# APPUNTI SU JAVA FX

Angelo Di Iorio
Università di Bologna

(dalla documentazione ufficiale)

# Java FX

- JavaFX is a set of graphics and media packages that enables developers to design, create, test, debug, and deploy rich client applications that operate consistently across diverse platforms
- **Java APIs**. JavaFX is a Java library that consists of classes and interfaces that are written in Java code.
- The JavaFX APIs are available as a fully integrated feature of the Java SE Runtime Environment (JRE) and the Java Development Kit (JDK).

# Hello World Application

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


    public void start(Stage primaryStage) {

        primaryStage.setTitle(”Welcome to FX!");

        primaryStage.show();
    }
}
```

# Java FX Application basics

- The main class of a JavaFX application extends the `javafx.application.Application` class
- The `start()` method is the main entry point for all JavaFX applications.
- A JavaFX application defines the user interface container by means of a **stage** and a **scene**.
- The JavaFX `Stage` class is the top-level JavaFX container.
- The JavaFX `Scene` class is the container for all content in a scene graph

# Scene graph and nodes

- The JavaFX **scene graph** is the starting point for constructing a JavaFX application.
- It is a **hierarchical tree of nodes** that represents all of the **visual elements of the application's user interface**. It can handle input and can be rendered.
- A single element in a scene graph is called a **node**. Each node (apart from the root) has a single parent and zero or more children.
- Different types of nodes:
  - Shapes (2D and 3D), images, media, text, UI controls, groups, containers

# Simple scene example

```
public void start(Stage primaryStage) {

  primaryStage.setTitle(”Welcome to FX!");

  Rectangle r = new Rectangle(300, 200, Color.RED);

  Group g = new Group();
  g.getChildren().add(r);

  Scene s1 = new Scene(g);

