

Programación de Interfaces Gráficas en Java

Objetivos: Animaciones, Multimedia, Menús

ELO329: Diseño y Programación Orientados a Objetos

Departamento de Electrónica

Universidad Técnica Federico Santa María

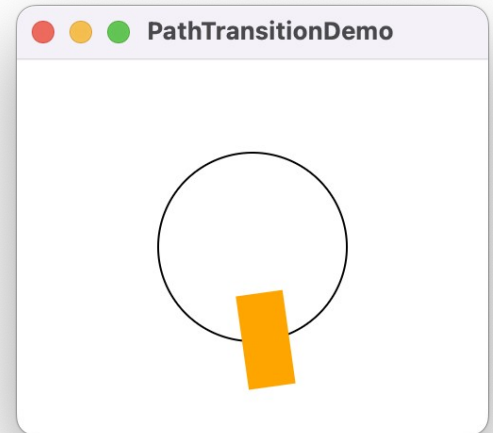
Este material ha sido preparado desde varias fuentes, entre ellas el material del Dr. Paul Fodor, Stony Brook University

Animaciones

- ❑ Una animación básicamente es cambiar las propiedades (tamaño, posición, color, etc) de un objeto a intervalos pequeños para dar la impresión de continuidad.
- ❑ Animaciones en JavaFX son divididas en animaciones de línea de tiempo y transiciones: clases Timeline y Transition. Ambas subclases de Animation.
- ❑ Hay varias subclases de Transition, cada una de ellas define la animación de un valor o propiedad particular.
- ❑ Así tenemos, por ejemplo:
 - Clase FadeTransition la cual cambia la opacidad (cualidad de opaco) a intervalos regulares.
 - Clase PathTransition la cual traslada las coordenada x e y del nodo a intervalos regulares.
- ❑ Con Timeline podemos definir animaciones de manera libre.

Ejemplo: PathTransitionDemo

```
public class PathTransitionDemo extends Application {
    public void start(Stage primaryStage) {
        Pane pane = new Pane();
        Rectangle rectangle = new Rectangle(0, 0, 25, 50);
        rectangle.setFill(Color.ORANGE);
        Circle circle = new Circle(125, 100, 50);
        circle.setFill(Color.WHITE);
        circle.setStroke(Color.BLACK);
        pane.getChildren().addAll(circle, rectangle);
        // Create a path transition
        PathTransition pt = new PathTransition();
        pt.setDuration(Duration.millis(4000));
        pt.setPath(circle);
        pt.setNode(rectangle);
        pt.setOrientation(PathTransition.OrientationType.ORTHOGONAL_TO_TANGENT);
        pt.setCycleCount(Timeline.INDEFINITE);
        pt.setAutoReverse(true); // comment this line and try
        //pt.setInterpolator(Interpolator.LINEAR); // uncomment and try
        pt.play(); // Start animation
        circle.setOnMousePressed(e -> pt.pause());
        circle.setOnMouseReleased(e -> pt.play());
        Scene scene = new Scene(pane, 250, 200);
        primaryStage.setTitle("PathTransitionDemo");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    /* main has been omitted */
}
```

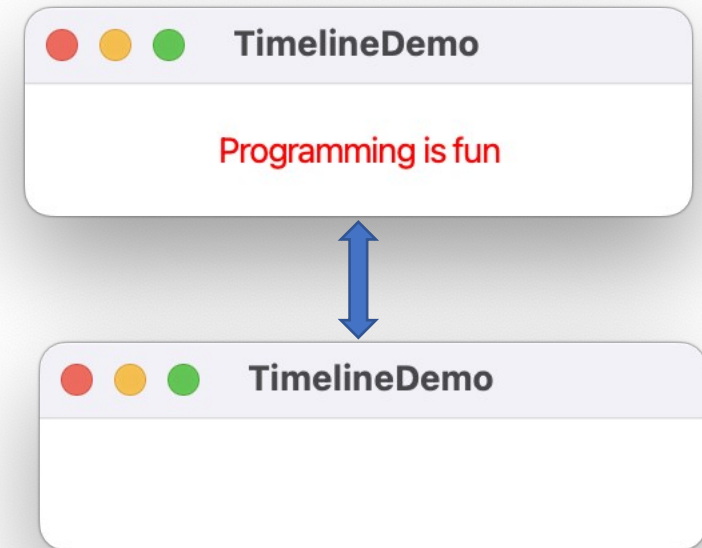


Ejemplo: TimelineDemo

```
public class TimelineDemo extends Application {
    public void start(Stage primaryStage) {
        StackPane pane = new StackPane();
        Text text = new Text(20, 50, "Programming is fun");
        text.setFill(Color.RED);
        pane.getChildren().add(text);
        // Create a handler for changing text
        EventHandler<ActionEvent> eH = e -> {
            if (text.getText().length() != 0)
                text.setText("");
            else
                text.setText("Programming is fun");
        };
        Timeline animation = new Timeline(new KeyFrame(Duration.millis(500), eH));
        animation.setCycleCount(Timeline.INDEFINITE);
        animation.play(); // Start animation
        text.setOnMouseClicked(e -> { // Pause and resume animation
            if (animation.getStatus() == Animation.Status.PAUSED)
                animation.play();
            else
                animation.pause();
        });
        Scene scene = new Scene(pane, 250, 50);
        primaryStage.setTitle("TimelineDemo");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    //... main
}
```

