Foundations of Programming 2: JavaFX Basics

FoP 2 Teaching Team, Faculty of Computer Science, Universitas Indonesia
Correspondence: Fariz Darari (fariz@cs.ui.ac.id)

Why?

Not everything has to be in command-line interfaces (CLIs).

**Non-CS people (and GUI-minded CS people)
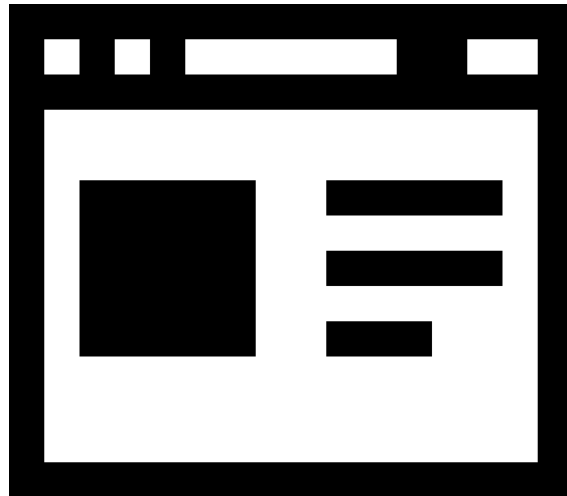would appreciate GUIs a lot!**

Why?

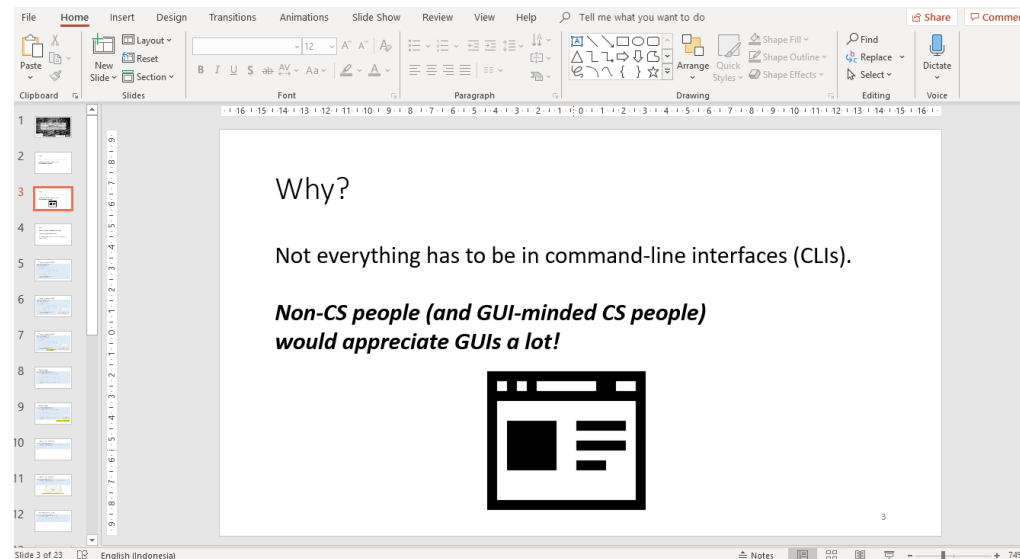Not everything has to be in command-line interfaces (CLIs).

***Non-CS people (and GUI-minded CS people)
would appreciate GUIs a lot!***

Why?

Not everything has to be in command-line interfaces (CLIs).

***Non-CS people (and GUI-minded CS people)
would appreciate GUIs a lot!***

Intro

JavaFX is a library for creating GUIs in Java.

OOPs are heavily utilized in JavaFX.

A JavaFX application can run both on a desktop and a Web browser.

The very basics of JavaFX

```java
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;

public class GUI001 extends Application {
        @Override
        public void start(Stage stg) throws Exception {
                Button button = new Button("Readyyy!");
                Scene scene = new Scene(button, 250, 100);
                stg.setTitle("GUI 001");
                stg.setScene(scene);
                stg.show();
        }

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

}
```
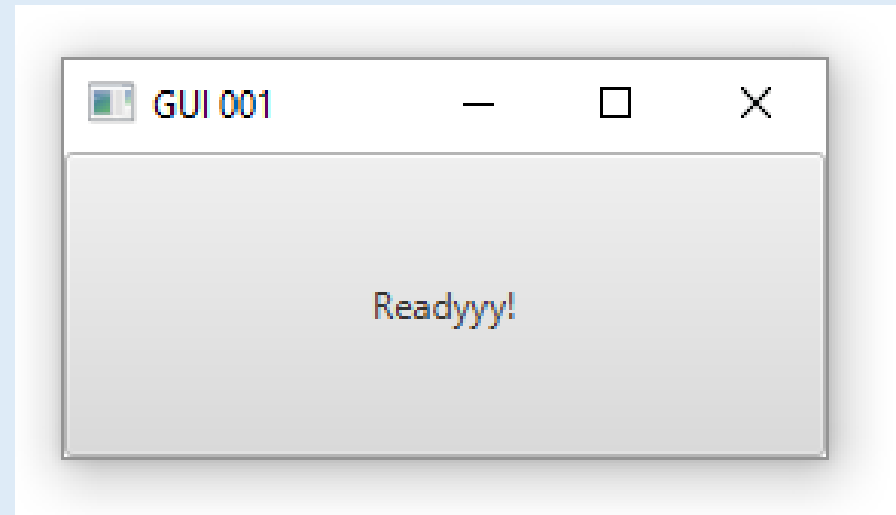
The very basics of JavaFX

```java
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;

public class GUI001 extends Application {
    @Override
    public void start(Stage stg) throws Exception {
        Button button = new Button("Readyyy!");
        Scene scene = new Scene(button, 250, 100);
        stg.setTitle("GUI 001");
        stg.setScene(scene);
        stg.show();
    }

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

The very basics of JavaFX

