

in javafx i want to open a new fxml scene. but before that i want to close the previous fxml scene. what will be the code for it write me in java

volume\_up

1+

To open a new FXML scene and close the previous FXML scene in JavaFX, you can use the following code:

#### Java

```
// Get the current stage
Stage stage = (Stage) root.getScene().getWindow();

// Close the current scene
stage.close();

// Load the new FXML scene
Scene newScene = FXMLLoader.load(getClass().getResource("new_scene.fxml"));

// Set the new scene on the stage
stage.setScene(newScene);

// Show the new stage
stage.show();
```

Use code with caution. [Learn more](#)  
content\_copy

This code will first get the current stage, which is the window that the current scene is displayed in. Then, it will close the current scene. Next, it will load the new FXML scene. Finally, it will set the new scene on the stage and show the stage.

Here is an example of how to use the code above:

#### Java

```
public class Main extends Application {

    private Stage primaryStage;

    @Override
    public void start(Stage primaryStage) {
        this.primaryStage = primaryStage;
```

```

// Load the initial scene
Scene scene = FXMLLoader.load(getClass().getResource("scene1.fxml"));

// Set the scene on the stage and show the stage
primaryStage.setScene(scene);
primaryStage.show();

// Add an action listener to the button to open the new scene
Button button = (Button) scene.lookup("#button");
button.setOnAction(event -> {
    // Close the current scene
    primaryStage.close();

    // Load the new FXML scene
    Scene newScene =
FXMLLoader.load(getClass().getResource("scene2.fxml"));

    // Set the new scene on the stage and show the stage
    primaryStage.setScene(newScene);
    primaryStage.show();
});
}

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

```

Use code with caution. [Learn more](#)  
content\_copy

When you run this code, you will see a window with a button. Clicking the button will open a new window with a different scene.

Sources

[info](#)

1. [www.java-forum.org/thema/java-lang-nullpointerexception-bei-javafx.186346/](http://www.java-forum.org/thema/java-lang-nullpointerexception-bei-javafx.186346/)
2. [github.com/RiemannSphere/Pixel-Art-Gallery](https://github.com/RiemannSphere/Pixel-Art-Gallery)

share

more\_vert



modify the given code in such way that whenever i will click on a button of current page than it will close the current page than take me to the new page : package

```
com.example.soccer_b; import javafx.event.ActionEvent;
import javafx.fxml.FXML; import javafx.fxml.FXMLLoader;
import javafx.scene.Scene; import javafx.stage.Stage;
import java.io.IOException; public class logincontrol {
    @FXML void forgetpassbutton(ActionEvent event) throws
    IOException { FXMLLoader fxmLoader = new
    FXMLLoader(HelloApplication.class.getResource("forgotp
    ass.fxml")); Scene scene = new Scene(fxmLoader.load());
    Stage stage = new Stage(); //stage.setTitle("Hello!");
    stage.setScene(scene); stage.show(); } @FXML void
    loginbutton(ActionEvent event) throws IOException {
    FXMLLoader fxmLoader = new
    FXMLLoader(HelloApplication.class.getResource("login.fx
    ml")); Scene scene = new Scene(fxmLoader.load());
    Stage stage = new Stage(); //stage.setTitle("Hello!");
    stage.setScene(scene); stage.show(); } @FXML void
    registerbutton(ActionEvent event) throws IOException {
    FXMLLoader fxmLoader = new
    FXMLLoader(HelloApplication.class.getResource("register
    .fxml")); Scene scene = new Scene(fxmLoader.load());
    Stage stage = new Stage(); //stage.setTitle("Hello!");
    stage.setScene(scene); stage.show(); } }
```

expand\_more

volume\_up

1  
2  
2  
2  
3  
1

To modify the given code in such a way that whenever you click on a button of the current page, it will close the current page and take you to the new page, you can do the following:

## Java

```
@FXML
void forgetpassbutton(ActionEvent event) throws IOException {
    // Get the current stage
    Stage stage = (Stage) event.getSource().getScene().getWindow();

    // Close the current stage
    stage.close();

    // Load the new FXML scene
    FXMLLoader fxmlLoader = new
FXMLLoader(HelloApplication.class.getResource("forgotpass.fxml"));
    Scene scene = new Scene(fxmlLoader.load());

    // Create a new stage
    Stage newStage = new Stage();

    // Set the new scene on the stage and show the stage
    newStage.setScene(scene);
    newStage.show();
}
```

```
@FXML
void loginbutton(ActionEvent event) throws IOException {
    // Get the current stage
    Stage stage = (Stage) event.getSource().getScene().getWindow();

    // Close the current stage
    stage.close();

    // Load the new FXML scene
    FXMLLoader fxmlLoader = new
FXMLLoader(HelloApplication.class.getResource("login.fxml"));
    Scene scene = new Scene(fxmlLoader.load());

    // Create a new stage
    Stage newStage = new Stage();

    // Set the new scene on the stage and show the stage
    newStage.setScene(scene);
    newStage.show();
}
```

```
@FXML
void registerbutton(ActionEvent event) throws IOException {
    // Get the current stage
```

```

Stage stage = (Stage) event.getSource().getScene().getWindow();

// Close the current stage
stage.close();

// Load the new FXML scene
FXMLLoader fxmLoader = new
FXMLLoader(HelloApplication.class.getResource("register.fxml"));
Scene scene = new Scene(fxmLoader.load());

// Create a new stage
Stage newStage = new Stage();

// Set the new scene on the stage and show the stage
newStage.setScene(scene);
newStage.show();
}

```

Use code with caution. [Learn more](#)  
content\_copy

This code is similar to the original code, but it adds a few steps:

1. It gets the current stage.
2. It closes the current stage.
3. It creates a new stage.
4. It sets the new scene on the new stage and shows the new stage.

By following these steps, you can ensure that the current page is closed before the new page is opened.

Here is an example of how to use the modified code:

## Java

```

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception {
        FXMLLoader fxmLoader = new
FXMLLoader(HelloApplication.class.getResource("login.fxml"));
        Scene scene = new Scene(fxmLoader.load());

        primaryStage.setScene(scene);
        primaryStage.show();

        // Add an action listener to the button to open the forgot password
page
        Button forgetPassButton = (Button) scene.lookup("#forgetpassbutton");
        forgetPassButton.setOnAction(event -> {
            // Get the current stage

```

```

        Stage stage = (Stage) event.getSource().getScene().getWindow();

        // Close the current stage
        stage.close();

        // Load the forgot password FXML scene
        FXMLLoader newFXMLLoader = new
FXMLLoader(HelloApplication.class.getResource("forgotpass.fxml"));
        Scene newScene = new Scene(newFXMLLoader.load());

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the new stage and show the new stage
        newStage.setScene(newScene);
        newStage.show();
    });
}

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

```

Use code with caution. [Learn more](#)  
content\_copy

When you run this code, you will see a window with a button. Clicking the button will open a new window with the forgot password page.

Sources

[info](#)

1. [github.com/UsernameBM/Frontend](https://github.com/UsernameBM/Frontend)
2. [github.com/broniuu/Java\\_project\\_2022](https://github.com/broniuu/Java_project_2022)
3. [stackoverflow.com/questions/71711311/javafx-and-gradle-what-is-the-correct-way-to-load-a-default-view](https://stackoverflow.com/questions/71711311/javafx-and-gradle-what-is-the-correct-way-to-load-a-default-view)

share  
more\_vert



for getscene() which class need to be imported here

volume\_up

To use the `getScene()` method, you need to import the `javafx.scene.Node` class.

## Java

```
import javafx.scene.Node;
```

Use code with caution. [Learn more](#)

content\_copy

The `getScene()` method is defined on the `Node` class, so you can call it on any `Node` object.

