НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Object-Oriented Programming in the Java language

Part 8. GUI: Java FX

Yevhen Berkunskyi, NUoS
eugeny.berkunsky@gmail.com
http://www.berkut.mk.ua

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

What JavaFX is?

- JavaFX is a framework and (large) set of objects we can use to develop GUI-based applications
- The JavaFX API is a good example of how object-oriented principles can be applied in software development

JavaFX vs Swing vs AWT

- Java was first released with GUI support in something called the Abstract Windows Toolkit (**AWT**)
- **AWT** wasn't bad, but it had some limitations, and some particular problems with how it was implemented on some platforms
- Ultimately, **AWT** (which still exists, but isn't used much anymore) was replaced by a new library called **Swing**, which was more versatile, more robust, and more flexible.

JavaFX vs Swing vs AWT

- **Swing** was designed primarily for use in desktop applications (although you could do some web-based things with it, too).
- **Swing** has now been replaced by a completely new GUI library called **JavaFX**
- You can still use **Swing** (for the foreseeable future), but Oracle isn't going to develop it any further – it's essentially a dead-end technology
- Java has replaced **Swing** with **JavaFX**
- How long until **JavaFX** is replaced by something else? Nobody knows; probably many years

The Basic Structure of a JavaFX Program

- JavaFX programs all start not as some “regular” class like we’ve been doing, but as an extension of the abstract Application class in JavaFX, javafx.application.Application

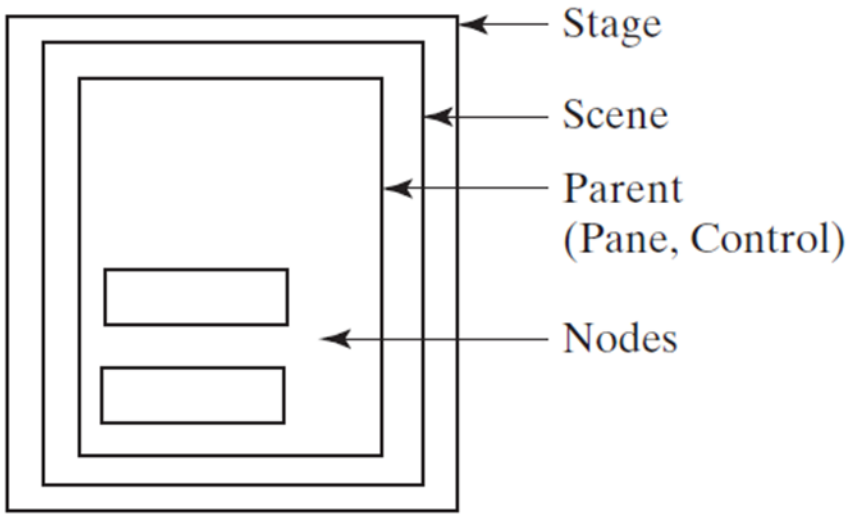
```
public class MyProgram {
    // Body of class
}
```

Becomes:

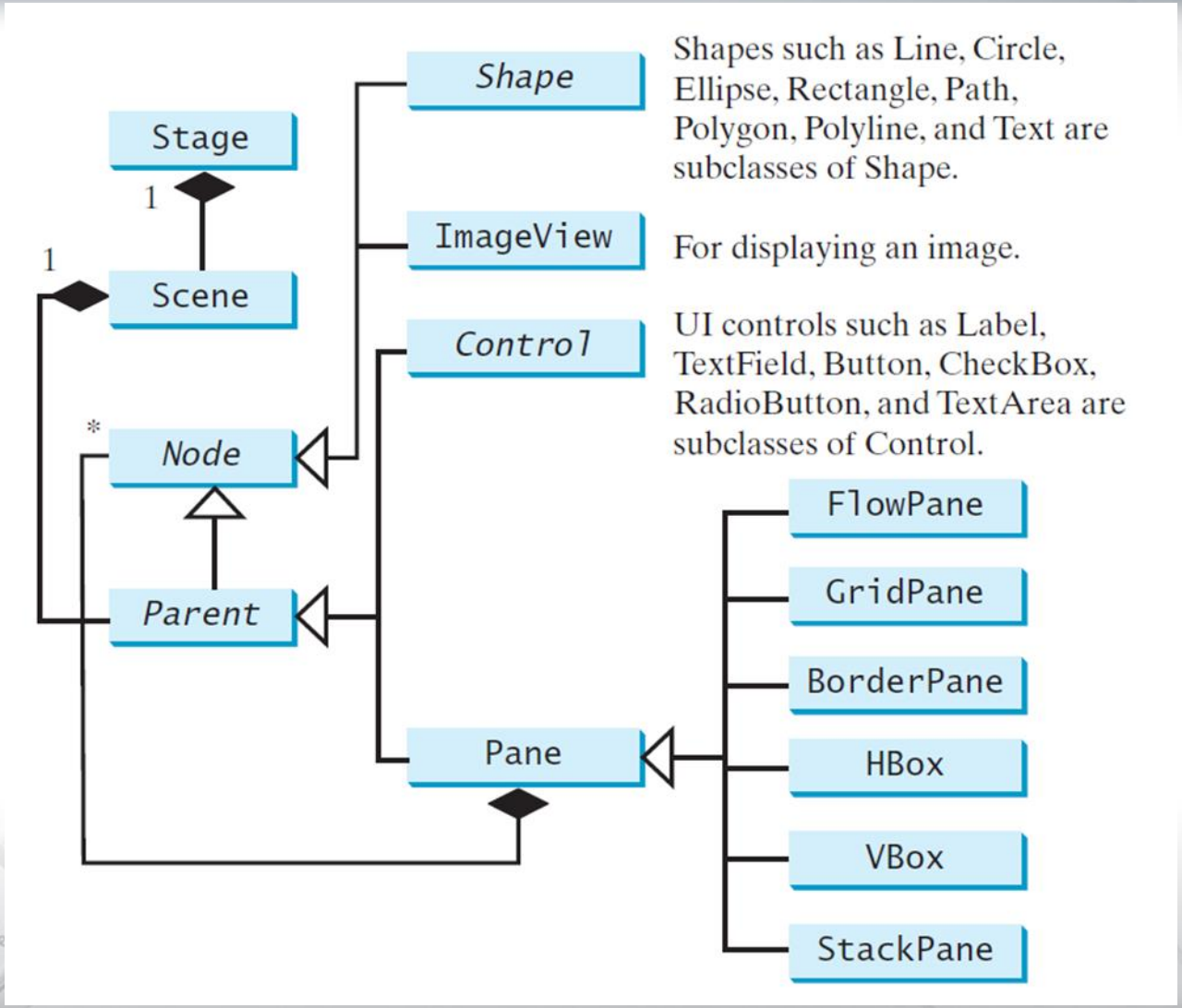
```
import javafx.application.Application;
…
public class MyProgram extends Application {
    // Body of class
}
```

Panes, UI Controls, and Shapes

- We can put the button directly on the scene, which centered the button and made it occupy the entire window.
- Rarely is this what we really want to do
- One approach is to specify the size and location of each UI element (like the buttons)
- A better solution is to put the UI elements (known as nodes) into containers called panes, and then add the panes to the scene



Panes, UI Controls, and Shapes



НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Panes, UI Controls, and Shapes

- Beyond the obvious, typical “active” UI elements (things we can interact with, like buttons, etc.), are static shapes – lines, circles, etc.
- Before we can do much with shapes, we have to talk about coordinates within a pane.
- The top-left corner of a scene is *always* (0, 0), and the (positive) X-axis goes to the right, and the (positive) Y-axis goes down. Visually, we’re in Cartesian quadrant IV, but Y stays positive.
- All coordinates are in pixels

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Example

Common Properties and Methods for Nodes

- Nodes share many common properties
- JavaFX style properties are a lot like CSS (Cascading Style Sheets) use to specify styles in HTML (Web) pages.
- For more on HTML and CSS, see Liang supplements V.A and V.B (on the text's companion website)
- Thus, it's known as JavaFX CSS

The Color Class

- There are 3 sets of color constructors (“mixers”):
- The ones named Color (or color) require double values ϵ [0.0, 1.0] for the R/G/B/A components
- The ones named rgb require int values ϵ [0, 255]
- The hsb / hsba color models are also supported
- Just like String, Color is immutable.
- If we want a lighter version of this Color, we can use .lighter(), but we get a NEW color, rather than changing the value of the current color, just like .toUpperCase doesn’t change a String; it gives us a new one with upper case characters.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Example