```java
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;

public class GUI001 extends Application {
    @Override
    public void start(Stage stg) throws Exception {
        Button button = new Button("Readyyy!");
        Scene scene = new Scene(button, 250, 100);
        stg.setTitle("GUI 001");
        stg.setScene(scene);
        stg.show();
    }

    public static void main(String[] args) {
     Application.launch(args);
    }
}
```
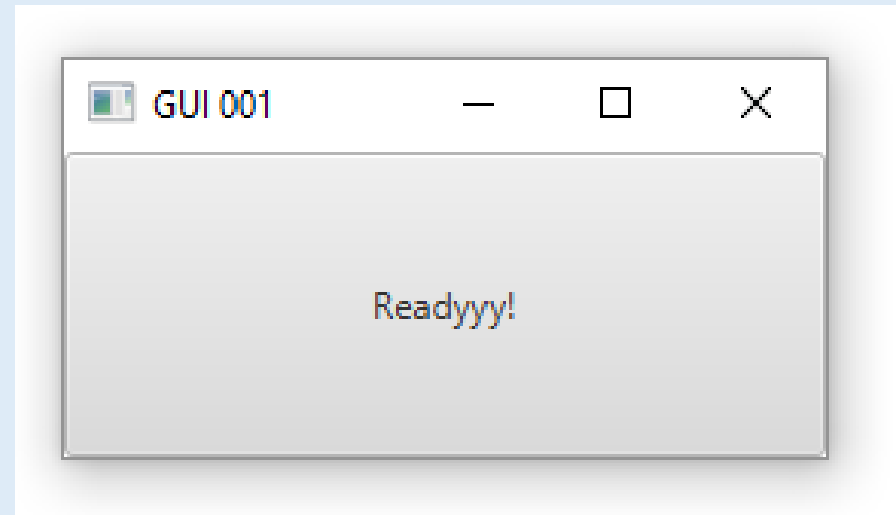
Add stg.setResizable(false); and what would happen?

Multiple stages

```
// ... other code parts follow previous code
public void start(Stage stg1) throws Exception {
        stg1.setTitle("GUI 002");
        stg1.setScene(new Scene(new Button("Gooo!"), 200,100));
        stg1.show();

        Stage stg2 = new Stage();
        stg2.setTitle("GUI 002");
        stg2.setScene(new Scene(new Button("Steadyyy!"), 200,100));
        stg2.show();

        Stage stg3 = new Stage();
        stg3.setTitle("GUI 002");
        stg3.setScene(new Scene(new Button("Readyyy!"), 200,100));
        stg3.show();
}
```
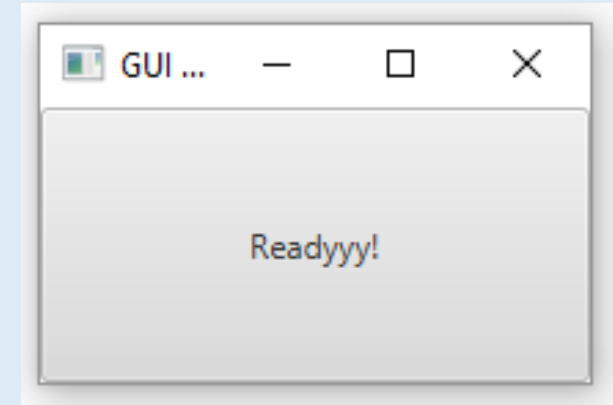
Multiple stages

```
// ... other code parts follow previous code
public void start(Stage stg1) throws Exception {
        stg1.setTitle("GUI 002");
        stg1.setScene(new Scene(new Button("Gooo!"), 200,100));
        stg1.show();

        Stage stg2 = new Stage();
        stg2.setTitle("GUI 002");
        stg2.setScene(new Scene(new Button("Steadyyy!"), 200,100));
        stg2.show();

        Stage stg3 = new Stage();
        stg3.setTitle("GUI 002");
        stg3.setScene(new Scene(new Button("Readyyy!"), 200,100));
        stg3.show();
}
```
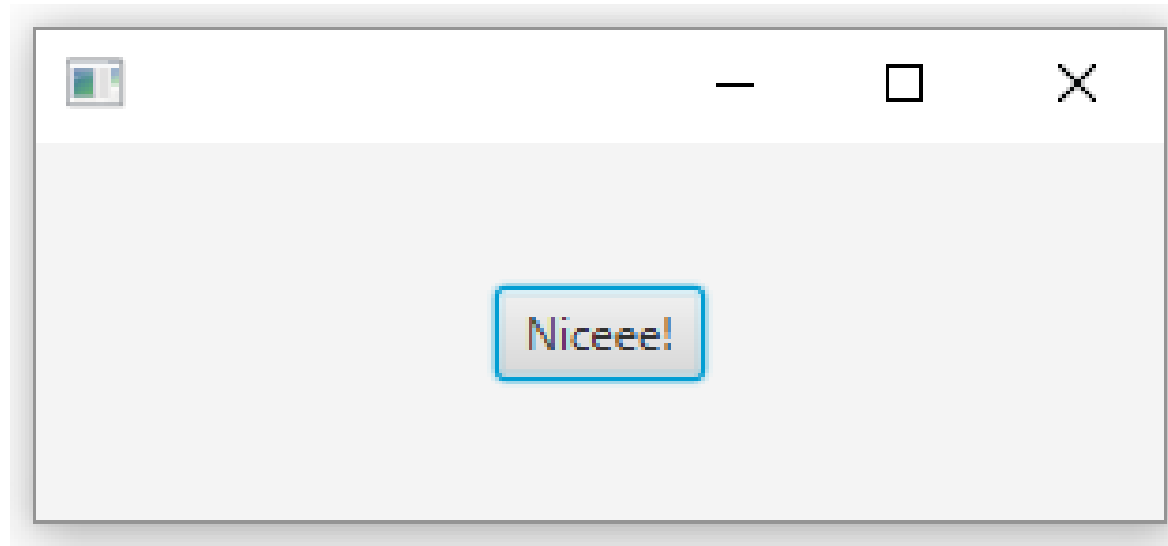

Close the window, you'll see the Steadyyy! stage, close the window again, you'll see the Gooo! stage.

A pane is like a container

```
public void start(Stage stg1) throws Exception {
        StackPane pn = new StackPane();
        pn.getChildren().add(new Button("Niceee!")); // because Nice! would sound rude
        Scene scn = new Scene(pn, 300, 100);
        stg1.setScene(scn);
        stg1.show();
}
```

A pane is like a container

