CST8221 – Java Application Programming
Hybrid Activity #3
Layout Managers

Terminology

A **Layout Manager** is a specialized class that is responsible for arranging the GUI components placed in a container. All layout managers implement a common interface ***java.awt.LayoutManager***. Developers can use one of the predefined layout managers or can create their own, but it is not an easy task. Currently, Java API provides the following layout managers:

java.awt

- BorderLayout
- FlowLayout
- GridLayout
- CardLayout
- GridBagLayout

javax.swing

- BoxLayout
- SpringLayout
- GroupLayout

javafx.scene.layout

- AnchorPane
- BorderPane
- FlowPane
- GridPane
- Pane
- StackPane
- TilePane
- HBox and VBox

In Swing all containers have a default layout manager. The default layout manager for the top-level containers **JFrame**, *JWindow*, and *JDialog* is ***BorderLayout***. The default layout for *JPanel* and *JApplet* is ***FlowLayout***. Once the container is created the default layout can be change by calling the ***setLayout()*** method of the container. The layout manager can be set to ***null***, in which case the developer must take the responsibility to arrange the components (In JavaFX Pane should be used for that purpose).

The Nature of Things

In general, there are three way to arrange components in a GUI container

1. Using existing layout managers: This is the easiest way and it works very well for not very complex GUIs. All the arrangements are decided by the layout manager. The size of the components and the positioning is determined by the layout manager. As you already know, containers can be placed into other containers. Setting different layout managers for different containers allows creating sophisticated GUIs with relative ease.
2. Using absolute positioning: Provides almost complete control over the GUI. By setting the container's layout to *null,* the developer must specify the size and the absolute position of

each component with respect to the upper-left corner of the container. Programming such GUI could be a difficult task unless an IDE can generate the code.

3. Using a visual programming (GUI design tools) in an IDE. The IDI must provide a tool that allows the developer to "paint" the GUI, and then the tool will generate the Java code for the GUI. Of course, the event handling code is still responsibility of the developer. For example, NetBeans IDE provides such a tool (you will try it later in one of the lab exercises). The painting tool can use absolute positioning or some of the preexisting layout manager. The most commonly used are *GridBagLayout* and *CroupLayout*. They were created to be used specifically in GUI building tools.

We will discuss the basic layout managers *BorderLayout*, *FlowLayout*, and *GridLayout* in class.
In **JavaFX** there are similar layout managers, but they are called layout *Panes*.

References

Textbook 1 – Chapter 14. Textbook 2 – Chapter 14
Java Swing, second edition.
Links:
http://download.oracle.com/javase/tutorial/uiswing/layout/index.html
http://docs.oracle.com/javafx/2/layout/builtin_layouts.htm

Code Examples

Code examples are provided for most of the layout managers that are easy to use without specialized IDE GUI design tools.
You will find the examples in ***CST8221_HA03_code_examples.zip***.

Exercise

Download, compile, and run the code examples. Once you see how they work, explore carefully the code. Concentrate your efforts on the **basic layout managers (Flow, Border, and Grid),** but also pay close attention to the *BoxLayout* examples. *BoxLayout* is used to create toolbars. *BoxLayout* type layouts (*Hbox* and *VBox*) play significant role in JavaFX.

Questions

Q1. What is the default layout manager of *JPanel*?
Q2. Can you change the layout manager of a container once the container is created?
Q3. Can developers create their own layout managers?
Q4. Does JavaFX provide predefined layout managers?

Submission

No submission is required for this activity.

Marks

No marks are allocated for this activity, but do not forget that understanding how to use layout managers is essential for building effective GUI applications.

And do not forget that:
"Organizing is what you do before you do something, so that when you do it, it's not all mixed up."
(*A.A. Milne, English author and creator of Winnie-the-Pooh*)

but also never forget that*:*

“The trouble with organizing a thing is that pretty soon folks get to paying more attention to the organization than to what they’re organized for.”
(*American author Laura Ingalls Wilder, who wrote Little Town on the Prairie*)