Lecture 4 - Java Graphical User Interface (GUI): JavaFX Part I

Guiliang Liu

The Chinese University of Hong Kong, Shenzhen

CSC-1004: Computational Laboratory Using Java
Course Page: [Click]

Java Graphical User Interface (GUI)

what is Java Graphical User Interface (GUI)?

ChatGPT-4o-Latest Poe

What is Java Graphical User Interface (GUI)?

A **Java Graphical User Interface (GUI)** is a user-friendly interface that allows users to interact with Java applications using graphical components such as buttons, text fields, windows, and menus, instead of text-based commands.

Java provides several frameworks and libraries to develop GUI-based applications, including **Swing, JavaFX, and AWT**.

香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

Java Graphical User Interface (GUI)

Feature	AWT	Swing	JavaFX
Best for	Legacy applications	Desktop applications	Modern applications
Performance	Fast but outdated	Slightly slower	Optimized with GPU acceleration
Customization	Limited	Flexible	Highly customizable with CSS
Ease of Use	Simple but outdated	Requires more code	Supports FXML (easier UI design)
Mobile Support	No	No	Yes (via **Gluon**)

香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

JavaFX

- JavaFX is a Java library used to develop Desktop applications as well as Rich Internet Applications (RIA).

- The applications built in JavaFX, can run on multiple platforms including Web, Mobile and Desktops.

JavaFX

JavaFX features include:

- **A set of graphics and media packages**: It provides a powerful 2D and 3D graphics engine that allows developers to create rich content applications.

- **Scene Builder**: A visual layout tool for designing JavaFX application interfaces.

- **Rich Set of UI Components**: like buttons, text fields, tables, trees, tabs, charts, and more, which can be easily integrated into applications.
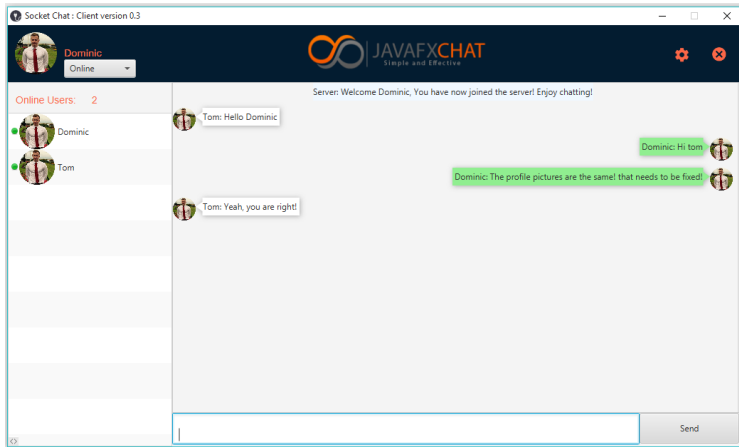
香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

JavaFX

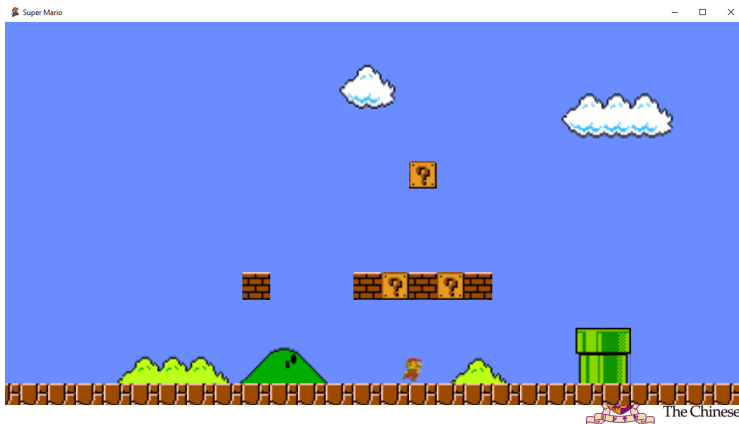JavaFX Applications Example 1: Chatroom (GitHub Project)

JavaFX

JavaFX Applications Example 2: SuperMario (GitHub Project)

JavaFX

JavaFX Applications Example 3: Gomoku(Example Project will be released later)
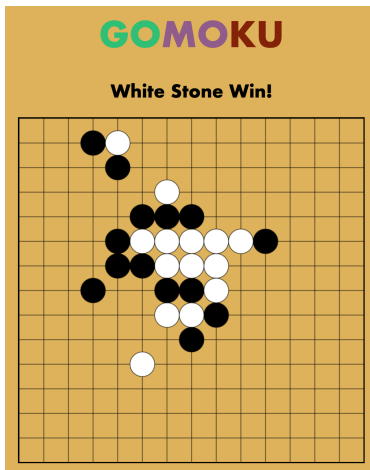
JavaFX Application Structure

JavaFX application is divided into **Stages**, **Scenes** and **nodes**.

