Lea Trueworthy

Professor Woods

CSD 402 Java for Programmers

2 March 2025

M11: Programming Assignment

Let’s begin by taking a look at what JavaFX is, where it came from and why it’s important. GeeksForGeeks explains JavaFX as “a Java library and a GUI toolkit designed to develop and facilitate Rich Internet applications, web applications, and desktop applications.” Basically, JavaFX is an open-source tool that allows developers to design, create, test, debug and deploy an application on just about any platform. This means only one library is required versus when developers and programmers needed a considerable amount.

Originally named Form Follows Functions (F3) project by Chris Oliver, before the project was purchased in 2005 by Sun Microsystems where the name changed to JavaFX. After many years of developments and releases, Oracle Corporation acquired Sun Microsystems in 2009 where it remains to this day.

“In JavaFX, Layout describes the process in which the elements are to be viewed on the screen. It primarily establishes the scene-graph nodes” (GeeksForGeeks). There are several built-in layouts, and each layout is showcased as a separate class. “All these classes belong to the javafx.scene.layout package. The javafx.scene.layout.Pane class is the root class for all the built-in classes of JavaFX layouts (GeeksForGeeks). At the end of the day, the goal is to have the layout, meaning the children, automatically update when the screen is resized.

Let’s explore a couple of JavaFX classes, FlowPane and GridPane. We’ll start with the FlowPane class.

**FlowPane**

FlowPane is one of JavaFX layout classes that arranges its children (or nodes) horizontally, from left to right, or vertically, from top to bottom, within a given boundary. At first glaze, it would appear to be similar to another class called HBox but the key difference is that when there is no more horizontal room, the overflow wraps itself to the next line making this useful for dynamic layouts.

Key Features

Orientation: To summarize what Jakob Jenkov speaks on, the default flow orientation the nodes are configured to is the horizontal layout, but you can change this to use the vertical layout. The result would follow the flow of the columns, top to bottom, instead of the rows, left to right.

Alignment: Child nodes can also be aligned in various ways such as left, right or center.

Spacing: Spacing refers to the gap, or pixels, between the children but instead of one property you need to have horizontal and vertical spacing.

Wrapping: One of the most important aspects to this class is the ability to wrap child nodes to the next available row or column allowing for a more fluid arrangement of components.

FlowPane Example

In this example, we create five buttons and add them to the FlowPane. The layout will adjust depending on the window size, automatically wrapping the buttons to the next line when necessary. This is particularly useful for responsive interfaces where the window size can change.

**GridPane**

GridPane, on the other hand, is another JavaFX layout class that arranges its children using a grid, or matrix to position the components. The special rule is that “in a row, all cells will be having the same height whereas, in a column, all cells will have the same width” (Victor). GridPane requires you to specify the coordinates when adding new child.

Key Features

Rows and columns: “… columns and rows are sized based on the components inside. Fortunately, you can have better control over how individual columns and rows are sized” (Ruzicka, “JavaFX Tutorial: Advanced Layouts”). Though GridPane takes a little bit more time to understand, the result allows for better structured and a more precise layout control.

Spacing: Just like FlowPane, spacing refers to the gap, or pixels, between the children that are positioned side by side to each other. This is defined as either horizontal or vertical spacing.

Span: This feature allows the ability to extend an item across multiple rows and multiple columns.

GridPane Example

In this example, we create two labels and two text fields and arrange them within a GridPane. The layout specifies that the labels go into the first column, and the text fields go into the second column. The horizontal and vertical gaps between the components are controlled using setHgap() and setVgap().

**Conclusion**

The FlowPane class is best used when a UI layout needs to be arranged back-to-back. It will continue to flow to the next line due to the previous line not having any more room, preferably using dynamic components. A good example would be a photo gallery. The GridPane class is best when it comes to something more structured like when creating forms or tables. A good example would be a login form. In conclusion, deciding which class to use depends on what exactly you are trying to accomplish, so choose wisely.

References:

Damm, Sebastian. “JavaFX 2.0 Layout Panes – FlowPane and TilePane.” *Java Code Geeks*, 10 July 2012, <http://www.javacodegeeks.com/2012/07/javafx-20-layout-panes-flowpane-and.html>. Accessed 26 Feb. 2025.

GeeksForGeeks. “JavaFX Tutorial.” *GeeksforGeeks*, 5 Oct. 2021, [www.geeksforgeeks.org/javafx-tutorial/](http://www.geeksforgeeks.org/javafx-tutorial/). Accessed 26 Feb. 2025.

Jenkov, Jakob. “JavaFX GridPane.” *Jenkov.com*, 29 Apr. 2020, <https://jenkov.com/tutorials/javafx/gridpane.html>. Accessed 26 Feb. 2025.

Ngugi, Paul. “Layout Panes.” *DEV Community*, 20 June 2024, <https://dev.to/paulike/layout-panes-2n8n>. Accessed 26 Feb. 2025.

Oracle. “JavaFX Layout Controls - Dev.java.” *Dev.java: The Destination for Java Developers*, 12 Sept. 2023, <https://dev.java/learn/javafx/layout/>. Accessed 26 Feb. 2025.

Prerna Tiwari. “JavaFX FlowPane.” *Naukri.com*, 26 Mar. 2024, [www.naukri.com/code360/library/javafx-flowpane](http://www.naukri.com/code360/library/javafx-flowpane). Accessed 26 Feb. 2025.

Ruzicka, Vojtech. “JavaFX Tutorial: Advanced Layouts.” *Vojtech Ruzicka’s Programming Blog*, Vojtech Ruzicka, 4 Oct. 2019, [www.vojtechruzicka.com/javafx-layouts-advanced/](http://www.vojtechruzicka.com/javafx-layouts-advanced/). Accessed 26 Feb. 2025.

---. “JavaFX Tutorial: Basic Layouts.” *Vojtech Ruzicka’s Programming Blog*, 1 Oct. 2019, [www.vojtechruzicka.com/javafx-layouts-basic/](http://www.vojtechruzicka.com/javafx-layouts-basic/). Accessed 2 Mar. 2025.

Victor, Princy. “JavaFX GridPane.” *EDUCBA*, 13 Dec. 2019, [www.educba.com/javafx-gridpane/](http://www.educba.com/javafx-gridpane/). Accessed 26 Feb. 2025.