Sources:

CISC190 Java Programming

A white background with black text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A close-up of a white background

AI-generated content may be incorrect.

A close-up of a computer screen

AI-generated content may be incorrect.

A white background with black text

AI-generated content may be incorrect.

A close-up of a white background

AI-generated content may be incorrect.

**-fx-text-fill** is like **color** property from normal CSS syntax

A screenshot of a computer

AI-generated content may be incorrect.

A close-up of a text

AI-generated content may be incorrect.

A screenshot of a computer code

AI-generated content may be incorrect.

A close-up of a white background

AI-generated content may be incorrect.

notice that JavaFX CSS doesn't directly support the display: none property like web CSS does. However, you can achieve a similar effect by manipulating the visibility and manageability of nodes within your application. \*

A screenshot of a computer

AI-generated content may be incorrect.

To make a node appear: .setVisible(true);

\*About z-index property from CSS:

JavaFX CSS does not directly support the z-index property like web CSS does. However, you can achieve similar effects by controlling the order of nodes within the parent container, you can use StackPane to manage these containers, and using methods to bring nodes to the front.

To learn more about StackPane, please visit:

**C:\Users\terro\OneDrive\Technology\Programming Languages\_CLI\Java\CISC\_190\_Java\_Programming\Basic\_Knowledge\Chapter12\_JavaFX\_GUI\_Programming\_Basic\_Controls\StackPane**

**Example Using toFront() Method**

You can use the toFront() method to bring a node to the front of its parent:

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.layout.StackPane;

import javafx.stage.Stage;

public class ZIndexExample extends Application {

@Override

public void start(Stage primaryStage) {

Button button1 = new Button("Button 1");

Button button2 = new Button("Button 2");

button2.setOnAction(e -> button1.toFront()); // Bring Button 1 to the front when Button 2 is clicked

StackPane stackPane = new StackPane(button1, button2);

Scene scene = new Scene(stackPane, 300, 200);

primaryStage.setScene(scene);

primaryStage.setTitle("Z-Index Example");

primaryStage.show();

}

public static void main(String[] args) {

launch(args);

}

}

In this example, clicking "Button 2" will bring "Button 1" to the front of the StackPane, effectively simulating a change in z-index.

**Controlling Node Order**

Another way to manage the stacking order is to control the order in which nodes are added to their parent container. Nodes added later will be on top of nodes added earlier.

\*About the position:fixed property from CSS:

JavaFX CSS does not support the position: fixed property as you might find in traditional web CSS. However, you can achieve similar effects using layout managers and controls available in JavaFX.

For example, you can use a Pane or AnchorPane to position nodes at fixed locations within the scene. Here's a simple example using AnchorPane:

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.layout.AnchorPane;

import javafx.stage.Stage;

public class FixedPositionExample extends Application {

@Override

public void start(Stage primaryStage) {

Button button = new Button("Fixed Position");

// Anchor the button at a fixed position

AnchorPane.setTopAnchor(button, 50.0);

AnchorPane.setLeftAnchor(button, 50.0);

AnchorPane anchorPane = new AnchorPane(button);

Scene scene = new Scene(anchorPane, 300, 200);

primaryStage.setScene(scene);

primaryStage.setTitle("Fixed Position Example");

primaryStage.show();

}

public static void main(String[] args) {

launch(args);

}

}

In this example, the button will always be positioned 50 pixels from the top and left edges of the AnchorPane, effectively mimicking a "fixed" position.

EX: Notice that the program example below may contains may syntax that you haven’t learned, it is for research purposes and practice only

**package** application;

**import** javafx.application.Application;

**import** javafx.stage.Stage;

**import** javafx.scene.Scene;

**import** javafx.scene.control.Label;

**import** javafx.scene.control.Button;

**import** javafx.scene.control.TextField;

**import** javafx.scene.image.Image;

**import** javafx.scene.image.ImageView;

**import** javafx.scene.control.ListView;

**import** javafx.scene.layout.HBox;

**import** javafx.scene.layout.VBox;

**import** javafx.scene.layout.GridPane;

**import** javafx.scene.layout.BorderPane;

**import** javafx.scene.layout.StackPane;

**import** javafx.scene.layout.AnchorPane;

**import** javafx.geometry.Insets;

**import** javafx.geometry.Pos;

**import** javafx.event.ActionEvent;

**import** javafx.event.EventHandler;

**public** **class** **Main** **extends** Application {

**public** **Label** label1;

**public** **Button** button1;

**public** **Button** button2;

**public** **Button** button3;

**public** **Button** button4;

**public** **Button** searchButton;

**public** **Button** button5;

**public** **Button** button6;

**public** **Button** button7;

**public** **Button** bottomButt1;

**public** **Button** bottomButt2;

**public** **Button** bottomButt3;

**public** **Label** aboutLabel;

**public** **TextField** textField1;

**public** **Label** searchLabel;

**public** **ListView**<String> listView1;

**public** **Image** image1;

**public** **ImageView** imageView1;

**public** **Image** image2;

**public** **ImageView** imageView2;

**public** **Image** image3;

**public** **ImageView** imageView3;

**public** **HBox** hbox;

**public** **VBox** CenterContent\_VBox;

**public** **HBox** CenterContent\_Content1\_HBox;

**public** **StackPane** CenterContent\_Content2\_StackPane;

**public** **AnchorPane** CenterContent\_Content2\_StackPane\_AnchorPane1;

//I can use GridPane when needed

//public GridPane gridPane1;