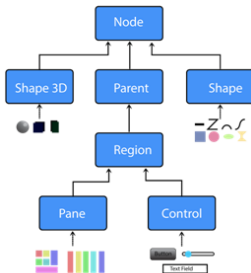- **Stage** acts like a container for all the JavaFX objects.
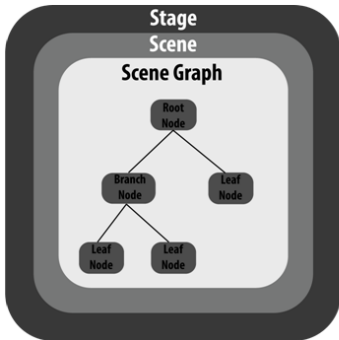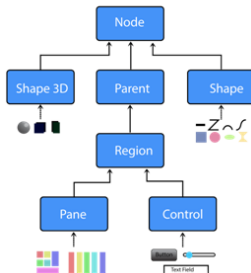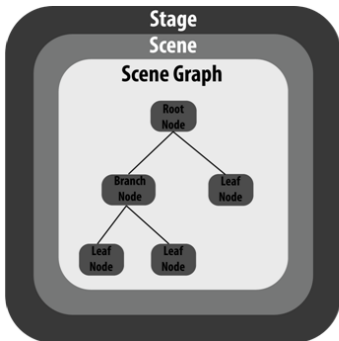
香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

JavaFX Application Structure

JavaFX application is divided into **Stages**, **Scenes** and **nodes**.

- **Scene** holds all the physical contents (nodes) of a JavaFX application. The object of the primary stage is passed to the **start()** method. We need to call **show()** method on the **primary stage object** to show our primary stage.

JavaFX Application Structure

JavaFX application is divided into **Stages**, **Scenes** and **nodes**.

- **Scene Graph** can be seen as the collection of various **nodes**. A node is an element that is visualized on the stage. It can be any button, text box, layout, image, radio button, check box, etc.
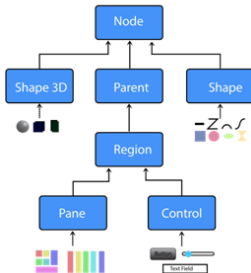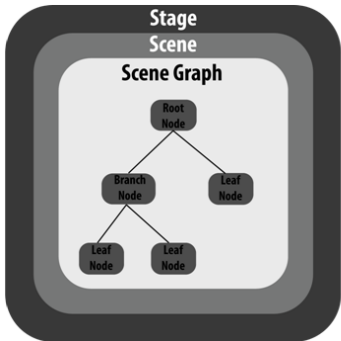
JavaFX Application Structure

JavaFX application is divided into **Stages**, **Scenes** and **nodes**.

- An example of **JavaFX structure**.
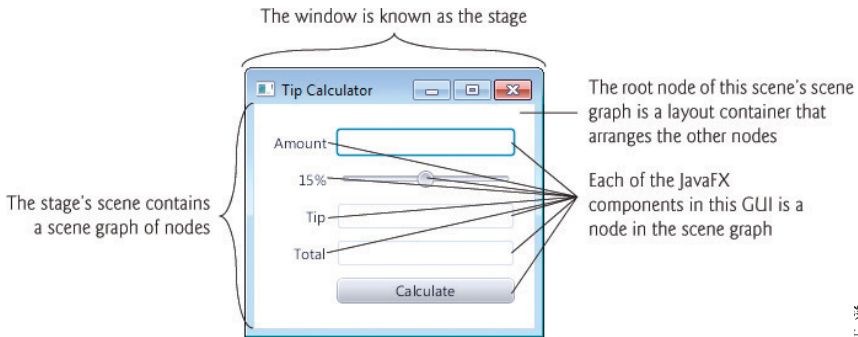
The window is known as the stage

The root node of this scene's scene graph is a layout container that arranges the other nodes

Each of the JavaFX components in this GUI is a node in the scene graph

The stage's scene contains a scene graph of nodes

深圳）

The Chinese University of Hong Kong, Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

- **Step 1**: Extend javafx.application.Application and override start().

```java
package application;
import javafx.application.Application;
import javafx.stage.Stage;
public class Hello_World extends Application{

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


    }

}
```

