Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

Java GUI

Java Programming language was released in 1995 and in 1996 released the built-in GUI toolkit called the Java Foundation Classes or Abstract Window Toolkit (AWT). Later this was renamed to Swing and updated the library with new features (Swing was developed by Sun Microsystems). Swing is one of the most used GUI libraries for Java programming until the new framework was released in 2007 by Sun Microsystems, JavaFX. JavaFX is the most updated version of Java GUI programming libraries and it was developed by Sun Microsystems and later acquired by Oracle in 2010. Now let's look more closely at these technologies to create Graphical User Interface programs using Java programming language.

Swing is an updated version of the previous framework for GUI programming, and it incorporates features of Netscape's Internet Foundation Classes (IFC), the Java Foundation Classes (JFC), IBM's Taligent division, and Lighthouse Design. The java swing version 1.4 contains 85 public interfaces and 451 public classes. The image below represents the hierarchy of the java swing API and some of its classes:

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

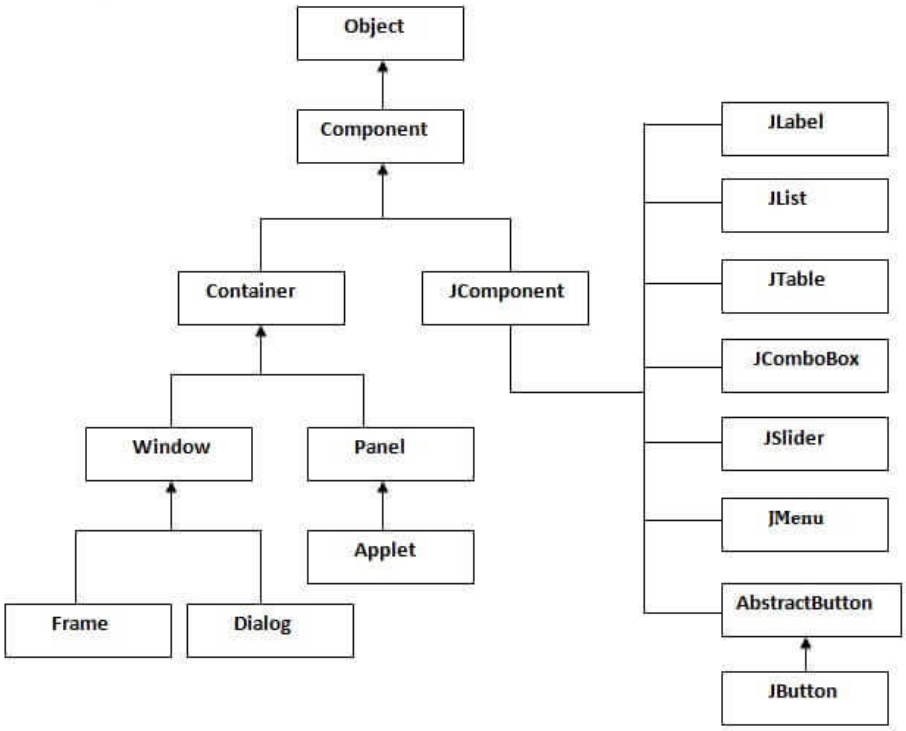


Figure 1.0 (source: https://static.javatpoint.com/images/swinghierarchy.jpg)

A really great aspect of Java Swing is the fact that it is built upon the AWT libraries and it implements new classes and interfaces to improve GUI programs. Some of the classes of the Swing library are but are not limited to the following: JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser, and many more.

JavaFX is the newest version of Java GUI libraries and it is still used today, even if many continue to use the Swing library. JavaFX was created to simplify and give more flexibility in the creation of Graphical User Interfaces and is highly portable. With JavaFX, you can develop programs for desktops, mobile TVs, etc…

JavaFX implements many new features which are useful for various programs. First of all, it includes all the classes and interfaces of Java swing, plus it includes new features such as Media,

user interface controls, web, 2D and 3D, and many more. JavaFX was independently developed and also it can run on Java Virtual Machines. It uses hardware acceleration to provide better performance than Swing, and its look and feel can be customized using CSS. JavaFX has been included in the Java SE Development Kit (JDK) since version 7. The following tables help us understanding the different features available with JavaFX

Feature	Description
Java Library	It is a Java library which consists of many classes and interfaces that are written in Java.
FXML	FXML is the XML based Declarative markup language. The coding can be done in FXML to provide the more enhanced GUI to the user.
Scene Builder	Scene Builder generates FXML mark-up which can be ported to an IDE.
Web view	Web pages can be embedded with JavaFX applications. Web View uses WebKitHTML technology to embed web pages.
Built in UI controls	JavaFX contains Built-in components which are not dependent on the operating system. The UI components are just enough to develop a full featured application.

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

CSS like styling	JavaFX code can be embedded with the CSS to improve the style of the application. We can enhance the view of our application with the simple knowledge of CSS.
Swing interoperability	The JavaFX applications can be embedded with swing code using the Swing Node class. We can update the existing swing application with the powerful features of JavaFX.
Canvas API	Canvas API provides the methods for drawing directly in an area of a JavaFX scene.
Rich Set of APIs	JavaFX provides a rich set of API's to develop GUI applications.
Integrated Graphics Library	An integrated set of classes are provided to deal with 2D and 3D graphics.
Graphics Pipeline	JavaFX graphics are based on Graphics rendered pipeline(prism). It offers smooth graphics which are hardware accelerated.
High Performance Media Engine	The media pipeline supports the playback of web multimedia on a low latency. It is based on a Gstreamer Multimedia framework.
Self-contained application deployment model	Self Contained application packages have all of the application resources and a private copy of Java and JavaFX Runtime.