```
public void start(Stage stg1) throws Exception {
        StackPane pn = new StackPane();
        pn.getChildren().add(new Button("Niceee!")); // because Nice! would sound rude
        Scene scn = new Scene(pn, 300, 100);
        stg1.setScene(scn);
        stg1.show();
}
```


Note that the button now doesn't occupy the whole scene, thanks to Pane!

Stacked square and button
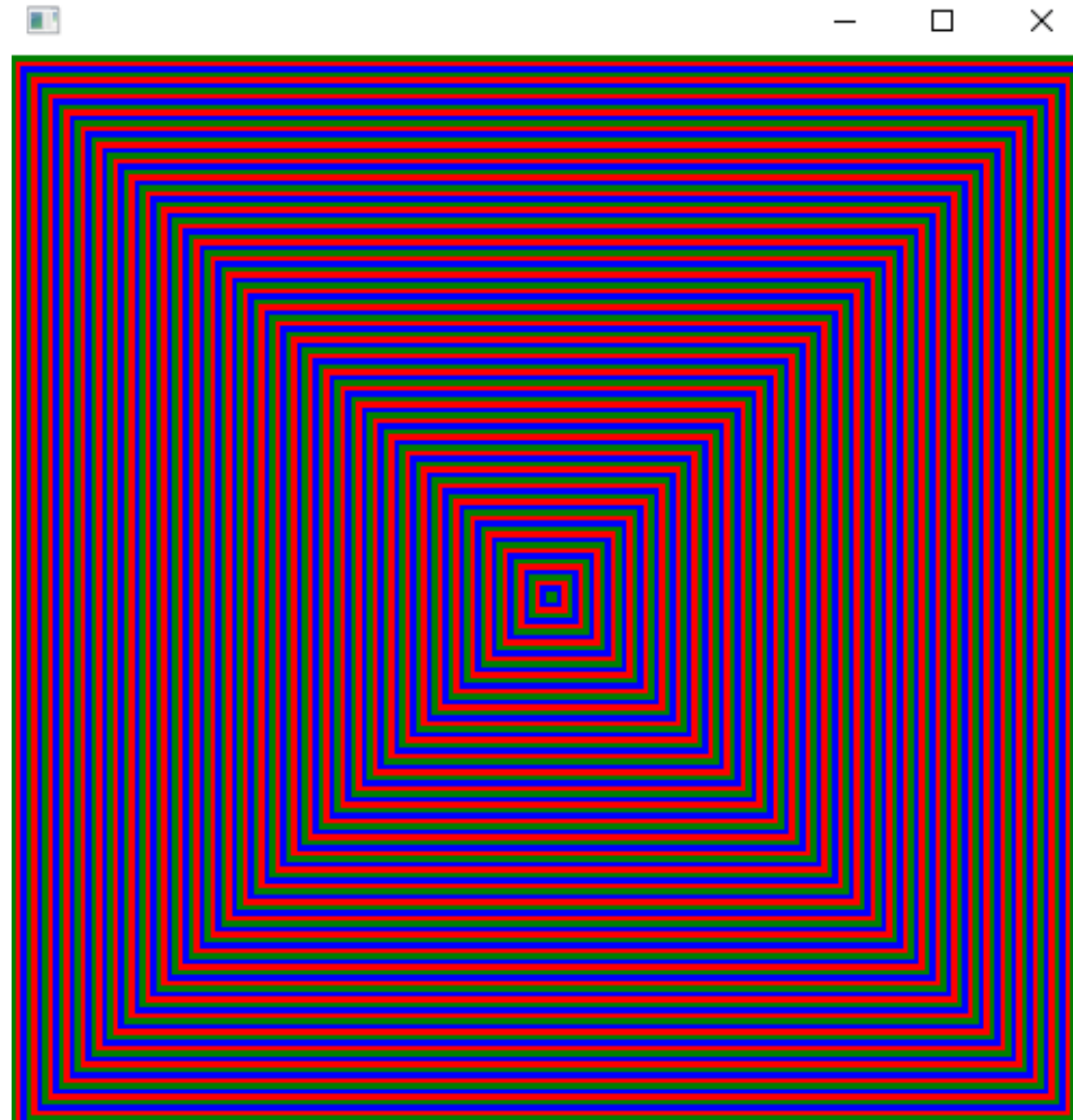
```
public void start(Stage stg1) throws Exception {
        Rectangle rect = new Rectangle(100, 100, 180, 140);
        rect.setFill(Color.BLUE);

        StackPane pn = new StackPane(rect, new Button("Niceee!"));
        Scene scn = new Scene(pn, 500, 200);
        stg1.setScene(scn);
        stg1.show();
}
```

Stacked square and button

```java
public void start(Stage stg1) throws Exception {
        Rectangle rect = new Rectangle(100, 100, 180, 140);
        rect.setFill(Color.BLUE);

        StackPane pn = new StackPane(rect, new Button("Niceee!"));
        Scene scn = new Scene(pn, 500, 200);
        stg1.setScene(scn);
        stg1.show();
}
```

Quiztime: Java art

Quiztime: Java art

```java
public void start(Stage stg1) throws Exception {
    StackPane pn = new StackPane();
    for(int i = 100; i > 0; i--) {
        Rectangle rect = new Rectangle(0,0,5*i,5*i);
        switch(i % 3) {
            case 0:
                rect.setFill(Color.RED); break;
            case 1:
                rect.setFill(Color.GREEN); break;
            case 2:
                rect.setFill(Color.BLUE); break;
        }
        pn.getChildren().add(rect);
    }

