



Programiranje korisničkih interfejsa 2017/18

# Desktop interfejs

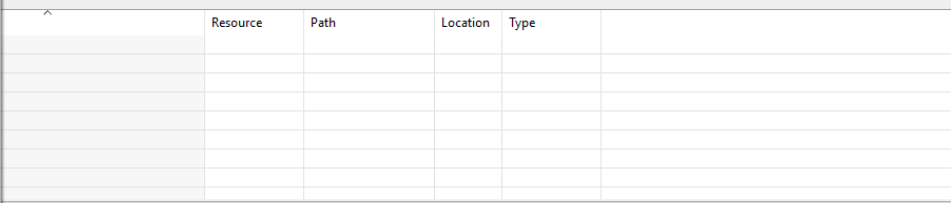
# Sadržaj

- Instalacija alata
- Kreiranje HelloWorldApp projekta
- Layout Panes
- Meniji
- CheckBox i RadioButton
- Input boxes
- Promena scene
- Prozori

# Instalacija alata

- Tehnologija: JavaFX
  - Platforma za dizajniranje grafičkih desktop i web aplikacija.
  - Dolazi kao biblioteka (jfxrt.jar) uz Java Development Kit.
  - Može se koristiti u bilo kom Java Integrated Development Environment –u za programiranje u Javi
- Preuzimanje alata:
  - <http://www.oracle.com/technetwork/java/javase/downloads/jdk9-downloads-3848520.html> - jdk9 (novembar 2017)
  - <https://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/keplersr2> - Eclipse IDE (novembar 2017)
  - <https://netbeans.org/downloads/> - NetBeans IDE (novembar 2017)
  - <https://www.jetbrains.com/idea/download/#section=windows> - IntelliJ IDEA IDE (novembar 2017)
  - Dostupni za sve platforme Windows/Linux/Mac
- Dokumentacija:
  - <https://docs.oracle.com/javase/8/javafx/api/toc.htm> (novembar 2017)

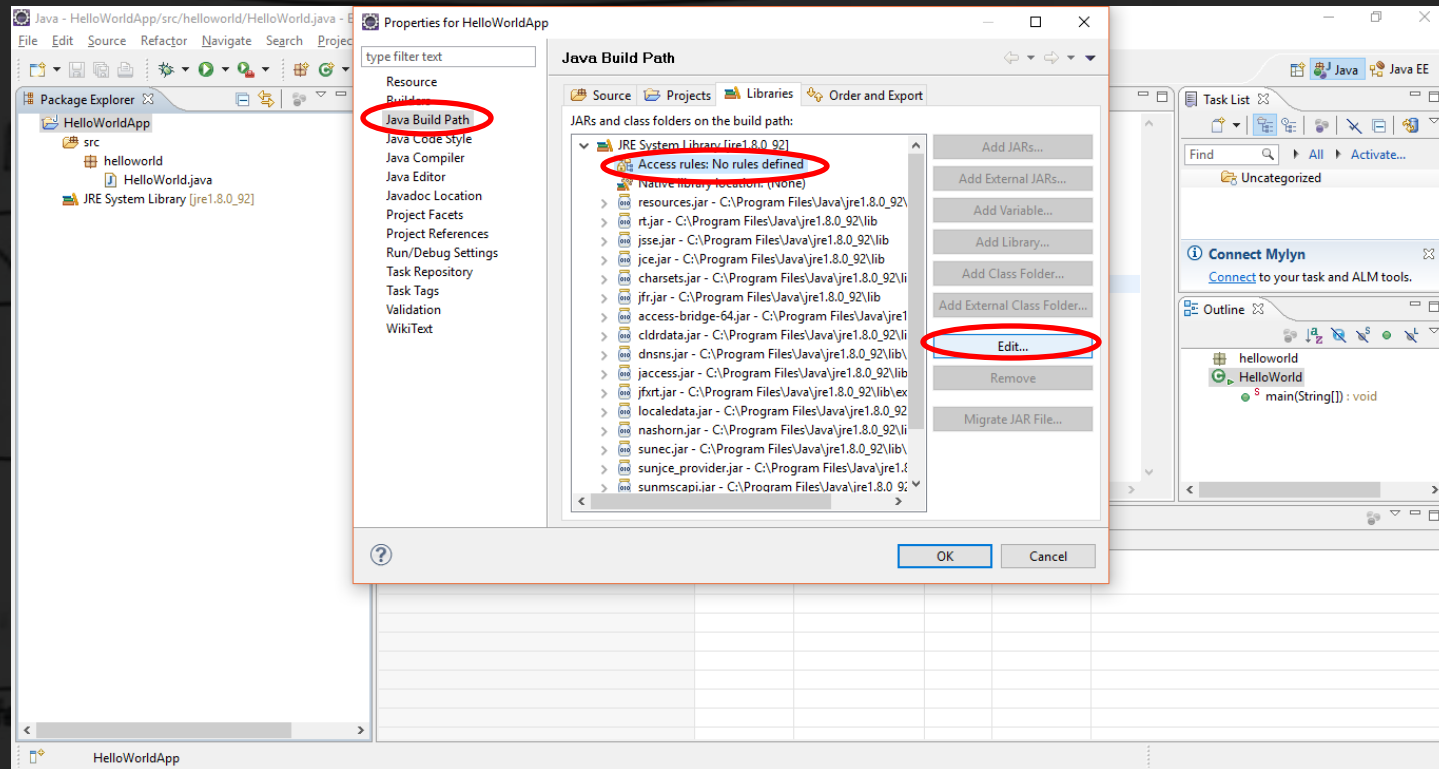
i. učiniti odgovarajuću .jar biblioteku dostupnom u build path-u projekta