Layout Panes

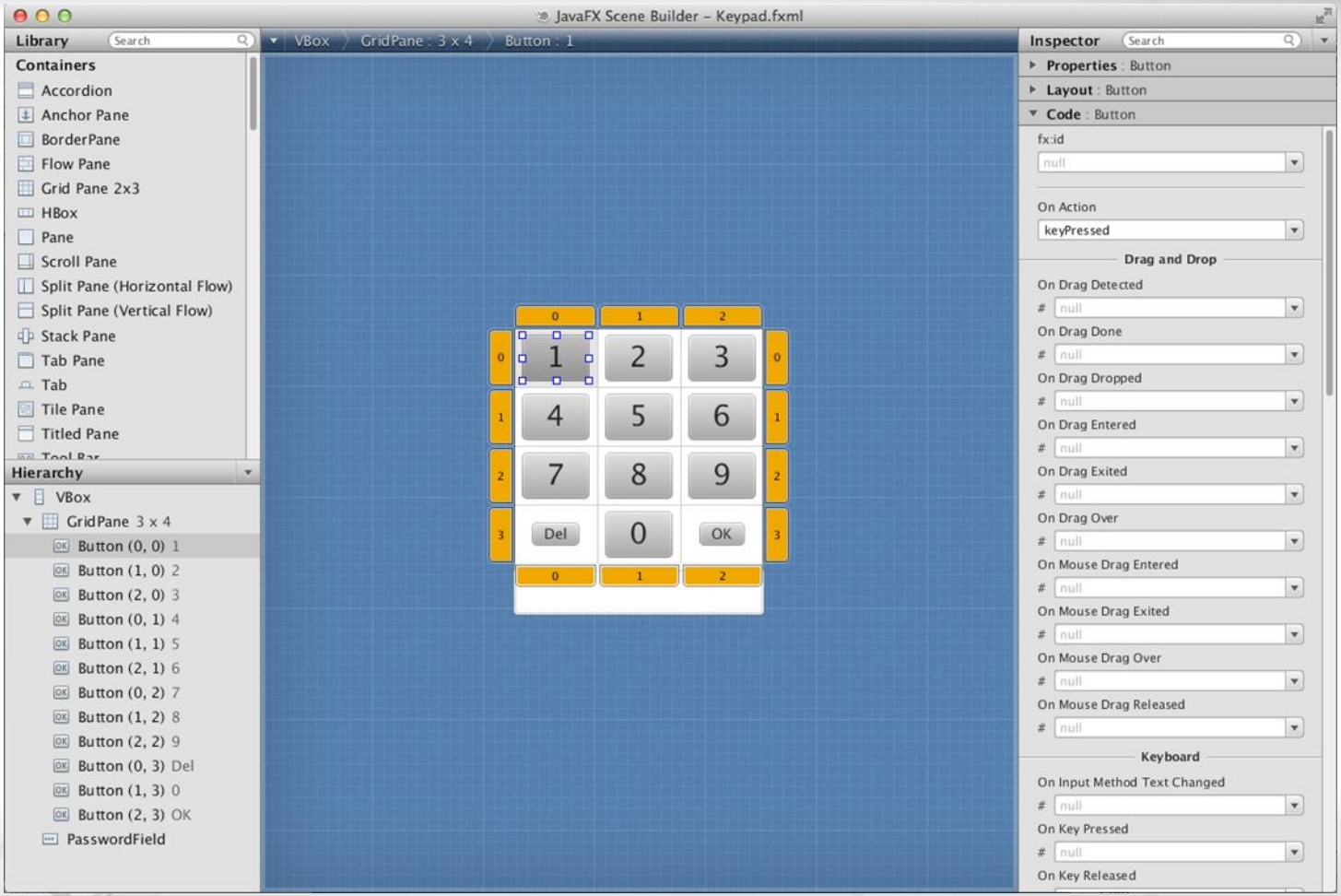
- As we've said (and have been doing), we add our Nodes to a Pane, and then add the Pane to a Scene, and then the Scene to a Stage (probably the primary stage).
- How do we arrange (i.e., lay out) the Nodes on the pane?
- Java has several different kinds of Panes that do a lot of the layout work for us.

Layout Panes

Name	Description
`Pane`	Base class for layout panes. Use its `getChildren()` method to return the list of nodes on the pane (or add to that list) Provides no particular layout capabilities – it's a "blank canvas" typically used to draw shapes on
`StackPane`	Places nodes on top of each other in the center of the pane
`FlowPane`	Places the notes row-by-row horizontally or column-by-column vertically (reading order)
`GridPane`	Provide a 2-D grid of cells, into which we can place nodes
`BorderPane`	Divides pane into top, bottom, left, right, and center regions
`HBox`	Places nodes in a single (horizontal) row
`VBox`	Places nodes in a single (vertical) column

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Scene Builder



НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Scene Builder

- GUIs are created a lot faster than in Swing and AWT
- More sophisticated and aesthetically pleasing UIs
- Easy integration of sounds, images and videos and of web content
- Code is simplified in JavaFX by separating the UI from the logic of the application
- JavaFX can be integrated in Swing applications, allowing for a smoother transition

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Example

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Questions?

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ ІМЕНІ АДМІРАЛА МАКАРОВА

Object-Oriented Programming in the Java language

Part 8. GUI: Java FX

Yevhen Berkunskyi, NUoS
eugeny.berkunsky@gmail.com
http://www.berkut.mk.ua