    Scene scn = new Scene(pn, 500, 500);
    stg1.setScene(scn);
    stg1.show();
}
```
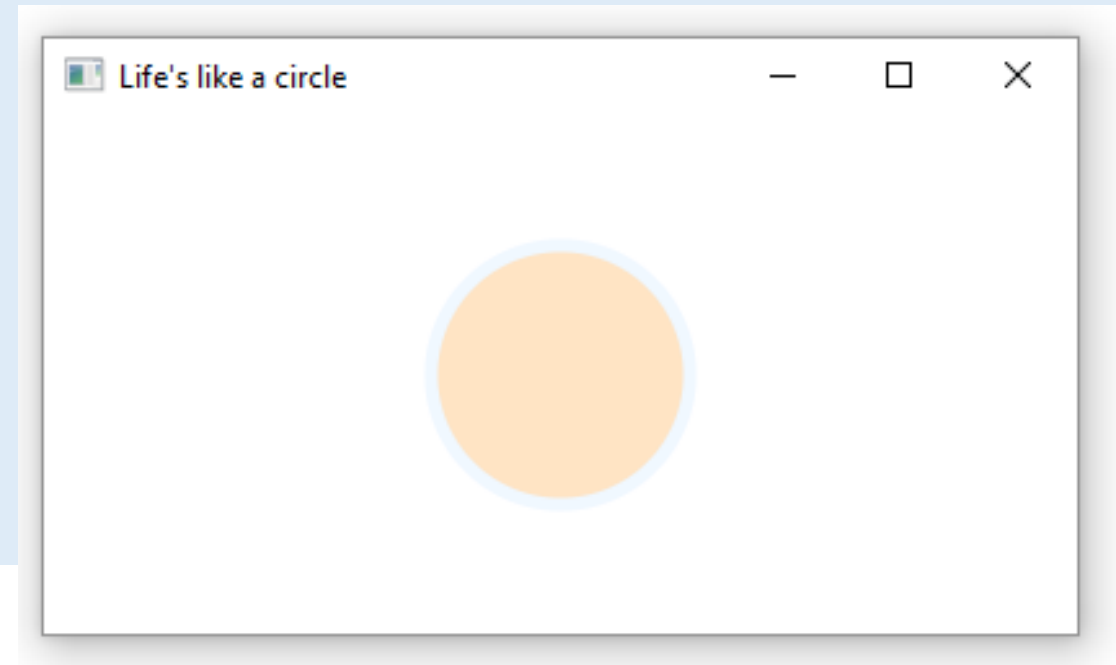
Circle

```java
public void start(Stage stg1) throws Exception {
        Circle c = new Circle();
        c.setCenterX(200);
        c.setCenterY(100);
        c.setRadius(50);
        c.setStroke(Color.ALICEBLUE);
        c.setStrokeWidth(5);
        c.setFill(Color.BISQUE);

        Pane pn = new Pane();
        pn.getChildren().add(c);
        Scene scn = new Scene(pn, 400, 200);
        stg1.setTitle("Life's like a circle");
        stg1.setScene(scn);
        stg1.show();
}
```
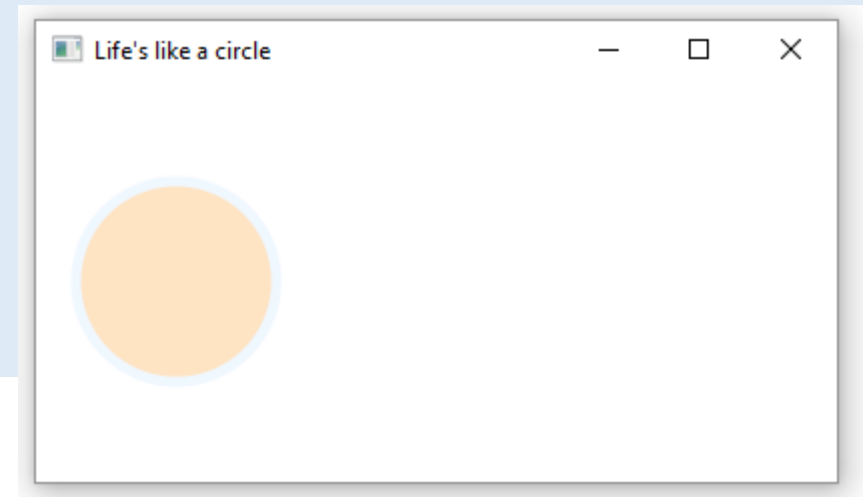
Circle

```java
public void start(Stage stg1) throws Exception {
        Circle c = new Circle();
        c.setCenterX(200);
        c.setCenterY(100);
        c.setRadius(50);
        c.setStroke(Color.ALICEBLUE);
        c.setStrokeWidth(5);
        c.setFill(Color.BISQUE);

        Pane pn = new Pane();
        pn.getChildren().add(c);
        Scene scn = new Scene(pn, 400, 200);
        stg1.setTitle("Life's like a circle");
        stg1.setScene(scn);
        stg1.show();
}
```

Circle

```java
public void start(Stage stg1) throws Exception {
        Circle c = new Circle();
        c.setCenterX(70);
        c.setCenterY(100);
        c.setRadius(50);
        c.setStroke(Color.ALICEBLUE);
        c.setStrokeWidth(5);
        c.setFill(Color.BISQUE);

        Pane pn = new Pane();
        pn.getChildren().add(c);
        Scene scn = new Scene(pn, 400, 200);
        stg1.setTitle("Life's like a circle");
        stg1.setScene(scn);
        stg1.show();
}
```
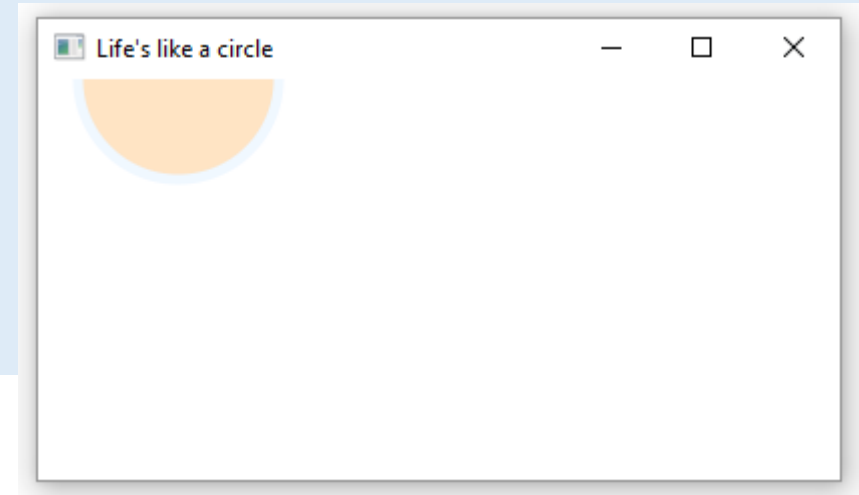
Circle