Here is an example of how to use the `getScene()` method:

## Java

```
Node node = ...; // Get a Node object
```

```
// Get the Scene that the Node is attached to  
Scene scene = node.getScene();
```

Use code with caution. [Learn more](#)

content\_copy

You can then use the `Scene` object to access the stage, window, and other scene properties.

share

more\_vert



i have already open a page. now from that page i want to go another page, but before that want to close the current page. now modify the given code according to that:

```
package com.example.soccer_b; import  
javafx.event.ActionEvent; import javafx.fxml.FXML; import  
javafx.fxml.FXMLLoader; import javafx.scene.Scene;  
import javafx.stage.Stage; import javafx.scene.Node;  
import java.io.IOException; import static  
javafx.application.Application.launch; public class  
logincontrol { @FXML void forgetpassbutton(ActionEvent  
event) throws IOException { // Get the current stage Stage  
stage = (Stage) event.getSource(). // Close the current
```

```

stage stage.close(); // Load the new FXML scene
FXMLLoader fxmLoader = new
FXMLLoader(HelloApplication.class.getResource("forgotp
ass.fxml")); Scene scene = new Scene(fxmLoader.load());
// Create a new stage Stage newStage = new Stage(); //
Set the new scene on the stage and show the stage
newStage.setScene(scene); newStage.show(); } @FXML
void loginbutton(ActionEvent event) throws IOException {
// Get the current stage Stage stage = (Stage)
event.getSource().getScene().getWindow(); // Close the
current stage stage.close(); // Load the new FXML scene
FXMLLoader fxmLoader = new
FXMLLoader(HelloApplication.class.getResource("login.fx
ml")); Scene scene = new Scene(fxmLoader.load()); //
Create a new stage Stage newStage = new Stage(); // Set
the new scene on the stage and show the stage
newStage.setScene(scene); newStage.show(); } @FXML
void registerbutton(ActionEvent event) throws IOException
{ // Get the current stage Stage stage = (Stage)
event.getSource().getScene().getWindow(); // Close the
current stage stage.close(); // Load the new FXML scene
FXMLLoader fxmLoader = new
FXMLLoader(HelloApplication.class.getResource("register
.fxml")); Scene scene = new Scene(fxmLoader.load()); //
Create a new stage Stage newStage = new Stage(); // Set
the new scene on the stage and show the stage
newStage.setScene(scene); newStage.show(); } public
static void main(String[] args) { launch(); } }

```

expand\_more

volume\_up



1  
2  
2

To modify the given code so that it closes the current page before opening the new page, you can simply add the following line to the beginning of each of the `forgetpassbutton()`, `loginbutton()`, and `registerbutton()` methods:

#### Java

```
((Stage) ((Node) event.getSource()).getScene().getWindow()).close();
```

Use code with caution. [Learn more](#)

content\_copy

This will get the current stage and close it.

Here is the modified code:

#### Java

```
public class logincontrol {

    @FXML
    void forgetpassbutton(ActionEvent event) throws IOException {
        // Get the current stage and close it
        ((Stage) ((Node) event.getSource()).getScene().getWindow()).close();

        // Load the new FXML scene
        FXMLLoader fxmlLoader = new
FXMLLoader(HelloApplication.class.getResource("forgotpass.fxml"));
        Scene scene = new Scene(fxmlLoader.load());

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(scene);
        newStage.show();
    }

    @FXML
    void loginbutton(ActionEvent event) throws IOException {
        // Get the current stage and close it
        ((Stage) ((Node) event.getSource()).getScene().getWindow()).close();

        // Load the new FXML scene
        FXMLLoader fxmlLoader = new
FXMLLoader(HelloApplication.class.getResource("login.fxml"));
```

```

        Scene scene = new Scene(fxmlLoader.load());

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(scene);
        newStage.show();
    }

    @FXML
    void registerbutton(ActionEvent event) throws IOException {
        // Get the current stage and close it
        ((Stage) ((Node) event.getSource()).getScene().getWindow()).close();

        // Load the new FXML scene
        FXMLLoader fxmlLoader = new
FXMLLoader(HelloApplication.class.getResource("register.fxml"));
        Scene scene = new Scene(fxmlLoader.load());

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(scene);
        newStage.show();
    }
}

```

Use code with caution. [Learn more](#)  
content\_copy

Now, when you click on any of the buttons, the current page will be closed before the new page is opened.