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

- **Step 2**: Create a Button.

```
package application;
import javafx.application.Application;
importjavafx.scene.control.Button;
import javafx.stage.Stage;
public class Hello_World extends Application{

    @Override
    public void start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub
        Buttonbtn1=newButton("Say, Hello World");


    }


}
```

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

- **Step 3**: Create a layout and add a button to it.

```java
package application;
import javafx.application.Application;
import javafx.scene.control.Button;
import javafx.stage.Stage;
import javafx.scene.layout.StackPane;
public class Hello_World extends Application{

    @Override
    public void start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub
        Button btn1=new Button("Say, Hello World");
        StackPane root=new StackPane();
        root.getChildren().add(btn1);


    }


}
```

香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

- **Step 4**: Create a Scene.

```java
package application;
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;
import javafx.scene.layout.StackPane;
public class Hello_World extends Application{

    @Override
    public void start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub
        Button btn1=new Button("Say, Hello World");
        StackPane root=new StackPane();
        root.getChildren().add(btn1);
        Scene scene=new Scene(root);
    }

}
```

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

- **Step 5**: Prepare the Stage.

```java
package application;
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;
import javafx.scene.layout.StackPane;
public class Hello_World extends Application{

    @Override
    public void start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub
        Button btn1=new Button("Say, Hello World");
        StackPane root=new StackPane();
```

```java
        root.getChildren().add(btn1);
        Scene scene=new Scene(root);
        primaryStage.setScene(scene);
        primaryStage.setTitle("First JavaFX Application");
        primaryStage.show();
    }

}
```

香港中文大學 (深圳)
The Chinese University of Hong Kong, Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

- **Step 6**: Create an event for the button.

```java
package application;
import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;
import javafx.scene.layout.StackPane;
public class Hello_World extends Application{

    @Override
    publicvoid start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub
        Button btn1=new Button("Say, Hello World");
        btn1.setOnAction(new EventHandler<ActionEvent>() {
```
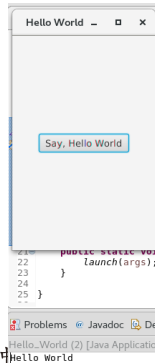
```java
    @Override
    publicvoid handle(ActionEvent arg0) {
        // TODO Auto-generated method stub
        System.out.println("hello world");
    }
});
StackPane root=new StackPane();
root.getChildren().add(btn1);
Scene scene=new Scene(root,600,400);
primaryStage.setScene(scene);
primaryStage.setTitle("First JavaFX Application");
primaryStage.show();
}

}
```

Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

```java
public class Hello_World extends Application{

    @Override
    public void start(Stage primaryStage) throws Exception {
        // TODO Auto-generated method stub
        Button btn1=new Button("Say, Hello World");
        btn1.setOnAction(new EventHandler<ActionEvent>() {

            @Override
            public void handle(ActionEvent arg0) {
                // TODO Auto-generated method stub
                System.out.println("hello world");
            }
        });
```

```java
        StackPane root=new StackPane();
        root.getChildren().add(btn1);
        Scene scene=new Scene(root,600,400);
        primaryStage.setTitle("First JavaFX Application");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    publicstaticvoid main (String[] args)
    {
        launch(args);
    }
}
```
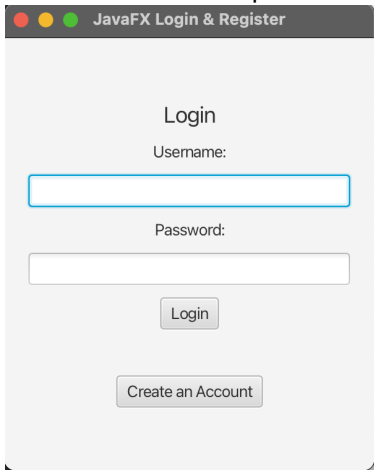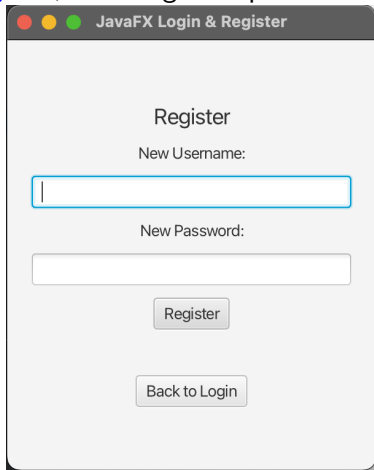

The Chinese University of Hong Kong, Shenzhen

JavaFX Application Example

A JavaFX application that prints hello world on the console by clicking the button.

Please check another example of JavaFXLoginRegister, involving multiple scenes.

Question and Answering (Q&A)