to null, the setBounds() method for the component works, so the button that is added has the location and dimensions set in the setBounds method. Since all of the 3 components are being added to the same place and have the same dimension only the first button added is viewed, it is like the others are placed behind it in layers.

The first button added here is the one with the text “AAA” on it.

38. D

In this example the version of the add() method is the one that accepts both a component and an index. An index of 0 means that the component is placed on the top. This means that if any other components are placed within the same coordinates, the one with the index of 0 will be the one on the top layer. In this case all of the 3 buttons are placed in the same location with the same index, so the last one added will use this 0 index and will be on “top” of the others. The last button added here is the one with the text “AAABBBCCC” on it.

39. E

In this example when the code is run an IllegalArgumentException will be thrown since when adding components with the add() constructor that accepts both a component and an index, the indexes of the components added must be in ascending order, starting from 0.

For example:

```
add(component1, 0);
add(component2, 1);
add(component3, 2);
```

40. A

In this example all of the buttons are added at the same index, 0. Since they are all placed on the exact same position and all have the exact same dimensions, it is like one is being placed on the top of the other, so the last one that was added will be displayed. This is the one that has the text “AAA”.

41. A

If a component is placed in a container after the setVisisble(true) method is called, then the validate method could be called to make that component be visible.

42. A

The paint() method is called automatically by the GUI thread when components are made visible, resized, and sometimes when their state changes due to an event.

43. A, B, C, D

All of these statements are true.