## Sources

[info](#)

1. [github.com/UsernameBM/Frontend](https://github.com/UsernameBM/Frontend)
2. [github.com/broniuu/Java\\_project\\_2022](https://github.com/broniuu/Java_project_2022)

share  
more\_vert



here i have gone to other page through using a function by "on action" option in scene builder. Now i want to do the

same job using an image's fxml id. write me the code in java

volume\_up

1  
2

To use an image's FXML ID to open a new FXML scene in Java, you can use the following code:

### Java

```
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.stage.Stage;

public class Controller {

    @FXML
    private ImageView imageView;

    @FXML
    public void openNewScene(ActionEvent event) throws Exception {
        // Get the FXML ID of the image
        String fxmlId = imageView.getId();

        // Load the new FXML scene
        FXMLLoader fxmlLoader = new FXMLLoader(getClass().getResource(fxmlId
+ ".fxml"));
        Parent root = fxmlLoader.load();

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(new Scene(root));
        newStage.show();
    }
}
```

Use code with caution. [Learn more](#)  
content\_copy

To use this code, you would first need to add an image to your FXML scene and give it an ID. Then, you would need to create a method called `openNewScene()` that takes an `ActionEvent` object as a parameter. In the `openNewScene()` method, you would get the FXML ID of the image and load the new FXML scene. Finally, you would create a new stage and set the new scene on the stage.

Here is an example of how to use the code:

Java

```
<ImageView fx:id="imageView" src="image.png" onMouseClicked="openNewScene" />
```

Use code with caution. [Learn more](#)

content\_copy

This code would add an image to the FXML scene with the ID `imageView`. When the user clicks on the image, the `openNewScene()` method would be called. The `openNewScene()` method would load the FXML scene `image.fxml` and create a new stage to display the scene.

You can modify the code to open a different FXML scene depending on which image is clicked. For example, you could have multiple images on the page, each with a different ID. Then, in the `openNewScene()` method, you could check the FXML ID of the image and load the appropriate FXML scene.

Sources

[info](#)

1. [www.quabr.com/61904626/i-faced-this-error-when-i-use-javafx-for-my-desktop-application-java-lang-nullpo](http://www.quabr.com/61904626/i-faced-this-error-when-i-use-javafx-for-my-desktop-application-java-lang-nullpo)
2. [github.com/alecarb10/Progetto-B20](https://github.com/alecarb10/Progetto-B20)

subject to license (BSD - 2 - Clause)

share

more\_vert



## how to create this opennewscene method in scenebuilder

volume\_up

To create the `openNewScene()` method in Scene Builder, follow these steps:

1. Select the image that you want to use to open the new scene.
2. In the Properties panel, click the Events tab.
3. Click the Add button and select Mouse Clicked from the list of events.
4. In the Action field, type `openNewScene()`.
5. Click the OK button.

This will create a new method called `openNewScene()` in the controller class for your scene. You can then add the code for the method to the controller class.

Here is an example of how to add the code for the `openNewScene()` method to the controller class:

#### Java

```
public class Controller {

    @FXML
    private ImageView imageView;

    public void openNewScene() throws Exception {
        // Get the FXML ID of the image
        String fxmId = imageView.getId();

        // Load the new FXML scene
        FXMLLoader fxmLoader = new FXMLLoader(getClass().getResource(fxmId
+ ".fxml"));
        Parent root = fxmLoader.load();

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(new Scene(root));
        newStage.show();
    }
}
```

Use code with caution. [Learn more](#)  
content\_copy

Once you have added the code for the `openNewScene()` method to the controller class, you can run the application and click on the image to open the new scene.