# Kreiranje HelloWorldApp projekta

## 1) Kreiranje JavaFX projekta u Eclipse-u

- učiniti odgovarajuću .jar biblioteku dostupnom u build path-u projekta

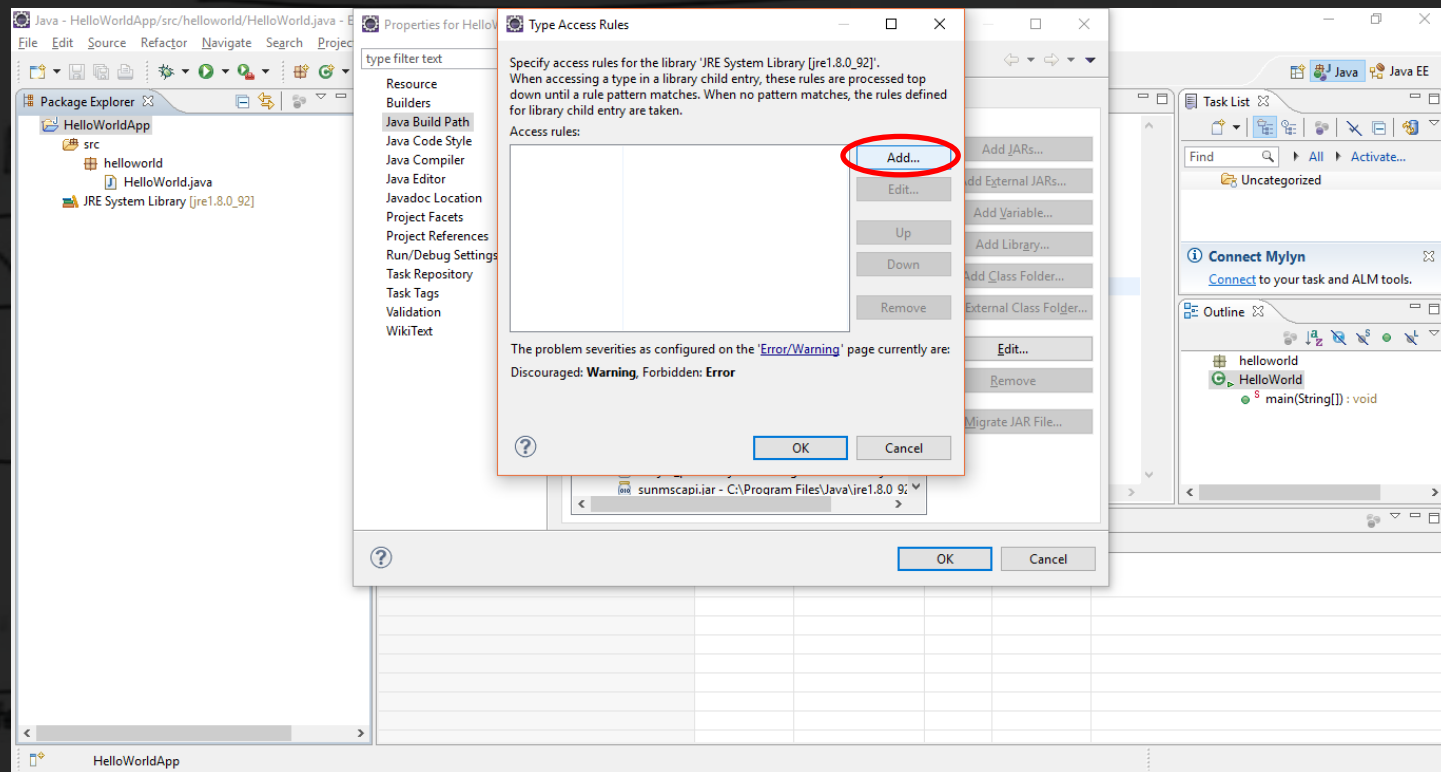