Define tiempos específicos de la animación. Aquí vemos un uso básico.

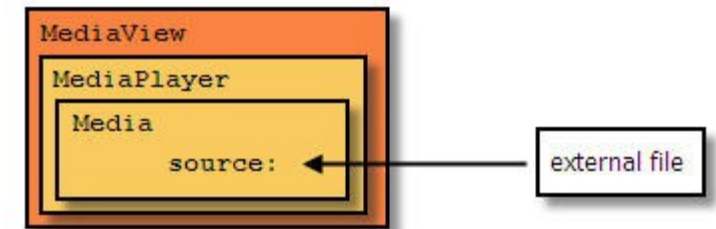
Handler invocado al término del tiempo



Reproducción de Audio y Video

Incorporando medios a la aplicación

- ❑ El paquete [javafx.scene.media](#) permite incorporar reproducción de audio y video en las aplicaciones.
- ❑ Actualmente hay soporte para una variedad de formatos de audio y video.
- ❑ Tres clases básicas son:
 - Media: Contiene información sobre el medio (su fuente, resolución, etc)
 - MediaPlayer: Provee métodos para reproducir un medio
 - MediaView: Subclase de Node, es requerida para video, sus instancias soportan animación y efectos.



Ejemplo: AudioMediaDemo

```
public class AudioMediaDemo extends Application {
    private static final String MEDIA_URL =
        "http://profesores.elo.utfsm.cl/~agv/elo329/JavaProg/JavaFX/AudioMediaDemo/droneSound.mp3";
    public void start(Stage primaryStage) {
        Media media = new Media(MEDIA_URL);
        MediaPlayer mediaPlayer = new MediaPlayer(media);
        mediaPlayer.setCycleCount(MediaPlayer.INDEFINITE);
        Button playButton = new Button(">");
        playButton.setOnAction(e -> {
            if (playButton.getText().equals(">")) {
                mediaPlayer.play(); playButton.setText("||");
            } else {
                mediaPlayer.pause(); playButton.setText(">");
            }
        });
        Slider slVolume = new Slider();
        slVolume.setPrefWidth(150);
        slVolume.setMaxWidth(Region.USE_PREF_SIZE);
        slVolume.setMinWidth(30);
        slVolume.setValue(50);
        mediaPlayer.volumeProperty().bind(slVolume.valueProperty().divide(100));
        HBox hBox = new HBox(10);
        hBox.setAlignment(Pos.CENTER);
        hBox.getChildren().addAll(playButton, new Label("Volume"), slVolume);
        Scene scene = new Scene(hBox, 350, 100);
        primaryStage.setTitle("AudioMediaDemo");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    // ... main
}
```

Otras opciones para especificar la fuente:

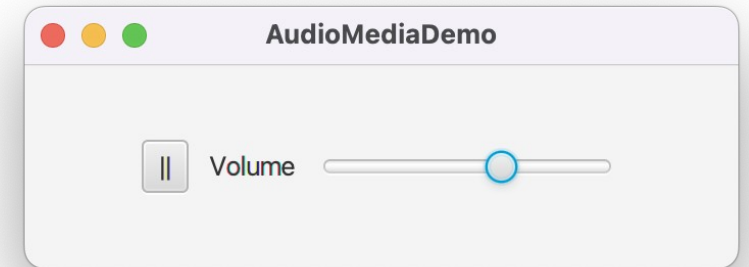
```
..... String MEDIA_URL = "file:/path/to/file/droneSound.mp3";
```

```
// or
```

```
..... String MEDIA_URL =
```

```
        AudioMediaDemo.class.getResource("droneSound.mp3").toExternalForm();
```