share  
more\_vert



when i will click on a image it will direct invoke the method to go another page. how to do that. write me the code

volume\_up

1  
2

To directly invoke the method to go to another page when you click on an image, you can use the following code:

### Java

```
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.stage.Stage;

public class Controller {

    @FXML
    private ImageView imageView;

    @FXML
    public void openNewScene(ActionEvent event) throws Exception {
        // Get the FXML ID of the image
        String fxmlId = imageView.getId();

        // Load the new FXML scene
        FXMLLoader fxmlLoader = new FXMLLoader(getClass().getResource(fxmlId
+ ".fxml"));
        Parent root = fxmlLoader.load();

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(new Scene(root));
        newStage.show();
    }
}
```

Use code with caution. [Learn more](#)  
content\_copy

To use this code, you would first need to add an image to your FXML scene and give it an ID. Then, you would need to set the `onMouseClicked` property of the image to `openNewScene()`. This will cause the `openNewScene()` method to be called when the user clicks on the image.

Here is an example of how to use the code:

#### XML

```
<ImageView fx:id="imageView" src="image.png" onMouseClicked="openNewScene" />
```

Use code with caution. [Learn more](#)

content\_copy

When the user clicks on the image, the `openNewScene()` method will be called. The `openNewScene()` method will load the FXML scene `image.fxml` and create a new stage to display the scene.

You can modify the code to open a different FXML scene depending on which image is clicked. For example, you could have multiple images on the page, each with a different ID. Then, in the `openNewScene()` method, you could check the FXML ID of the image and load the appropriate FXML scene.

#### Sources

[info](#)

1. [www.quabr.com/61904626/i-faced-this-error-when-i-use-javafx-for-my-desktop-application-java-lang-nullpo](http://www.quabr.com/61904626/i-faced-this-error-when-i-use-javafx-for-my-desktop-application-java-lang-nullpo)
2. [github.com/alecarb10/Progetto-B20](https://github.com/alecarb10/Progetto-B20)

subject to license (BSD - 2 - Clause)

share

more\_vert



in a fxml file there is two action happening by a button and a image. whenever any of them is clicked than same method is called in its controller class. how to handle that write me the code

edit

View other drafts

volume\_up



To handle the situation where two actions happen (a button click and an image click) and call the same method in the controller class, you can use the following code:

Java

```
public class Controller {

    @FXML
    private Button button;

    @FXML
    private ImageView imageView;

    @FXML
    public void openNewScene() throws Exception {
        // Get the FXML ID of the button or image that was clicked
        String fxmlId = event.getSource().getId();

        // Load the new FXML scene
        FXMLLoader fxmlLoader = new FXMLLoader(getClass().getResource(fxmlId
+ ".fxml"));
        Parent root = fxmlLoader.load();

        // Create a new stage
        Stage newStage = new Stage();

        // Set the new scene on the stage and show the stage
        newStage.setScene(new Scene(root));
        newStage.show();
    }
}
```

Use code with caution. [Learn more](#)  
content\_copy

In this code, the `openNewScene()` method takes an `ActionEvent` object as a parameter. This object contains information about the event that triggered the method to be called. You can use the `getSource()` method on the `ActionEvent` object to get the button or image that was clicked.

You can then use the FXML ID of the button or image to load the appropriate FXML scene. For example, if the button was clicked, you would load the FXML scene `button.fxml`. If the image was clicked, you would load the FXML scene `image.fxml`.

Here is an example of how to use the code:



## XML

```
<Button fx:id="button" onAction="openNewScene" text="Open New Scene" />
<ImageView fx:id="imageView" src="image.png" onMouseClicked="openNewScene" />
```

Use code with caution. [Learn more](#)

content\_copy

When the user clicks on the button or the image, the `openNewScene()` method will be called. The `openNewScene()` method will load the appropriate FXML scene and create a new stage to display the scene.

tuneshare

more\_vert