  primaryStage.setScene(s1);
  primaryStage.show();
}
```

# Layout containers

- Layout containers or panes can be used to allow for flexible and dynamic arrangements of the UI controls within a scene graph of a JavaFX application:
  - The `VBox` class arranges its content nodes vertically in a single column
  - The `HBox` class arranges its content nodes horizontally in a single row
  - The `GridPane` class enables the developer to create a flexible grid of rows and columns in which to lay out content nodes.
  - `AnchorPane` allows the edges of child nodes to be anchored to an offset from the anchor pane's edges.
  - ...
- To achieve a desired layout structure, different containers can be nested within a JavaFX application.

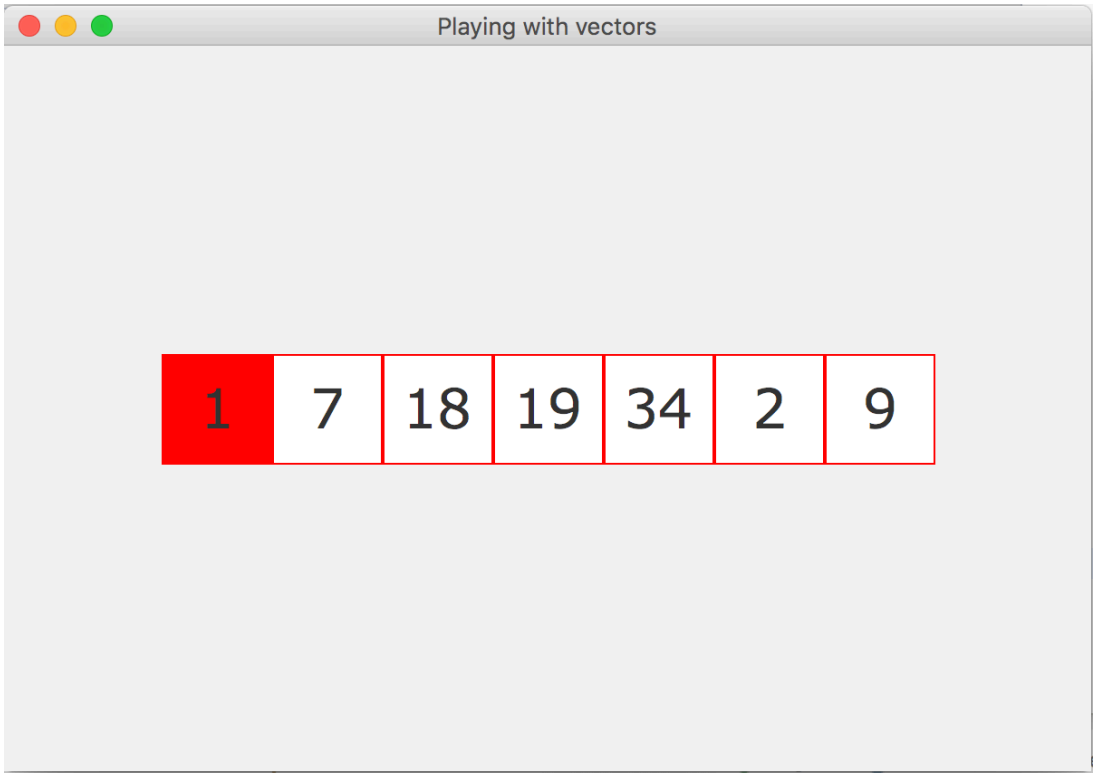
# ESERCIZI

# Esercizio 0

- Realizzare un'applicazione 'HelloWorld' in Java FX visualizzata come segue (dimensioni esatte non rilevanti):

# Esercizio 1

- Realizzare un'applicazione in Java FX che mostra un vettore di interi come in figura:



# Background: UI Controls

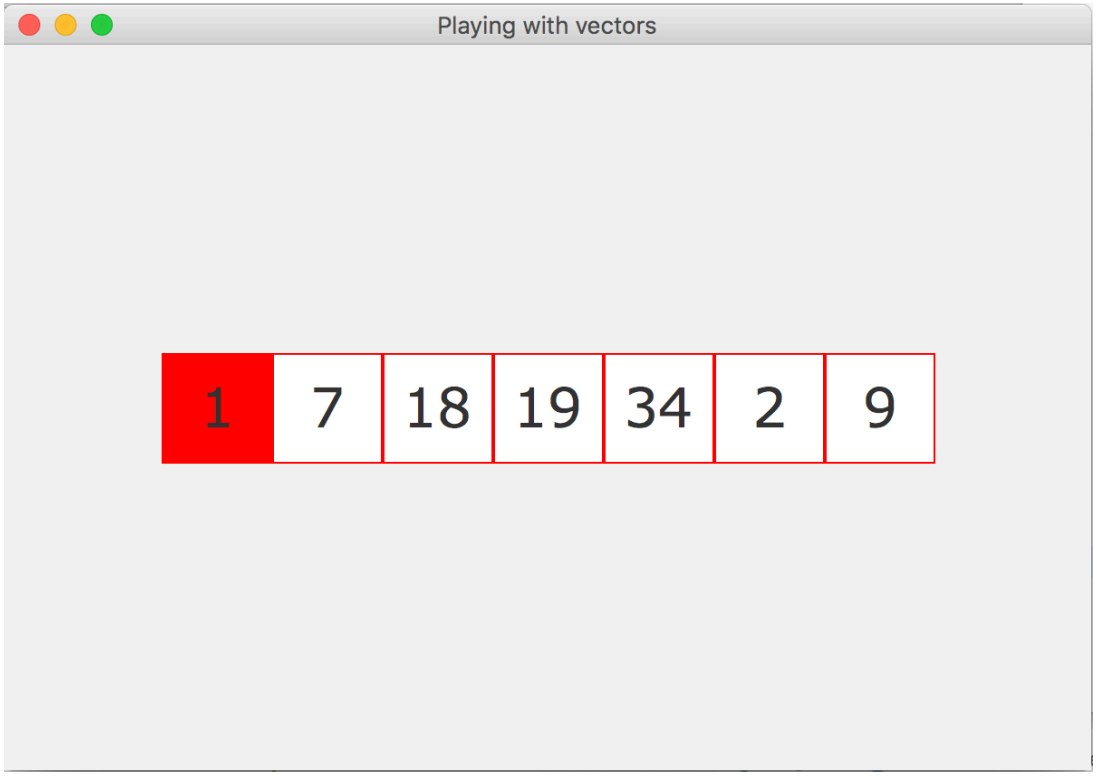
- The JavaFX UI controls available through the JavaFX API are built by using nodes in the scene graph
- JavaFX CSS allows for theming and skinning of the UI controls