**public** **HBox** bottomHbox;

**public** **VBox** LeftVBox;

**public** **BorderPane** borderPane;

**public** **Scene** scene;

**public** **static** **void** **main**(**String**[] args) {

***launch***(args);

}

**@Override**

**public** **void** **start**(**Stage** primaryStage) {

primaryStage.**setTitle**(**"GUI Applicaiton"**);

label1 = **new** **Label**(**"Welcome"**);

button1 = **new** **Button**(**"🏠 HOME"**);

button1.**getStyleClass**().**add**(**"home\_button"**);

button2 = **new** **Button**(**"⏬ MENU"**);

button2.**getStyleClass**().**add**(**"menu\_button"**);

button3 = **new** **Button**(**"! ABOUT"**);

button3.**getStyleClass**().**add**(**"about\_button"**);

button4 = **new** **Button**(**"? HELP"**);

button4.**getStyleClass**().**add**(**"help\_button"**);

button4.**setOnAction**(**event**->{

CenterContent\_Content2\_StackPane\_AnchorPane1.**setVisible**(**true**);

aboutLabel.**setVisible**(**false**);

CenterContent\_Content2\_StackPane\_AnchorPane1.**toFront**();

});

textField1 = **new** **TextField**();

textField1.**getStyleClass**().**add**(**"textField\_search"**);

searchButton = **new** **Button**(**"🔎 SEARCH"**);

searchButton.**getStyleClass**().**add**(**"search\_button"**);

searchLabel = **new** **Label**(**""**);

searchLabel.**getStyleClass**().**add**(**"search\_label"**);

searchLabel.**setPadding**(**new** **Insets**(**10**));

button3.**setOnAction**(**new** **ButtonAboutClickHandler**());

searchButton.**setOnAction**(**new** **ButtonSearchClickHandler**());

button5 = **new** **Button**(**"IMAGE"**);

button5.**getStyleClass**().**add**(**"image\_button"**);

button6 = **new** **Button**(**"STORAGE"**);

button6.**getStyleClass**().**add**(**"storage\_button"**);

button7 = **new** **Button**(**"DESIGN"**);

button7.**getStyleClass**().**add**(**"design\_button"**);

image1 = **new** **Image**(**"file:wallpaper1.jpg"**);

imageView1 = **new** **ImageView**(image1);

imageView1.**setPreserveRatio**(**true**);

imageView1.**setFitWidth**(**300**);

imageView1.**setFitHeight**(**300**);

image2 = **new** **Image**(**"file:wallpaper2.jpg"**);

imageView2 = **new** **ImageView**(image2);

imageView2.**setPreserveRatio**(**true**);

imageView2.**setFitWidth**(**300**);

imageView2.**setFitHeight**(**300**);

image3 = **new** **Image**(**"file:wallpaper3.jpg"**);

imageView3 = **new** **ImageView**(image3);

imageView3.**setPreserveRatio**(**true**);

imageView3.**setFitWidth**(**300**);

imageView3.**setFitHeight**(**300**);

//I can use GridPane when needed

//gridPane1 = new GridPane();

//gridPane1.add(label1, 0, 0); gridPaneObject.add(control, column, row);

//gridPane1.add(label1, 1, 0);

//gridPane1.add(label1, 3, 0);

//gridPane1.add(label1, 0, 1);

//gridPane1.add(label1, 1, 1);

//gridPane1.add(label1, 3, 1);

//gridPane1.setHgap(10);

//gridPane1.setVgap(10);

//gridPane1.setGridLinesVisible(true);

listView1 = **new** **ListView**<String>();

listView1.**getItems**().**addAll**(

**"1.Getting Started."**,

**"2.Issues with this app."**,

**"3.Contact."**,

**"4.Other issues."**);

listView1.**setPrefSize**(**150**,**150**);

listView1.**setId**(**"listView1"**);

**AnchorPane**.***setTopAnchor***(listView1, 50.0);

**AnchorPane**.***setLeftAnchor***(listView1, 350.0);

CenterContent\_Content2\_StackPane\_AnchorPane1 =

**new** **AnchorPane**(listView1);

CenterContent\_Content2\_StackPane\_AnchorPane1.**setVisible**(**false**);

aboutLabel = **new** **Label**(**"ABOUT: This is a Testing App"**);

aboutLabel.**getStyleClass**().**add**(**"about\_label"**);

aboutLabel.**setPadding**(**new** **Insets**(**10**));

aboutLabel.**setVisible**(**false**);

