Lab # 3 (The View)

Part I, Experiment − classtime, Tuesday, Feb 18th, 2019.

Part II, Program - by midnight, Tuesday, Feb 18th, 2019.

This assignment will require that you submit both a *Programming Assignment* (Part II) via D2L and an *Experimentation Assignment* (Part I) − to be printed off and passed in at the beginning of _class_ on the date due (Tuesday Sept 25th).

Note also that we provide a zipped folder (i.e. images.zip) that you have to unzipp to get the required images.

The Experiment will NOT be accepted late!!

Part I MUST be typed and submitted to your instructor at the BEGINNING of class on the due date. IT WILL NOT BE ACCEPTED LATE. *Failure to print, left it in your dorm room, etc. are unacceptable excuses.*

Part II is the programming component. It should be submitted the Lab3.java that you are instructed to build for every Java **JFrame** that you create in this class using D2L to submit the **Lab3.java** file for grading.

You should be aware that there is a Java reference readily available on the web (https://docs.oracle.com/javase/7/docs/api/) that can be used to find information on all Java packages, classes, methods, etc. Some questions in Part I are designed to help familiarize you with this web site.

Part I. Experiment. (40%)

For Part I you should create a Lab3Experiment project using the Lab3Experiment.java file provided. The program is a good example of GUI design and programming. It does, however, uses some HTML code without much explanation. That, on top of all the new material on visual design, can be a bit overwhelming. Do not worry much about that. Just run the code to see how nested JPanels work using BorderLayout and GridLayout.

To help understand these new concepts, we will approach the program in two steps. We will first study the program as the Lab3ExperimentCode, you should understand this code from material covered in class. Then you will modify it to answer some questions.

In general, you will want to take a top-down approach to writing your programs. That is, consider the GUI hierarchy first and then come back and fill in the details of all the pieces.

Answer these questions by experimenting with Lab3Experiment.java file. You should run

the program after each modification, observe and record the results, and then always restore the applet to its original form before proceeding. In cases where you are asked to write original code, you should give the EXACT code required.

After your have run the code as it is given modify the code according to what is asked and report the results.

1. Consider the method setupButton(JButton b) and remove the following lines of code

   a) b.setContentAreaFilled(false)

   b) b.setBorderPainted(false);

   c) b.setForeground(tanColor);

After each step restore code back to original and try the next. Describe the results of each change.

2. Consider the method doLeftSide( ) and change the following lines of code

   a) change the line   add(leftside, BorderLayout.WEST);

              for line   add(top, BorderLayout.EAST);

   b) change the line   add(leftside, BorderLayout.WEST);

              for line   add(top, BorderLayout.SOUTH);

   c) remove the line   leftside.setBackground (darkColor);

Describe the results of each change.

3. Consider the method doBottom ( ) and change the following lines of code

   a) change the line   copyright.setOpaque(true);

              for   copyright.setOpaque(false);

Describe the results of each change.

4. Consider the method doCenter ( ) and change the following lines of code

   a) change the line   separator.setPreferredSize(new Dimension(10, 20));

              for   separator.setPreferredSize(new Dimension(40, 40));

   b) change the line   center.add(separator, BorderLayout.NORTH);

              for   center.add(separator, BorderLayout.EAST);

c)   remove   the   line   separator.setBackground   (darkColor   );

Describe the results of each change.

5. Consider the method doCenter ( ) and change the following lines of code

   a) change the line   welcomePanel.add(welcomeMsg);

                for   welcomePanel.add(welcomeMsg, BorderLayout.NORTH);

   b) change the line   freeNotebook.setOpaque(false);

                for   freeNotebook.setOpaque(true);

   c) change the   lines   welcomePanel.add(freeNotebook, BorderLayout.SOUTH);

                     center.add(welcomePanel, BorderLayout.CENTER);

             for the line   center.add(freeNotebook, BorderLayout.SOUTH);

Describe the results of each change.

Part II. Programming. (60%)

<u>Note</u>: this assignment will require only a constructor method Lab3 (*and any other user-defined methods that you write*). Remember also that all your classes must include a header with comments similar to the one for Lab1.
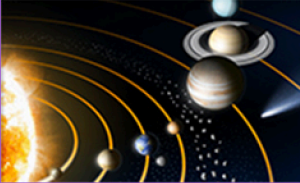
Continuing with the theme the Solar System Selector in this lab you should improve the look the GUI using nested JPanels with BorderLayout and GridLayouts to manage the display. Also you have to add color and font sizes to the various panels. Remember to maintain the minimum number of features for your JFrame as described in Lab 2

Your applet should illustrate the use of **<u>ALL</u>** common widgets discussed in class as well as your skills to show images within the widgets and the right organization of the layout using JPanels. Please feel free to be creative, here we provide our applet demo so you can get the look and feel we expect from your own implementation.

As an example of the GUI we give you the following sample GUI It should look like this

Select Planet
Mercury
Venus
Earth
Mars
Asteroids
Jupiter
Saturn
Uranus
Neptune
Pluto

Select Fact

☐ **show moons**

☐ **in English Units**

Fact **units**

Select   Clear

SOLAR SYSTEM SELECTOR

your weight   in this planet would be

Solar System Selector simple guide
_____
Select a Planet to display its most
important facts and characteristics
then check the options of your preferences
Then press the SELECT button
_____
For more detailed information check the NASA site
URL= http://solarsystem.nasa.gov/index.cfm