```java
public void start(Stage stg1) throws Exception {
        Circle c = new Circle();
        c.setCenterX(70);
        c.setCenterY(0);
        c.setRadius(50);
        c.setStroke(Color.ALICEBLUE);
        c.setStrokeWidth(5);
        c.setFill(Color.BISQUE);

        Pane pn = new Pane();
        pn.getChildren().add(c);
        Scene scn = new Scene(pn, 400, 200);
        stg1.setTitle("Life's like a circle");
        stg1.setScene(scn);
        stg1.show();
}
```
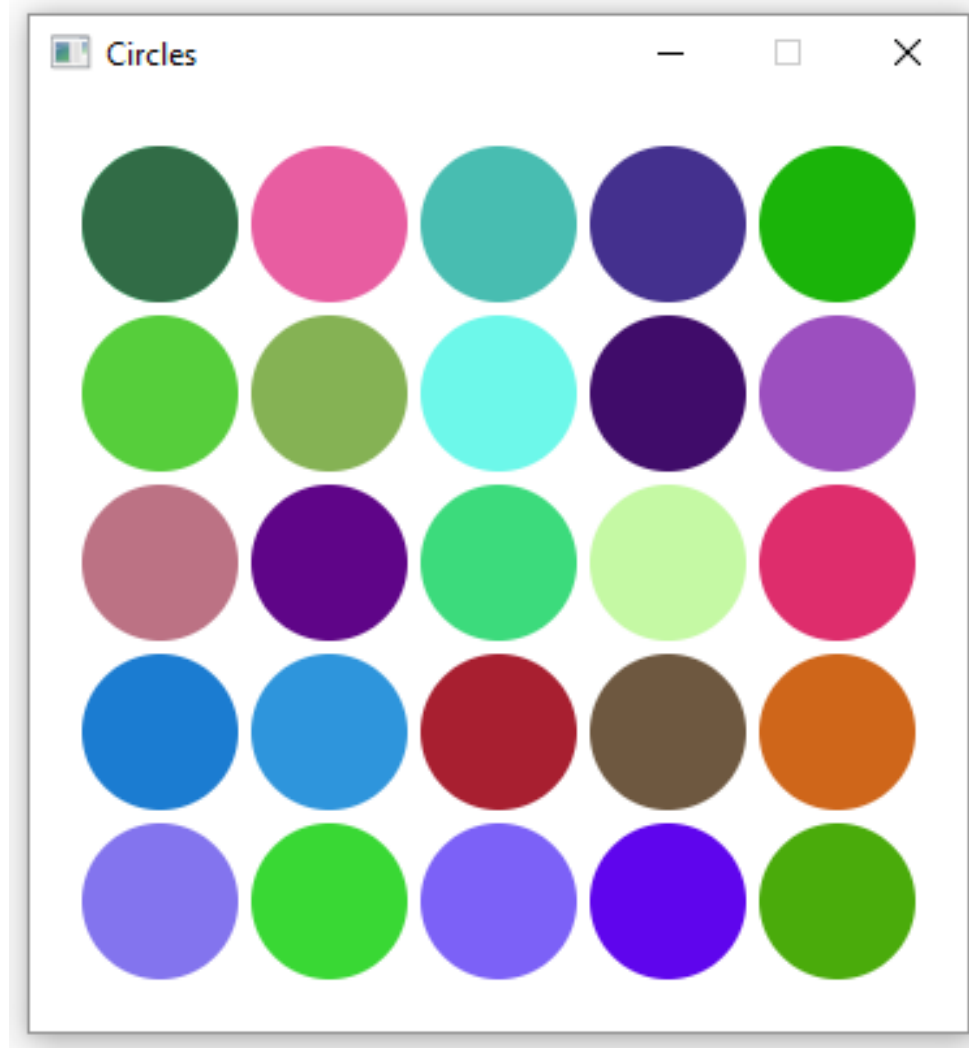
Quiztime: Loops of circles

Solution

```java
public Circle createCircle(int X, int Y) {
        Random r = new Random();
        Circle c = new Circle();
        c.setCenterX(X);
        c.setCenterY(Y);
        c.setRadius(30);
        c.setFill(Color.rgb(r.nextInt(256), r.nextInt(256), r.nextInt(256)));
        return c;
}