# Background: events

- In JavaFX, an event is an instance of the `javafx.event.Event` class or any subclass of `Event`.
- JavaFX provides several events, including `DragEvent`, `KeyEvent`, `MouseEvent`, `ScrollEvent`, and others.
- To process an event a node must register an `EventHandler`, which implements the `EventHandler` interface.
- The `handle()` method of this interface provides the code that is executed when the event that is associated with the handler is received by the node that registered the handler.
- To register a handler, use the `addEventHandler()` method or the `setOn<EVENT>` methods

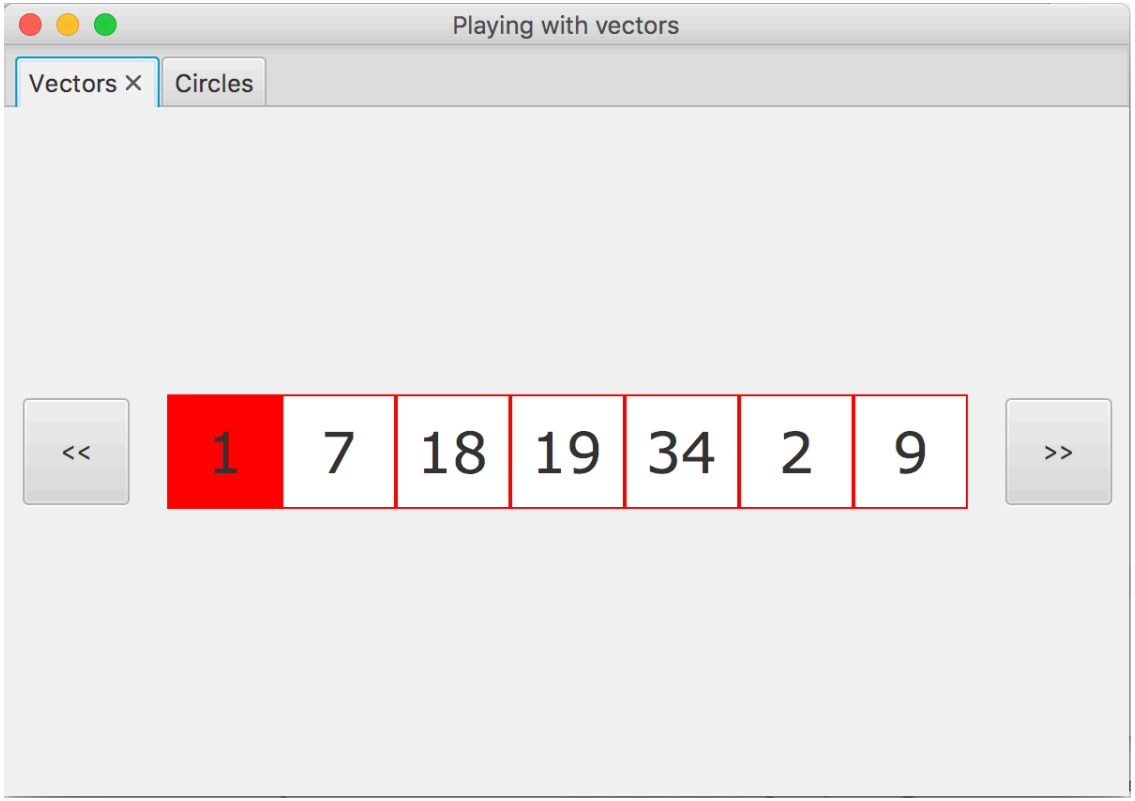
# Esercizio 2

- Aggiungere il seguente comportamento dinamico:
  - Puntando su un elemento lo sfondo diventa rosso; spostando il puntatore fuori dall'elemento lo sfondo torna bianco



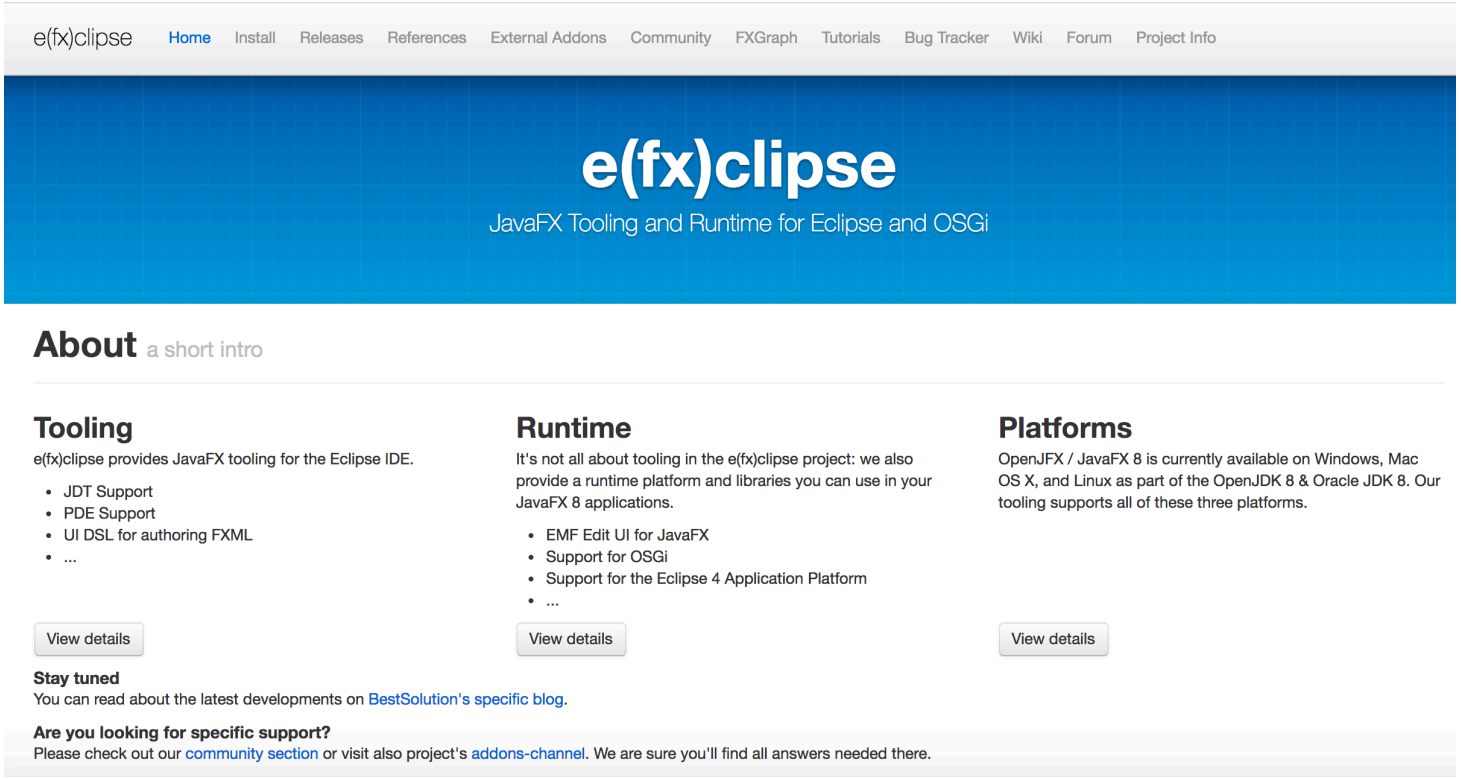
# Esercizio 3

- Aggiungere due bottoni (‘<<‘ e ‘>>’) per muoversi di una posizione avanti o indietro sul vettore, come mostrato in figura
- Organizzare l’applicazione in due tab



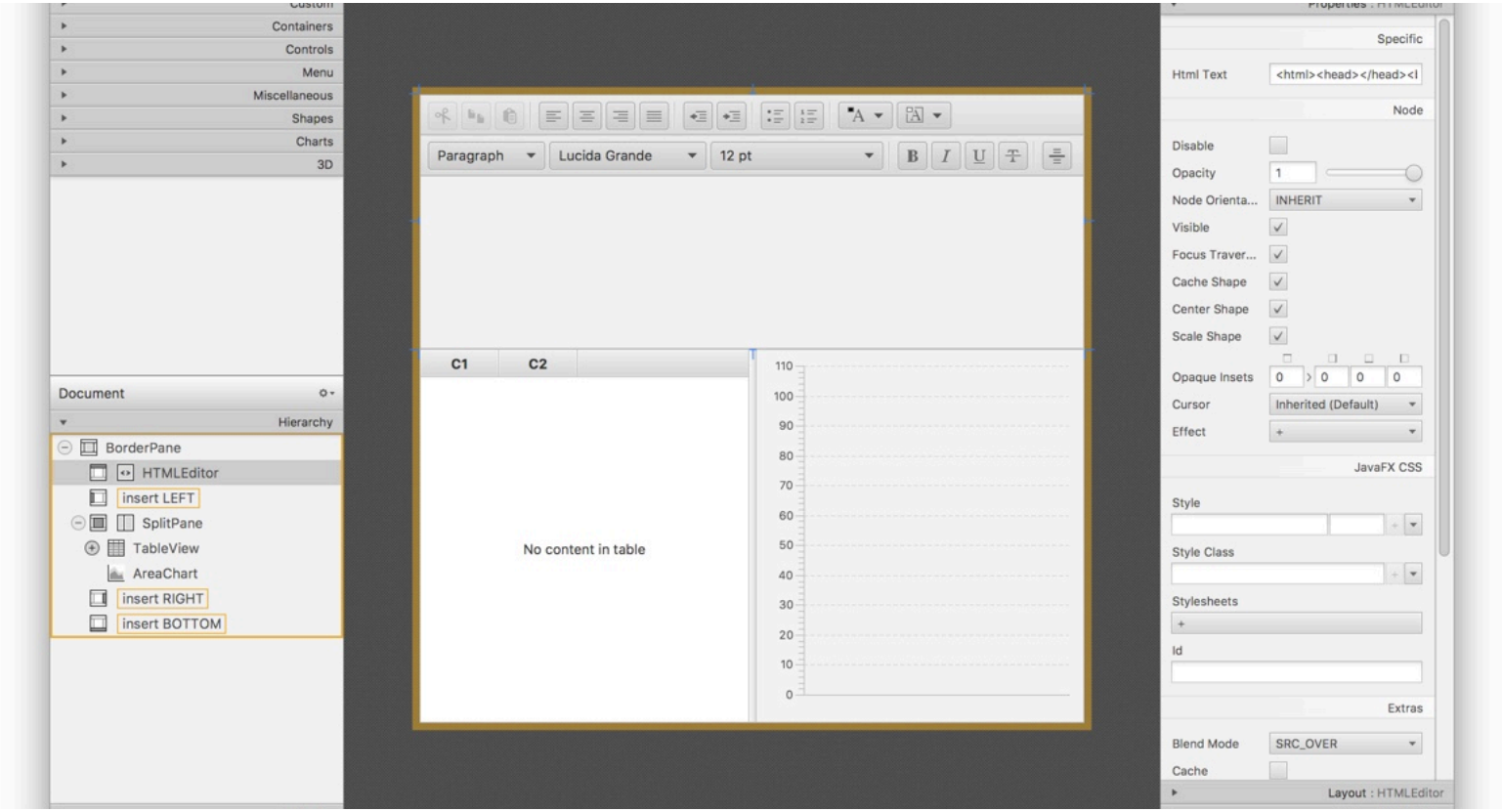
# JAVAFX TOOLS

# e(fx)clipse




http://www.eclipse.org/efxclipse/index.html

# Scene Builder




http://gluonhq.com/products/scene-builder/