# Kreiranje HelloWorldApp projekta

## 1) Kreiranje JavaFX projekta u Eclipse-u

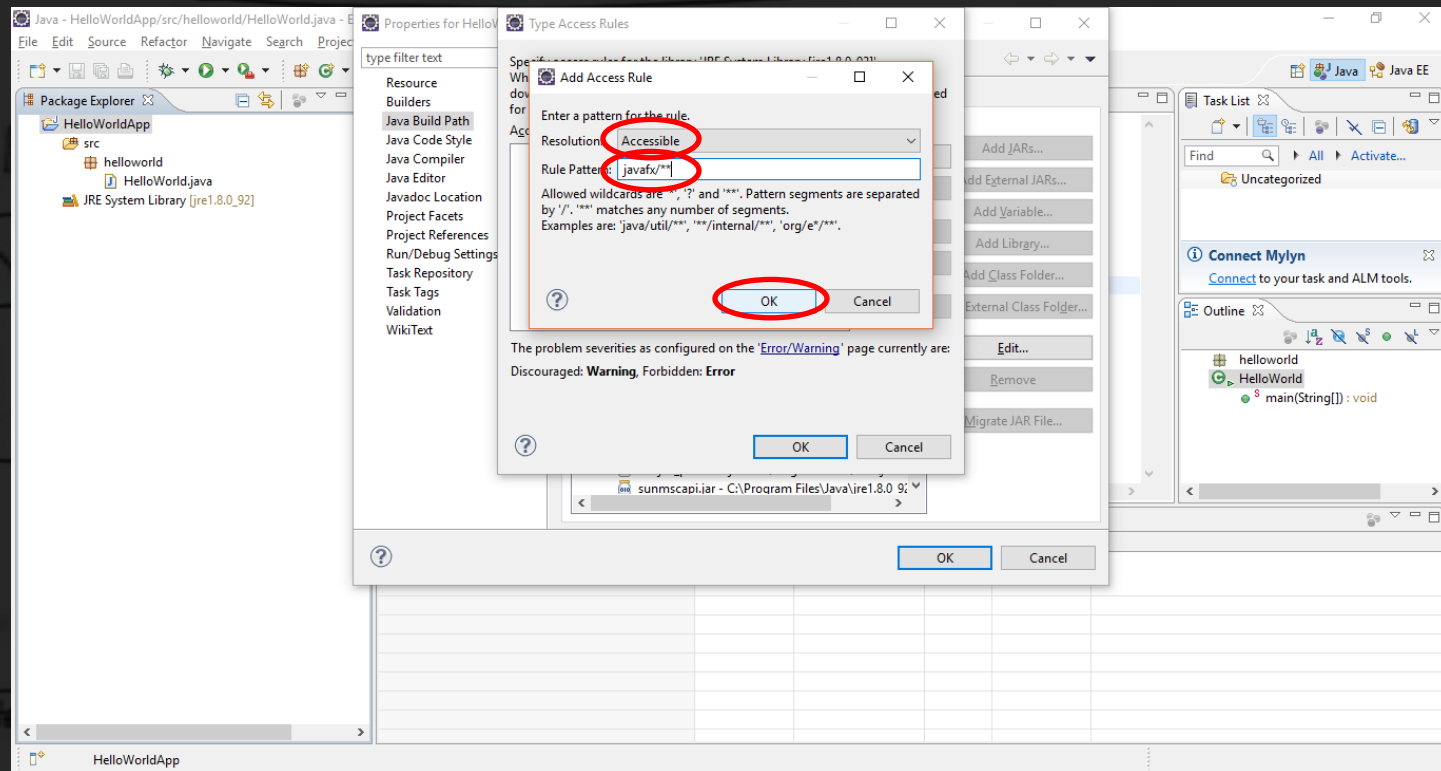
- učiniti odgovarajuću .jar biblioteku dostupnom u build path-u projekta