CenterContent\_Content2\_StackPane = **new** **StackPane**(CenterContent\_Content2\_StackPane\_AnchorPane1, aboutLabel);

CenterContent\_Content2\_StackPane.**setAlignment**(Pos.**CENTER**);

bottomButt1 = **new** **Button**(**"Privacy Policy"**);

bottomButt1.**getStyleClass**().**add**(**"privary\_policy\_button"**);

//bottomButt1.setOnAction(event->{...doing something});

bottomButt2 = **new** **Button**(**"Copyright"**);

bottomButt2.**getStyleClass**().**add**(**"copyright\_button"**);

bottomButt3 = **new** **Button**(**"Terms of Use"**);

bottomButt3.**getStyleClass**().**add**(**"terms\_of\_use\_button"**);

hbox = **new** **HBox**(**10**, button1, button2, button3, button4, textField1, searchButton);

hbox.**getStyleClass**().**add**(**"top\_nav\_bar"**);

hbox.**setPadding**(**new** **Insets**(**10**));

//hbox.setAlignment(Pos.CENTER);

CenterContent\_Content1\_HBox= **new** **HBox**(**10**,imageView1, imageView2, imageView3);

CenterContent\_Content1\_HBox.**setPadding**(**new** **Insets**(**20**));

CenterContent\_VBox = **new** **VBox**(**10**,CenterContent\_Content1\_HBox,CenterContent\_Content2\_StackPane);

LeftVBox = **new** **VBox**(**10**, button5, button6, button7);

LeftVBox.**getStyleClass**().**add**(**"left\_nav\_bar"**);

LeftVBox.**setPadding**(**new** **Insets**(**10**));

bottomHbox = **new** **HBox**(**10**, bottomButt1, bottomButt2, bottomButt3);

bottomHbox.**getStyleClass**().**add**(**"bottom\_nav\_bar"**);

bottomHbox.**setPadding**(**new** **Insets**(**10**));

borderPane = **new** **BorderPane**();

borderPane.**setTop**(hbox);

borderPane.**setLeft**(LeftVBox);

borderPane.**setCenter**(CenterContent\_VBox);

borderPane.**setBottom**(bottomHbox);

scene = **new** **Scene**(borderPane);

scene.**getStylesheets**().**add**(**"file:style1.css"**);

primaryStage.**setScene**(scene);

primaryStage.**show**();

}

**class** **ButtonAboutClickHandler** **implements** EventHandler<ActionEvent>{

**@Override**

**public** **void** **handle**(**ActionEvent** event) {

aboutLabel.**setVisible**(**true**);

CenterContent\_Content2\_StackPane\_AnchorPane1.**setVisible**(**false**);

aboutLabel.**toFront**();

}

}

**class** **ButtonSearchClickHandler** **implements** EventHandler<ActionEvent>{

**@Override**

**public** **void** **handle**(**ActionEvent** event) {

**String** **userInput** = textField1.**getText**();

searchLabel.**setText**(**"Search Result: "**+userInput);

hbox.**getChildren**().**add**(searchLabel);

}

}

}

Syntax in **style1.css**

.top\_nav\_bar {

-fx-background-color:black;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: aqua;

}

.label {

-fx-background-color: black;

-fx-text-fill: white;

-fx-font-size: 20px;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: yellowgreen;

}

.home\_button, .menu\_button, .about\_button, .help\_button, .search\_button{

-fx-background-color: black;

-fx-text-fill: aqua;

-fx-font-size: 20px;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: yellowgreen;

}

.home\_button:hover, .menu\_button:hover,

.about\_button:hover, .help\_button:hover, .search\_button:hover{

-fx-background-color: aqua;

-fx-text-fill: white;

}

.textField\_search{

-fx-font-size: 20px;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: yellowgreen;

}

.about\_label, .search\_label {

-fx-background-color: black;

-fx-text-fill: white;

-fx-font-size: 20px;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: yellowgreen;

}

.left\_nav\_bar {

-fx-background-color:black;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: yellowgreen;

}

[id="listView1"]{

-fx-position:fixed;

-fx-top:10;

-fx-left:50;

}

.image\_button, .storage\_button, .design\_button{

-fx-background-color: black;

-fx-text-fill: yellowgreen;

-fx-font-size: 20px;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: aqua;

}

.image\_button:hover, .storage\_button:hover, .design\_button:hover{

-fx-background-color: aqua;

-fx-text-fill: white;

}

.bottom\_nav\_bar {

-fx-background-color:black;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: aqua;

}

.privary\_policy\_button, .copyright\_button, .terms\_of\_use\_button{

-fx-background-color: black;

-fx-text-fill: white;

-fx-font-size: 20px;

-fx-border-style: solid;

-fx-border-width: 2px;

-fx-border-color: yellowgreen;

}

.privary\_policy\_button:hover, .copyright\_button:hover, .terms\_of\_use\_button:hover{

-fx-background-color: aqua;

-fx-text-fill: white;

}

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.