Cambio en slider,
cambia volumen



Ejemplo VideoMediaDemo

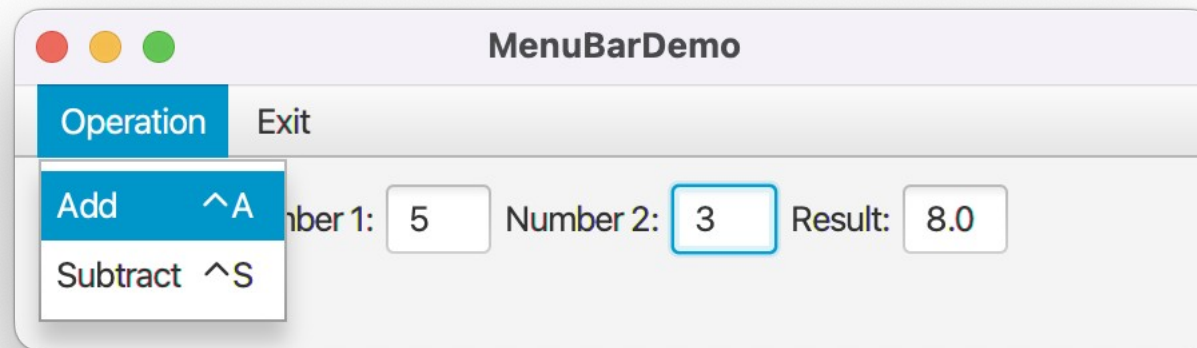
```
public class VideoMediaDemo extends Application {
    private static final String MEDIA_URL = VideoMediaDemo.class.getResource("video.mp4").toExternalForm();
    public void start(Stage primaryStage) {
        Media media = new Media(MEDIA_URL);
        MediaPlayer mediaPlayer = new MediaPlayer(media);
        MediaView mediaView = new MediaView(mediaPlayer);
        Button playButton = new Button(">");
        playButton.setOnAction(e -> {
            if (playButton.getText().equals(">")) {
                mediaPlayer.play(); playButton.setText("||");
            } else {
                mediaPlayer.pause(); playButton.setText(">");
            }
        });
        Button rewindButton = new Button("<<");
        rewindButton.setOnAction(e -> mediaPlayer.seek(Duration.ZERO));
        Slider slVolume = new Slider();
        slVolume.setPrefWidth(150);
        slVolume.setMinWidth(30); slVolume.setValue(50);
        mediaPlayer.volumeProperty().bind(slVolume.valueProperty().divide(100));
        HBox hBox = new HBox(10);
        hBox.setAlignment(Pos.CENTER);
        hBox.getChildren().addAll(playButton, rewindButton,
            new Label("Volume"), slVolume);
        BorderPane pane = new BorderPane();
        pane.setCenter(mediaView); pane.setBottom(hBox);
        Scene scene = new Scene(pane, 600, 350);
        primaryStage.setTitle("VideoMediaDemo");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
}
```



Menús

Menús

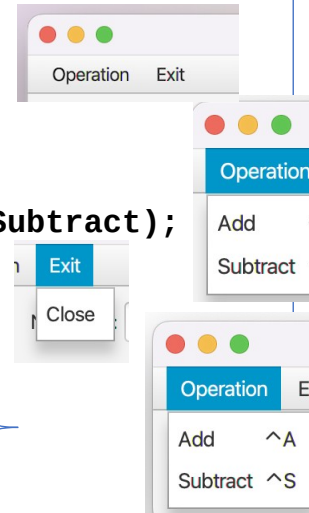
- ❑ JavaFx provee 5 clases para implementar menús: MenuBar, Menu, MenuItem, CheckMenuItem, y RadioButtonMenuItem.
- ❑ Ítems de un menú son instancias de MenuItem, CheckMenuItem, o RadioButtonMenuItem.
- ❑ Veamos un ejemplo...



Ejemplo: MenuBarDemo



```
public class MenuBarDemo extends Application {
    private TextField tfNumber1 = new TextField();
    private TextField tfNumber2 = new TextField();
    private TextField tfResult = new TextField();
    public void start(Stage primaryStage) {
        MenuBar menuBar = new MenuBar();
        Menu menuOperation = new Menu("Operation");
        Menu menuExit = new Menu("Exit");
        menuBar.getMenus().addAll(menuOperation, menuExit);
        MenuItem menuItemAdd = new MenuItem("Add");
        MenuItem menuItemSubtract = new MenuItem("Subtract");
        menuItemAdd.setAccelerator(KeyCombination.
            keyCombination("Ctrl+A"));
        menuItemSubtract.setAccelerator(KeyCombination.
            keyCombination("Ctrl+S"));
        HBox hBox1 = new HBox(5);
        hBox1.getChildren().addAll(new Label("Number 1:"), tfNumber1,
            new Label("Number 2:"), tfNumber2, new Label("Result:"),
            tfResult);
        hBox1.setAlignment(Pos.CENTER);
        tfNumber1.setPrefColumnCount(2);
        tfNumber2.setPrefColumnCount(2);
        tfResult.setPrefColumnCount(2);
    }
}
```



```
VBox vbox = new VBox(10);
vbox.getChildren().addAll(menuBar, hBox1);
Scene scene = new Scene(vbox, 450, 100);
primaryStage.setTitle("MenuBarDemo");
primaryStage.setScene(scene);
primaryStage.show();
```