# Kreiranje HelloWorldApp projekta

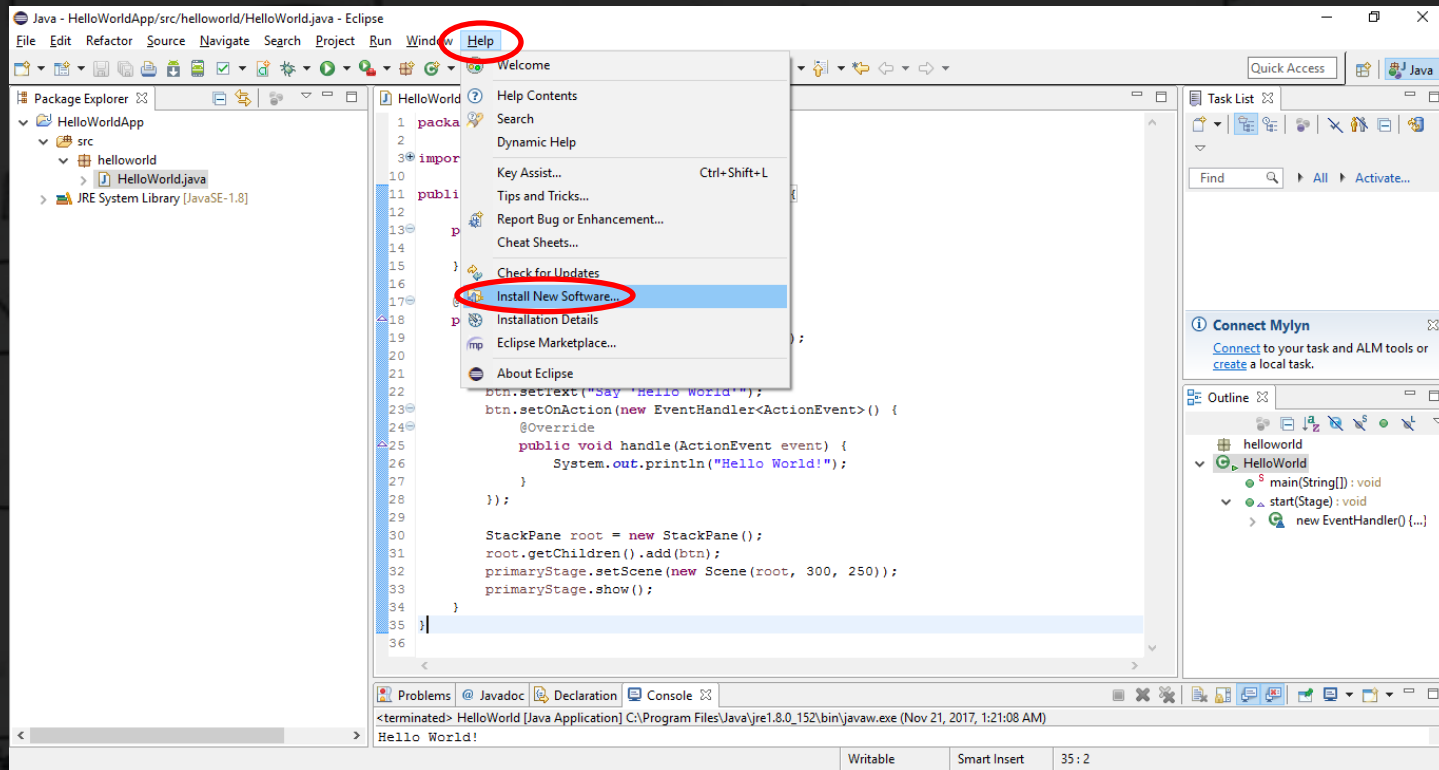
## 1) Kreiranje JavaFX projekta u Eclipse-u