// ... cont
```

Solution

```
// ... cont

@Override
public void start(Stage stg1) throws Exception {
        Pane pn = new Pane();
        for(int i = 0; i < 5; i++)
                for(int j = 0; j < 5; j++)
                        pn.getChildren().add(createCircle(50 + i*65,50 + j*65));
        Scene scn = new Scene(pn, 360, 360);
        stg1.setTitle("Circles");
        stg1.setScene(scn);
        stg1.setResizable(false);
        stg1.show();
}
```
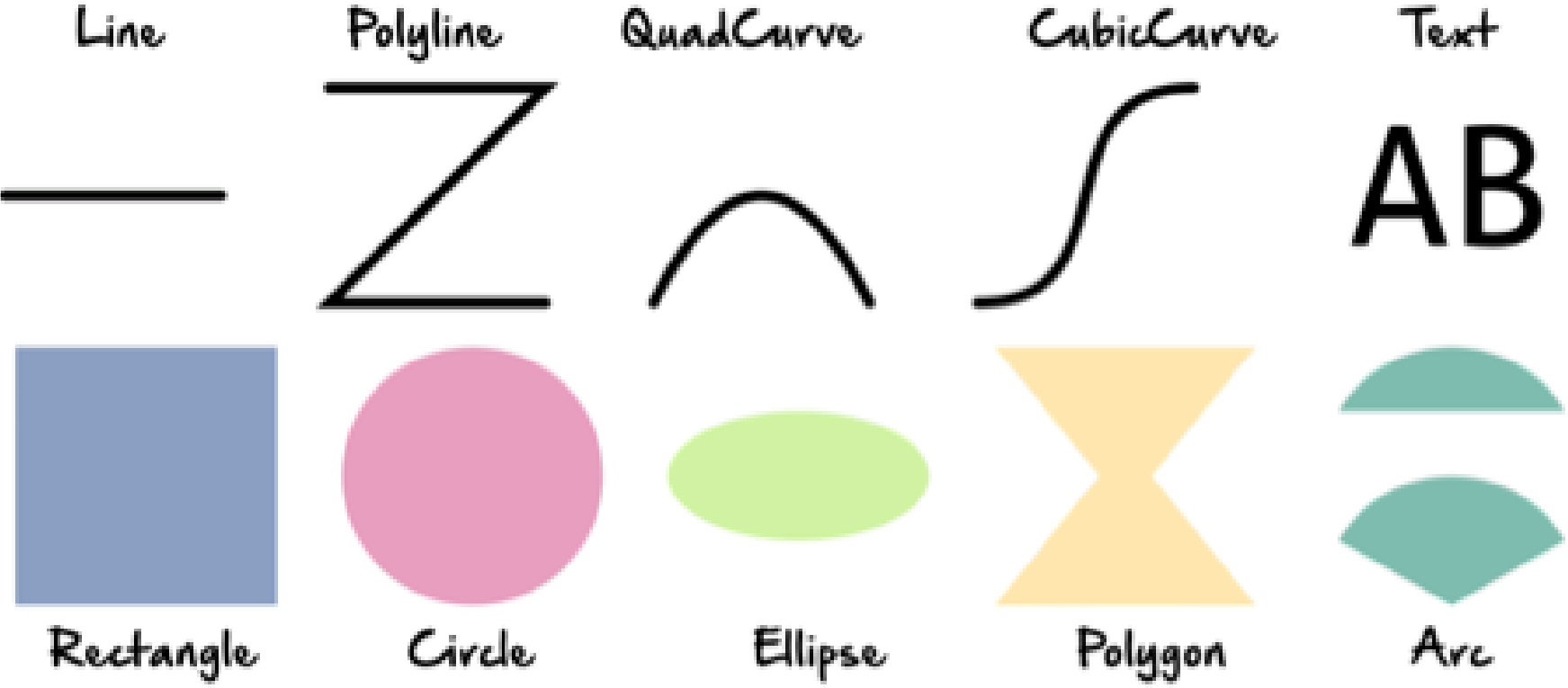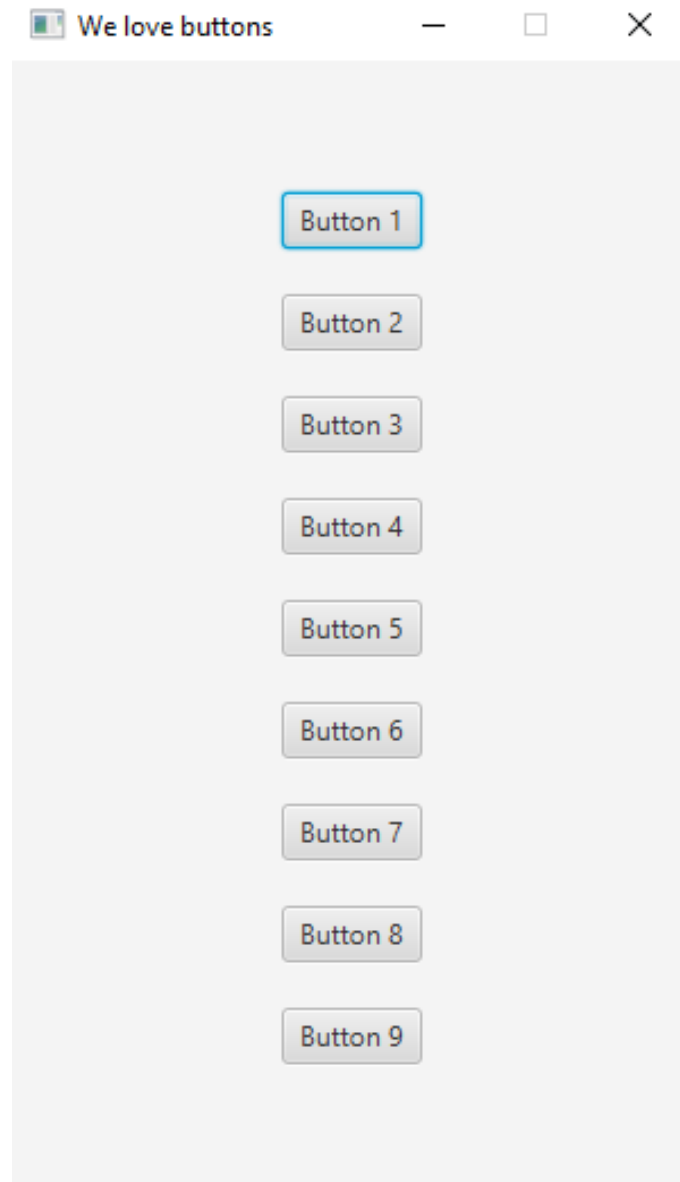
JavaFX provides many other shapes

Image copyright: https://dzone.com/refcardz/javafx-8-1?chapter=6

Quiztime: We love buttons

Solution

