Prepare for final by going over the information below: Everything below will be covered in the final exam,

- To know the difference between GUI lightweight and heavyweight components.

Answer  
Heavy weight – partially native platform dependant.  
Light weight – totally platform independent.

- To know which components are heavyweight components.  
  
Answers:  
The following are heavyweight components:  
JFrame, JApplet, JDialog

All other GUI components are light weight.

- To know how to register with a JButton object; to be a listener of button clicks.

Answer:  
Call JButton,s addActionListener method.

- To know what kind of Event object is generated for the following actions:  
• [1] Clicking on a button  
• [2] Pressing an enter key inside a text field  
• [3] Typing a letter inside a text field  
• [4] Clicking the close ( X ) button in a JFrame  
• [5] Clicking the minimize ( \_ ) button in a Jframe  
• [6] Pressing, releasing, clicking, entering, exiting a mouse in an object  
• [7] Dragging or Moving a mouse in an object  
Answers:  
• [1] ActionEvent object  
• [2] ActionEvent object  
• [3] ItemEvent object  
• [4] WindowEvent object  
• [5] WindowEvent object  
• [6] MouseEvent object  
• [7] MouseEvent objectm –(there is no MouseMotionEvent object )  
  
- For each of the above cases what interface the listener must implement to listen for the event?  
Answers:  
• [1] implements ActionListener  
• [2] implements ActionListener  
• [3] implements ItemListener  
• [4] implements WindowListener  
• [5] implements WindowListener  
• [6] implements MouseListener  
• [7] implements MouseMotionListener

- To know the methods provided by the ActionEvent object for communicating with it and what each method returns.  
Answers: (ActionEvent provides the following methods)  
• getSource: returns the reference of the source object reporting the event  
• getActionCommand: returns additional information about the source object  
for example in the case of a JButton object, it returns the label of the JButton that the user sees  
  
- To know how to set the background color of a container object such as JFrame or JPanel.  
Answers:  
• For a JPanel object jp you will set the color to green as below:

jp.setBackground( Color.green );  
  
- To know how to get the size, width, height, or dimensions of an object such as JFrame or JPanel.  
Answers:  
• For example, to get the width of a JFrame f, you will do as follows:

f.getWidth( );

Look up the other methods yourself such as for getting height, dimensions etc.  
  
- To know how to set up the layout manager of a container object.  
Answers:  
• You will set up the layout of a JFrame object to be grid layout with four rows and two columns as below:  
Method 1:  
GridLayout gl = new GridLayout( 4, 2 );  
f.setLayout( gl );  
Method 2:  
f.setLayout( new GridLayout( 4, 2 ) );  
  
Do this yourself:  
How will you set up the layout to be BorderLayout?

How will you set up the layout to be FlowLayout such that all the components are left aligned within the FlowLayout?  
Answer  
FlowLayout fl = new FlowLayout (FlowLayout.LEFT);  
f.setLayout (fl);  
  
- To know what is the default layout for the following container objects:   
JFrame  
JApplet  
JPanel   
  
Answer  
JFrame (BorderLayout)  
JApplet, (BorderLayout)  
JPanel (FlowLayout)

- To know the name of the method that you must override for drawing graphics in the following object types:   
JFrame  
JApplet   
JPanel

Answer  
JFrame (paint)  
JApplet (paint)  
JPanel (paintComponent and paint)

- To know how to set the size of a JFrame object and how to make it show up.

Answer   
f.setSize (400,300);

To know how to make a JFrame object show up.

Answer   
f.setVisible(true);

- To know where the coordinates (0, 0) is located in the following drawing objects:  
• JFrame (top-left corner)  
• JApplet (top-left corner)  
• JPanel (top-left corner)  
- To know the method and parameters for drawing the following graphics:  
• Rectangle  
• Oval  
• Arc  
• String

Answer  
drawRect ( )  
drawOval ( )  
drawArc ( )  
drawString ( )

- To know how to set the color and the font of the Graphics object g.  
Answer:  
//Code snippet below sets the font and the color before displaying stirng “Hi”.  
Font font = new Font("Courier",Font.BOLD,30);  
g.setFont(font);  
g.setColor(Color.red);  
g.drawString("Hi",100,100);

- To know how to do the following:  
• Read the contents of a TextField.  
• Change the contents of a TextField.  
• Clear the contents of a TextField.  
• Set the label of a button.  
• Add text to the end of a text area.

Answer  
getText ( )  
setText ( )  
setText (“ “);  
setText ( )  
appent ( )  
- To know the difference between a JComboBox and a JListObject.  
Answers:  
• JComboBox: single item selection.  
• JListBox: single and multiple item selection.  
  
- To know what event is generated for the following actions:  
• Selecting/clicking an item in a JComboBox.  
• Clicking an item in a JList.  
• Clicking a JRadioButton.  
• Clicking a JTextBox.  
• Clicking a menu item.  
• Pushing an Enter Key in a JTextField  
• Entering any key in a JTextField  
Answers:  
• ActionEvent and ItemEvent  
• ActionEvent and ItemEvent  
• ActionEvent and ItemEvent  
• ActionEvent and ItemEvent  
• ActionEvent  
• ActionEvent  
• ItemEvent  
- To know how do you add an item to a JComboBox programmatically.  
-   
Answer: use JComboBox’s method addItem( ).  
The code below creates a combo box and add three items to it.  
JComboBox jcbo = new JComboBox( );  
jcbo.addItem (“apples”);  
jcbo.addItem (“pears”);  
jcbo.addItem (“peaches”);  
- To know how to create the following objects:  
• A JMenuItem object.  
• A JMenu object.  
• A JMenuBar object.  
- To know how to do the following:  
• To add or associate a menu item to a menu.  
• To add or associate a menu to a menu bar.  
• To add or associate a menu bar to a JFrame object.  
• To add a separate bar to a menu.  
Answer  
Call the addSeparator method of the JMenu object to add a separator in a menu.  
  
  
JFrame  
JMenuBar  
JMenu  
JMenuItem

How to create the following two menus each with two menu items shown.   
File  
Open  
Close  
Edit  
Copy  
Paste

//The example code below creates the menu and menu items shown above.

// create objects  
JMenuItem jmiOpen = new JMenuItem( “Open” );  
JMenuItem jmiClose = new JMenuItem( “Close” );  
  
JMenuItem jmiCopy = new JMenuItem( “Copy” );  
JMenuItem jmiPaste = new JMenuItem( “Paste” );  
  
JMenu jmuFile = new JMenu( “File” );  
JMenu jmuEdit = new JMenu( “Edit” );  
  
JMenuBar jbar = new JMenuBar( );  
  
// start adding  
// add items to menus  
// consider JMenu a container  
  
jmuFile.add( jmiOpen );  
jmuFile.add( jmiClose );  
  
jmuEdit.add( jmiCopy );  
jmuEdit.add( jmiPaste );  
  
// add menus to bar  
// consider JMenuBar a container  
  
jbar.add( jmuFile );  
jbar.add( jmuEdit );  
  
// don’t add JMenuBar to contentPane  
// instead set it to JFrame as shown below:  
  
f.setJMenuBar( jbar );

//Not above f is the JFrame or JFrameExt object.  
  
- To know how to add a separator bar to a menu:   
find out yourself