- učiniti odgovarajuću .jar biblioteku dostupnom u build path-u projekta



# Kreiranje HelloWorldApp projekta

- 1) Kreiranje JavaFX projekta u Eclipse-u
  - ii. instalirati JavaFx plugin - e(fx)clipse

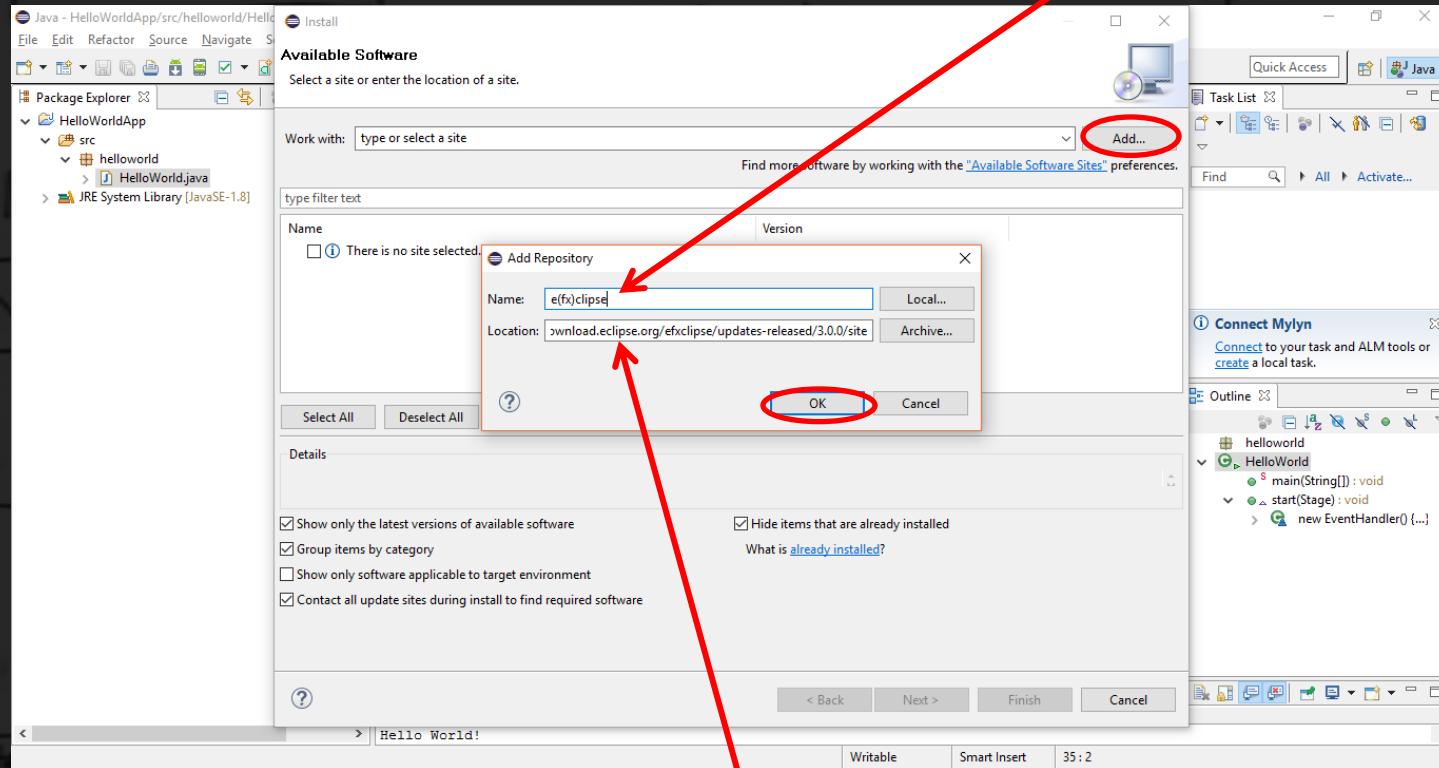




# Kreiranje HelloWorldApp projekta

- 1) Kreiranje JavaFX projekta u Eclipse-u
  - ii. instalirati JavaFx plugin - e(fx)clipse