```java
@Override
public void start(Stage stg1) throws Exception {
    VBox pn = new VBox();
    pn.setAlignment(Pos.CENTER);
    pn.setSpacing(20);
    for(int i = 1; i <= 9; i++)
        pn.getChildren().add(new Button("Button " + i));
    Scene scn = new Scene(pn, 300, 500);
    stg1.setTitle("We love buttons");
    stg1.setScene(scn);
    stg1.setResizable(false);
    stg1.show();
}
```

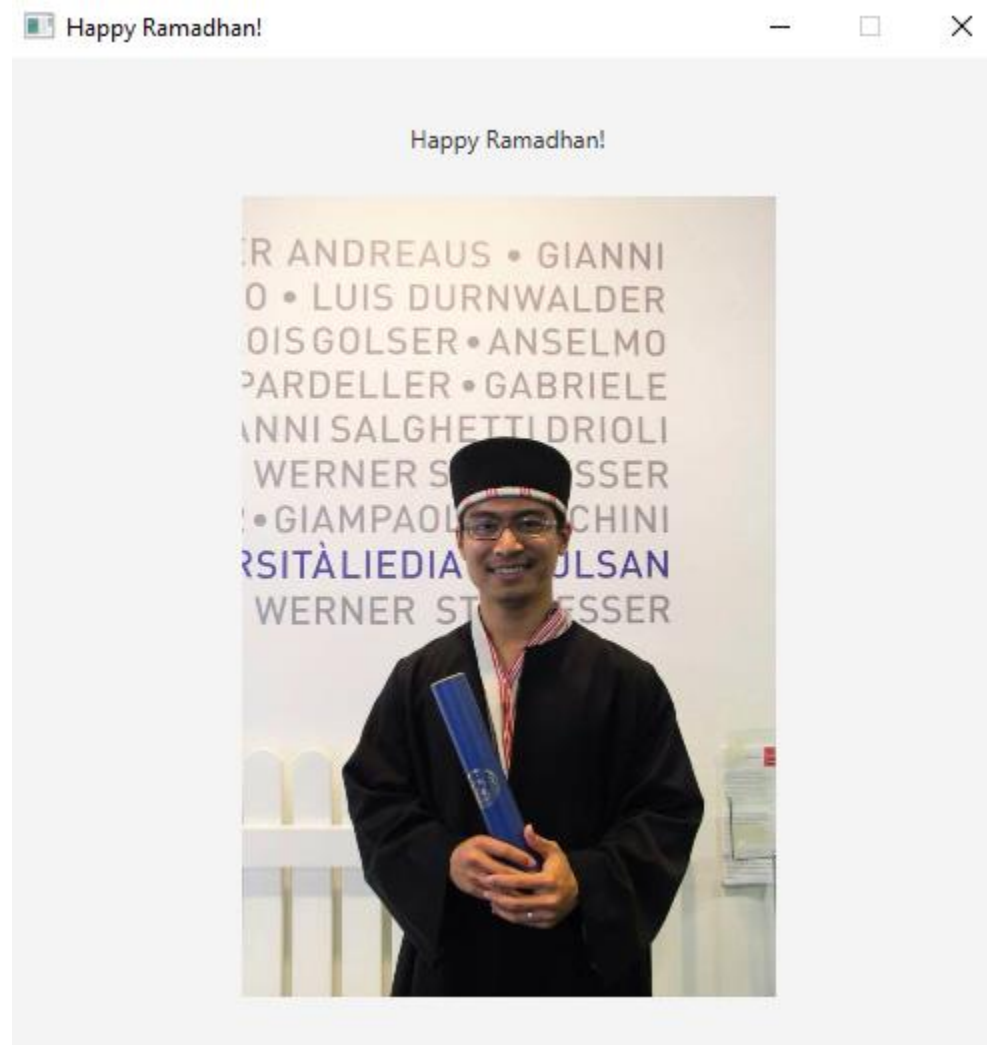# Quiztime: We love horizontal buttons

Solution

```java
@Override
    public void start(Stage stg1) throws Exception {
        HBox pn = new HBox();
        pn.setAlignment(Pos.CENTER);
        pn.setSpacing(20);
        for(int i = 1; i <= 9; i++)
            pn.getChildren().add(new Button("Button " + i));
        Scene scn = new Scene(pn, 800, 200);
        stg1.setTitle("We love buttons");
        stg1.setScene(scn);
        stg1.setResizable(false);
        stg1.show();
    }
```
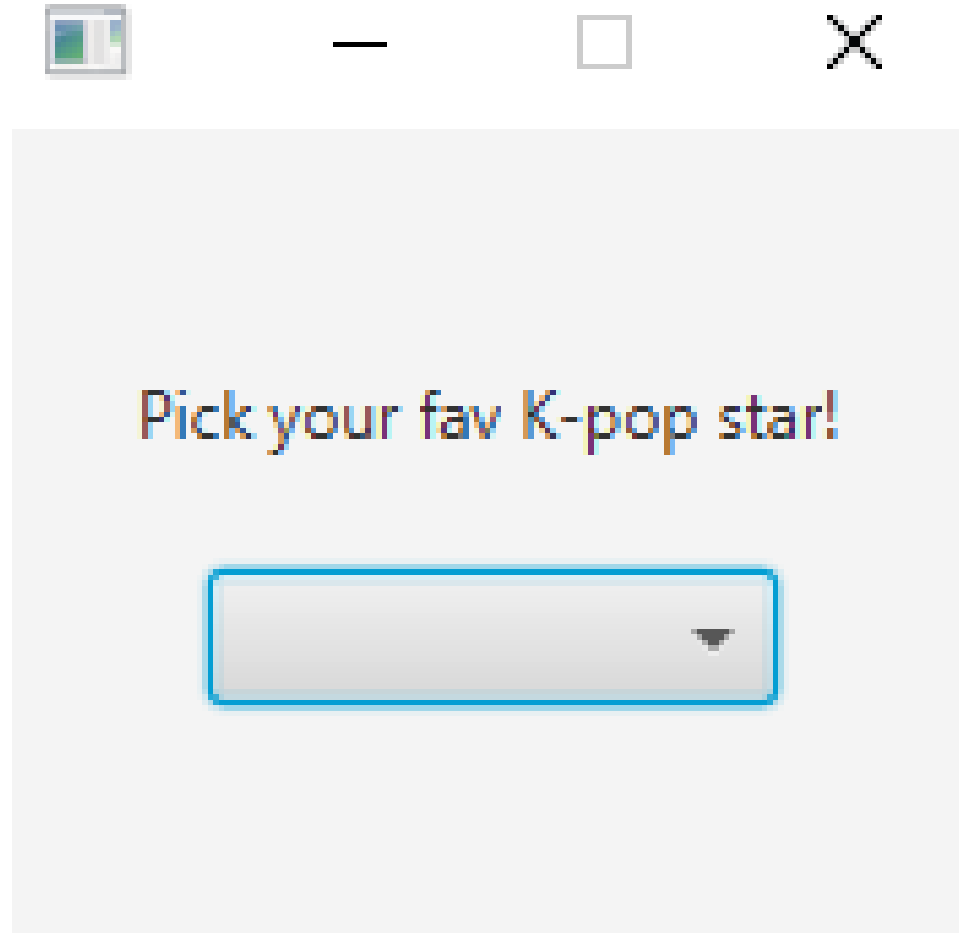
JavaFX Labels and Images

Solution