This table was retrieved from the following source: https://www.javatpoint.com/javafx-tutorial

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

Now after the explanation of what these technologies are and their history we can look at two examples to understand better what are the differences in actual programming/syntax.

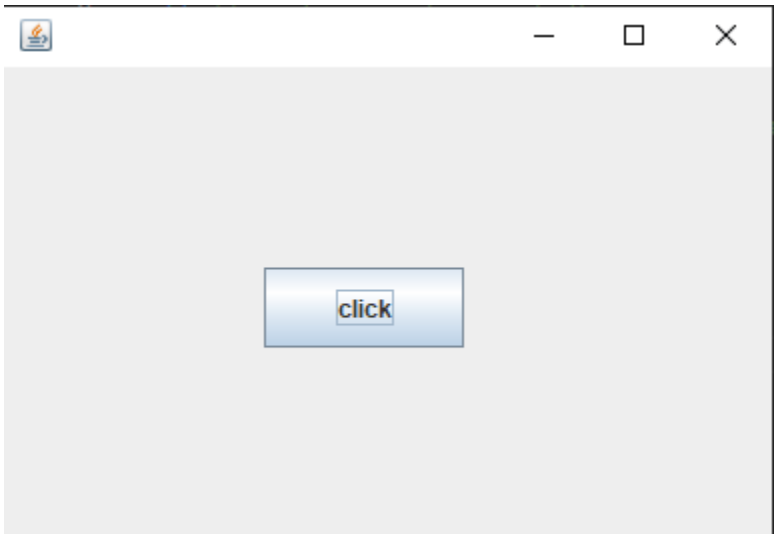
Java swing example:

```
import javax.swing.*;

public class FirstSwingExample {

public static void main(String[] args) {

JFrame f=new JFrame();//creating instance of JFrame


JButton b=new JButton("click");//creating instance of JButton

b.setBounds(130,100,100, 40);//x axis, y axis, width, height


f.add(b);//adding button in JFrame


f.setSize(400,500);//400 width and 500 height

f.setLayout(null);//using no layout managers

f.setVisible(true);//making the frame visible

}

}
```

Example retrieved from the following source: https://www.javatpoint.com/java-swing

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

Output of the example above:



Run on own environment

While to do the same program of displaying a button using JavaFX we can look at the following code:

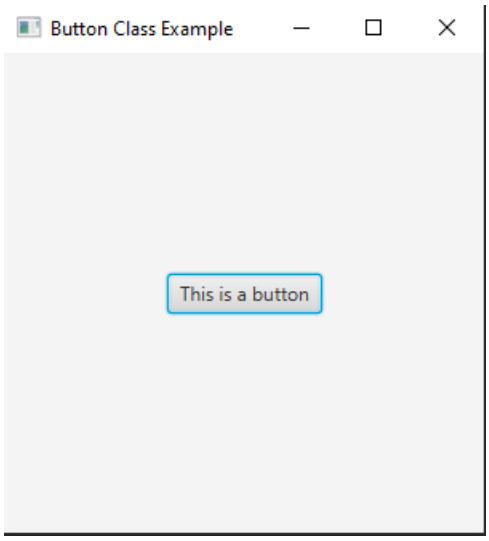
```
package application;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.layout.StackPane;

import javafx.stage.Stage;


public class ButtonTest extends Application {
```

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

```
    @Override
    public void start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub

        StackPane root = new StackPane();
        Button btn=new Button("This is a button");
        Scene scene=new Scene(root,300,300);
        root.getChildren().add(btn);
        primaryStage.setScene(scene);
        primaryStage.setTitle("Button Class Example");
        primaryStage.show();

    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

Output of the program above:

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10



Retrieved from own environment

Andrea Marcelli
CPSC 24500 NET02
April 3, 2023
Homework #10

References

Javatpoint. (n.d.). Java Swing. Retrieved April 3, 2023, from https://www.javatpoint.com/java-swing

Oaks, S., & Wong, H. (2014). Java™ How to Program (Early Objects) (Ninth Edition). Pearson Education, Inc. https://www.oreilly.com/library/view/javatm-how-to/9780133813036/

Swing (Java). (n.d.). In Wikipedia. Retrieved April 3, 2023, from https://en.wikipedia.org/wiki/Swing_(Java)

Zukowski, J. (2002). Java Swing, 2nd Edition. O'Reilly Media, Inc.

Wikipedia contributors. (2022, March 28). JavaFX. In Wikipedia, The Free Encyclopedia. Retrieved 21:17, April 3, 2023, from https://en.wikipedia.org/wiki/JavaFX

Javatpoint. (n.d.). JavaFX Button. Retrieved April 3, 2023, from https://www.javatpoint.com/javafx-button.

Horstmann, C. S., & Cornell, G. (2018). Core Java SE 9 for the Impatient: A Guide to Modern Java APIs. Addison-Wesley Professional. https://www.oreilly.com/library/view/core-java-se/9780134694822/