e(fx)clipse

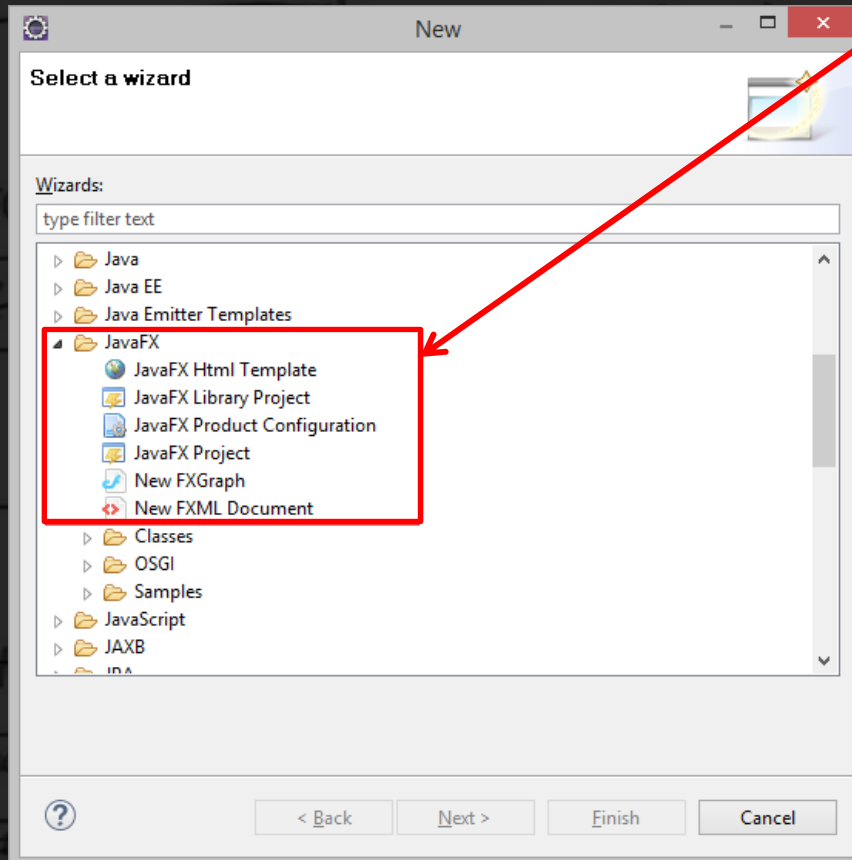


URL poslednje verzije <http://download.eclipse.org/efxclipse/updates-released/3.0.0/site>

# Kreiranje HelloWorldApp projekta

- 1) Kreiranje JavaFX projekta u Eclipse-u
  - ii. instalirati JavaFx plugin - e(fx)clipse

e(fx)clipse plugin omogućava direktno kreiranje JavaFX projekata.