```java
public void start(Stage stg1) throws Exception {
        VBox pn = new VBox();
        pn.setAlignment(Pos.CENTER);
        pn.setSpacing(20);
        pn.getChildren().add(new Label("Happy Ramadhan!"));
        FileInputStream fis = new FileInputStream("pics/ramadhan.jpg");
        Image img = new Image(fis);
        ImageView iv = new ImageView(img);
        iv.setFitHeight(400);
        iv.setPreserveRatio(true);
        pn.getChildren().add(iv);
        Scene scn = new Scene(pn, 500, 500);
        stg1.setTitle("Happy Ramadhan!");
        stg1.setScene(scn);
        stg1.setResizable(false);
        stg1.show();
}
```

Quiztime: Create your own Happy Ramadhan (or Merry X-Mas, or any greetings!
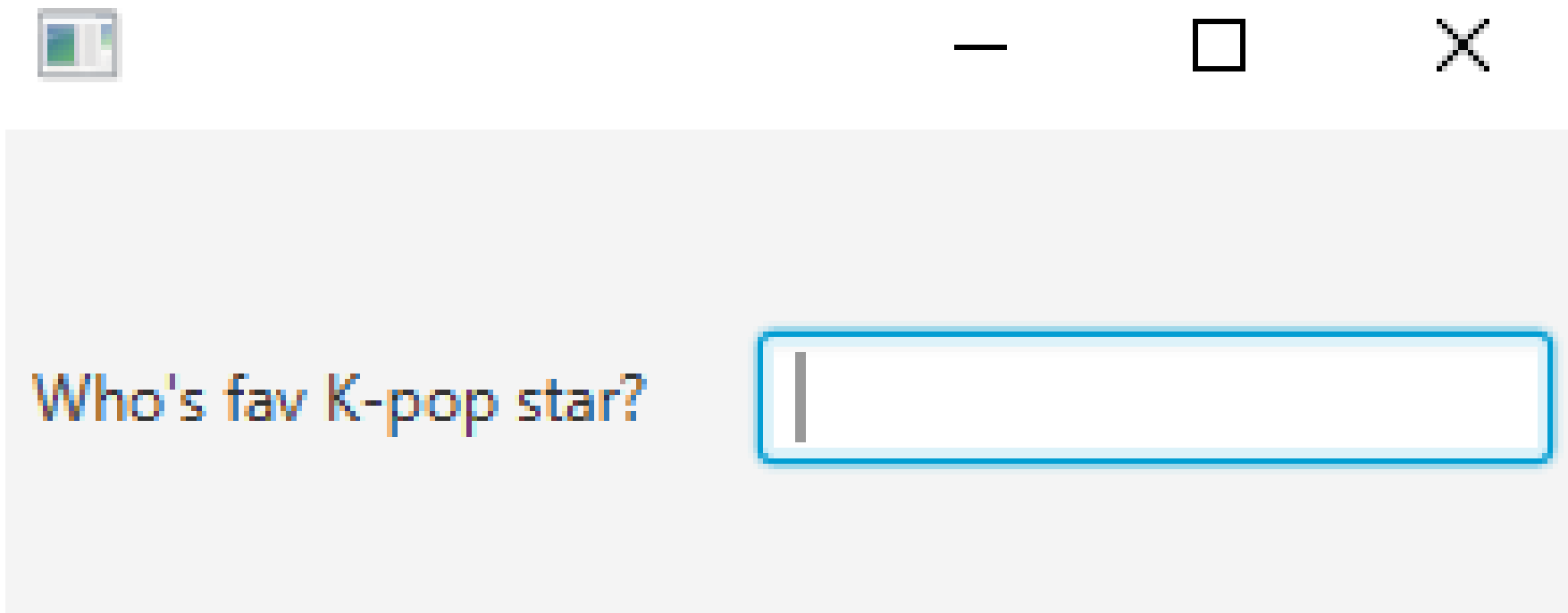
ChoiceBox

ChoiceBox: Code

```java
    public void start(Stage stg1) throws Exception {
        VBox pn = new VBox();
        pn.setAlignment(Pos.CENTER);
        pn.setSpacing(20);
        pn.getChildren().add(new Label("Pick your fav K-pop
star!"));
        String[] stars = {"Joo Ko-Wee", "Park Bo-Wow"};
        ChoiceBox cb = new
ChoiceBox(FXCollections.observableArrayList(stars));
        pn.getChildren().add(cb);
        Scene scn = new Scene(pn, 180, 150);
        stg1.setScene(scn);
        stg1.setResizable(false);
        stg1.show();
    }
```

TextField

Who's fav K-pop star? [_____]

TextField: Code

```java
public void start(Stage stg1) throws Exception {
        HBox pn = new HBox();
        pn.setAlignment(Pos.CENTER);
        pn.setSpacing(20);
        pn.getChildren().add(new Label("Who's fav K-pop star?"));
        TextField tf = new TextField();
        tf.setPrefWidth(150);
        pn.getChildren().add(tf);
        Scene scn = new Scene(pn, 300, 100);
        stg1.setScene(scn);
        stg1.show();
}
```

CheckBox

Who are your fav K-pop stars?

- ☐ Joo Ko-Wee
- ☐ Park Bo-Wow
- ☐ Sandiaga Yunho

CheckBox: Code

```
public void start(Stage stg1) throws Exception {
        VBox pn = new VBox();
        pn.setAlignment(Pos.CENTER);
        pn.setSpacing(20);
        pn.getChildren().add(new Label("Who are your fav K-pop stars?"));
        CheckBox cb1 = new CheckBox("Joo Ko-Wee");
        CheckBox cb2 = new CheckBox("Park Bo-Wow");
        CheckBox cb3 = new CheckBox("Sandiaga Yunho");
        pn.getChildren().add(cb1);
        pn.getChildren().add(cb2);
        pn.getChildren().add(cb3);
        Scene scn = new Scene(pn, 200, 200);
        stg1.setScene(scn);
        stg1.show();
}
```

THANK YOU

Inspired by:

Liang, Introduction to Java, 10th edition, Pearson.

https://www.geeksforgeeks.org/

Google