// Handle menu actions

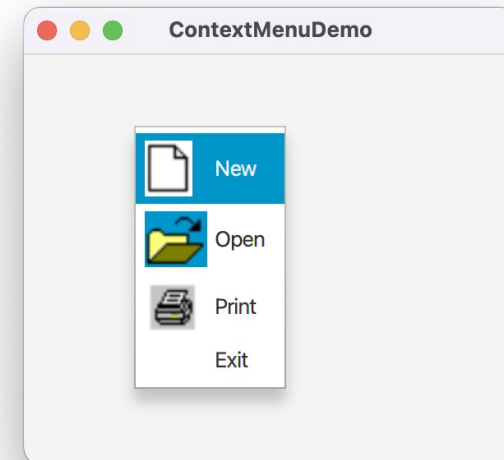
```
menuItemAdd.setOnAction(e -> perform('+'));
menuItemSubtract.setOnAction(e -> perform('-'));
menuItemClose.setOnAction(e -> System.exit(0));
```

```
private void perform(char operator) {
    double number1 = Double.parseDouble(tfNumber1.getText());
    double number2 = Double.parseDouble(tfNumber2.getText());
    double result = 0;
    switch (operator) {
        case '+': result = number1 + number2; break;
        case '-': result = number1 - number2; break;
    }
    tfResult.setText(result + "");
}

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

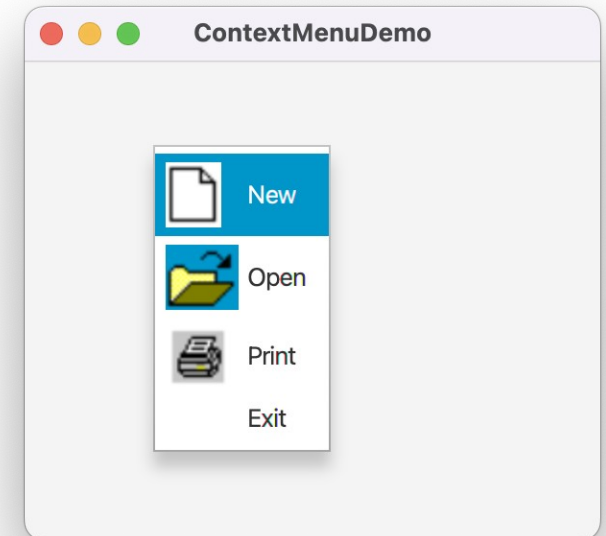
Menú de contexto (ContextMenu)

- ❑ Un menú de contexto (conocidos como menú popup) es como un menú regular, pero no tiene barra de menú y puede flotar en cualquier parte de la escena.
- ❑ La creación de un menú popup es similar al menú regular, la diferencia está en que un menú de contexto se puede asociar a cualquier instancia de Nodo.
- ❑ Una vez construido el menú de contexto éste parece al invocar:
- ❑ `void show(Node anchor, double screenX, double screenY)`



Ejemplo: ContextMenuDemo

```
public class ContextMenuDemo extends Application {  
    public void start(Stage primaryStage) {  
        ContextMenu contextMenu = new ContextMenu();  
        MenuItem menuItemNew = new MenuItem("New", new ImageView("image/new.gif"));  
        MenuItem menuItemOpen = new MenuItem("Open", new ImageView("image/open.gif"));  
        MenuItem menuItemPrint = new MenuItem("Print", new ImageView("image/print.gif"));  
        MenuItem menuItemExit = new MenuItem("Exit");  
        contextMenu.getItems().addAll(menuItemNew, menuItemOpen,  
                                       menuItemPrint, menuItemExit);  
  
        Pane pane = new Pane();  
        Scene scene = new Scene(pane, 300, 250);  
        primaryStage.setTitle("ContextMenuDemo");  
        primaryStage.setScene(scene);  
        primaryStage.show();  
        pane.setOnMousePressed(  
            e -> contextMenu.show(pane, e.getScreenX(), e.getScreenY()));  
        menuItemNew.setOnAction(e -> System.out.println("New"));  
        menuItemOpen.setOnAction(e -> System.out.println("Open"));  
        menuItemPrint.setOnAction(e -> System.out.println("Print"));  
        menuItemExit.setOnAction(e -> System.exit(0));  
    }  
    // .... main ...  
}
```



Podríamos ver más temas como transformaciones 2D y 3D, efectos visuales, etc., pero quedaremos hasta aquí.

Los interesados ver:

<https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>