Pored e(fx)clipse-a, moguće je instalirati layout alat koji omogućava jednostavnije kreiranje JavaFX aplikacija bez kodiranja (drag and drop princip) - JavaFX Scene Builder. Alat je dostupan na <http://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-1x-archive-2199384.html> (novembar 2017).

Nakon instalacije alata potrebno je u Window->Preferences->JavaFX postaviti SceneBuilder executable na putanju do .exe fajla.

# Kreiranje HelloWorldApp projekta

## 2) Osnovni elementi JavaFX aplikacije

The screenshot displays the Eclipse IDE interface for a Java project named 'HelloWorldApp'. The Package Explorer on the left shows the project structure: 'HelloWorldApp' contains a 'src' folder, which contains a 'helloworld' package, which in turn contains the 'HelloWorld.java' file. The main editor window shows the source code of 'HelloWorld.java', which is a JavaFX application. The code defines a 'HelloWorld' class that extends 'Application'. It includes a 'main' method to launch the application and a 'start' method that creates a 'StackPane' root, adds a 'Button' with the text 'Say 'Hello World'', and sets the scene. The 'handle' method of the button's event handler prints 'Hello World!' to the console. The Outline view on the right shows the class hierarchy and methods. The Console view at the bottom shows the output of the application, which is 'Hello World!'.

```
1 package helloworld;
2
3 import javafx.application.Application;
4
5
6
7
8
9
10
11 public class HelloWorld extends Application {
12
13     public static void main(String[] args) {
14         launch(args);
15     }
16
17     @Override
18     public void start(Stage primaryStage) {
19         primaryStage.setTitle("Hello World!");
20
21         Button btn = new Button();
22         btn.setText("Say 'Hello World'");
23         btn.setOnAction(new EventHandler<ActionEvent>() {
24             @Override
25             public void handle(ActionEvent event) {
26                 System.out.println("Hello World!");
27             }
28         });
29
30         StackPane root = new StackPane();
31         root.getChildren().add(btn);
32         primaryStage.setScene(new Scene(root, 300, 250));
33         primaryStage.show();
34     }
35 }
36
```

Connect Mylyn  
Connect to your task and ALM tools or create a local task.

Outline  
helloworld  
HelloWorld  
main(String[]): void  
start(Stage): void  
new EventHandler() {...}

Problems @ Javadoc Declaration Console  
<terminated> HelloWorld [Java Application] C:\Program Files\Java\jre1.8.0\_152\bin\javaw.exe (Nov 21, 2017, 1:21:08 AM)  
Hello World!

Writable Smart Insert 35: 2

# Kreiranje HelloWorldApp projekta

## 2) Osnovni elementi JavaFX aplikacije

Glavna klasa u aplikaciji mora proširivati `javafx.application.Application` klasu. To omogućava da aplikacija koja se kreira ima sve odlike JavaFX aplikacije.

`void javafx.application.Application.launch(String... arg0)` – metoda koja inicijalizuje JavaFX aplikaciju.

`void javafx.application.Application.start(Stage primaryStage)` throws Exception – metoda predstavlja glavnu ulaznu tačku JavaFX aplikacija. Poziva se odmah nakon inicijalizacije. Svaka klasa koja proširuje `javafx.application.Application` klasu mora da redefiniše ovu metodu.

```
1 package helloworld;
2
3 import javafx.application.Application;
4
10
11 public class HelloWorld extends Application {
12
13     public static void main(String[] args) {
14         launch(args);
15     }
16
17     @Override
18     public void start(Stage primaryStage) {
19         primaryStage.setTitle("Hello World!");
20
21         Button btn = new Button();
22         btn.setText("Say 'Hello World'");
23         btn.setOnAction(new EventHandler<ActionEvent>() {
24             @Override
25             public void handle(ActionEvent event) {
26                 System.out.println("Hello World!");
27             }
28         });
29
30         StackPane root = new StackPane();
31         root.getChildren().add(btn);
32         primaryStage.setScene(new Scene(root, 300, 250));
33         primaryStage.show();
34     }
35 }
```

# Kreiranje HelloWorldApp projekta

## 2) Osnovni elementi JavaFX aplikacije

JavaFX za opisivanje interfejsa aplikacija koristi Stage-ove i Scene-e.

