Lesson 13: Introduction to GUI

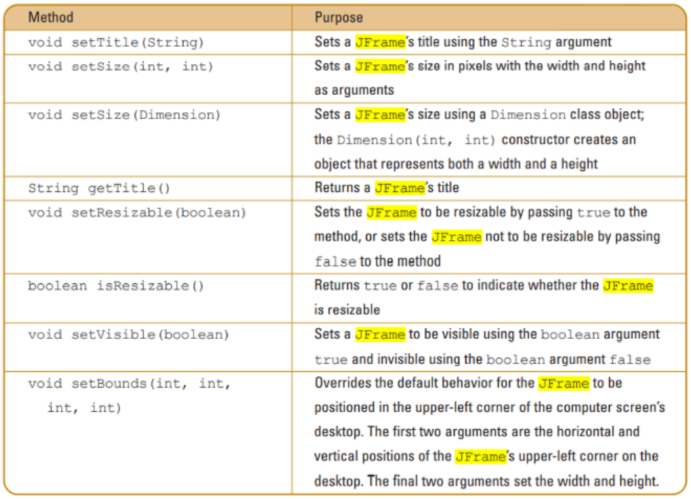
1. GUI (Graphical User Interface) in Java is an easy-to-use visual experience builder for Java applications. It is mainly made of graphical components like buttons, labels, windows, etc. through which the user can interact with an application.

2. Swing in Java is a Graphical User Interface (GUI) toolkit that includes the GUI components. Swing provides a rich set of widgets and packages to make sophisticated GUI components for Java applications.

3. All the components in swing like JButton, JComboBox, JList, JLabel are inherited from the JComponent class which can be added to the container classes. Containers are the windows like frame and dialog boxes.

4. JFrame can be constructed as you do other objects using, class name, identifier, assignment operator. For example, the following two statements construct two JFrames: one with the title "Hello" and another with no title:

```
JFrame firstFrame = new JFrame("Hello");
JFrame secondFrame = new JFrame();
```



Method	Purpose
`void setTitle(String)`	Sets a `JFrame`'s title using the `String` argument
`void setSize(int, int)`	Sets a `JFrame`'s size in pixels with the width and height as arguments
`void setSize(Dimension)`	Sets a `JFrame`'s size using a `Dimension` class object; the `Dimension(int, int)` constructor creates an object that represents both a width and a height
`String getTitle()`	Returns a `JFrame`'s title
`void setResizable(boolean)`	Sets the `JFrame` to be resizable by passing `true` to the method, or sets the `JFrame` not to be resizable by passing `false` to the method
`boolean isResizable()`	Returns `true` or `false` to indicate whether the `JFrame` is resizable
`void setVisible(boolean)`	Sets a `JFrame` to be visible using the `boolean` argument `true` and invisible using the `boolean` argument `false`
`void setBounds(int, int, int, int)`	Overrides the default behavior for the `JFrame` to be positioned in the upper-left corner of the computer screen's desktop. The first two arguments are the horizontal and vertical positions of the `JFrame`'s upper-left corner on the desktop. The final two arguments set the width and height.

```
firstFrame.setSize(250, 100);
firstFrame.setTitle("My frame");
```

the following statements set the firstFrame object's size to 250 pixels horizontally by 100 pixels vertically

5. JLabel Class - It is used for placing text in a container. It also inherits JComponent class.

6. JLabel label_name = new JLabel("Text")- We used the add() method to attach the new Label component to your Frame.
aFrame.add(label_name);

7. JTextField Class - is a component into which a user can type a single line of text data. (Text data comprises any characters you can enter from the keyboard, including numbers and punctuation.)

8. JTextField textField= new JTextField(10)- You can use a method name setBounds(x,y, width, height) to customize the size of the textField. Other methods you might want to consider as well are setText(), setEditable()

9. JButton Class - is a Component the user can click with a mouse to make a selection. A JButton is even easier to create than a JTextField.

10. JButton readyJButton = new JButton("Press when ready")