Stage klasa predstavlja top-level kontejner aplikacije – prozor (naslovna linija sa caption dugmicima). Mora se učiniti vidljivim da bi bio prikazan.

Scene klasa predstavlja kontejner za sve što se nalazi na Stage-u – sve što se nalazi na prozoru. Sadržaj scene formira hijerarhijsku strukturu grafa, što omogućava da aplikacija bude responsive na promenu veličine prozora. Postavljanjem veličine scene postavlja je veličina prozora.

```
1 package helloworld;
2
3 import javafx.application.Application;
4
5
6
7
8
9
10
11 public class HelloWorld extends Application {
12
13     public static void main(String[] args) {
14         launch(args);
15     }
16
17     @Override
18     public void start(Stage primaryStage) {
19         primaryStage.setTitle("Hello World!");
20
21         Button btn = new Button();
22         btn.setText("Say 'Hello World'");
23         btn.setOnAction(new EventHandler<ActionEvent>() {
24             @Override
25             public void handle(ActionEvent event) {
26                 System.out.println("Hello World!");
27             }
28         });
29
30         StackPane root = new StackPane();
31         root.getChildren().add(btn);
32         primaryStage.setScene(new Scene(root, 300, 250));
33         primaryStage.show();
34     }
35 }
```



# Kreiranje HelloWorldApp projekta

## 2) Osnovni elementi JavaFX aplikacije

```
HelloWorld.java
1 package helloworld;
2
3 import javafx.application.Application;
10
11 public class HelloWorld extends Application {
12
13     public static void main(String[] args) {
14         launch(args);
15     }
16
17     @Override
18     public void start(Stage primaryStage) {
19         primaryStage.setTitle("Hello World!");
20
21         Button btn = new Button();
22         btn.setText("Say 'Hello World'");
23         btn.setOnAction(new EventHandler<ActionEvent>() {
24             @Override
25             public void handle(ActionEvent event) {
26                 System.out.println("Hello World!");
27             }
28         });
29
30         StackPane root = new StackPane();
31         root.getChildren().add(btn);
32         primaryStage.setScene(new Scene(root, 300, 250));
33         primaryStage.show();
34     }
35 }
```

javafx.scene.control.Button – JavaFX dugme.

javafx.event.EventHandler<ActionEvent> - interfejs za reagovanje na događaje od strane korisnika (rukovalac događajem).

void javafx.event.EventHandler<ActionEvent> .handle(ActionEvent event) – metoda za obradu događaja od strane korisnika; poziva se kada se pritisne dugme.

void javafx.scene.control.Button .setOnAction(EventHandler<ActionEvent>) – metoda koja postavlja zadati rukovalac događajem pritiska dugmeta.

javafx.scene.layout.StackPane – JavaFX layout.

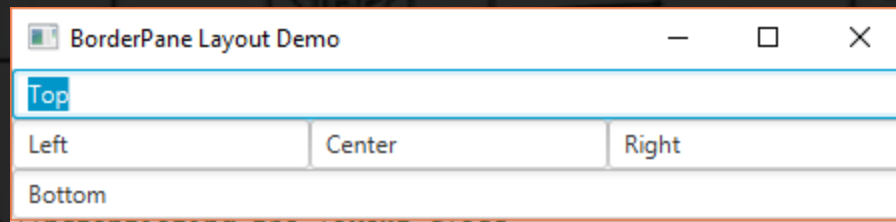
javafx.scene.layout.Pane.getChildren() .add(Node arg0) – metoda za dodavanje elementa na layout.

# JavaFX Layout Panes

- Omogućavaju automatsku preraspodelu prostora pri promeni veličine prozora.
- Osnovna klasa iz koje su izvedeni svi layout-i je `javafx.scene.layout.Pane`.
- Moguće je ručno raspoređivanje elemenata po layout-u postavljanjem apsolutne pozicije elementa (tada se koristi `Pane`). Tada nema preraspodele prostora prilikom promene veličine prozora
- Layout Pane-ovi u JavaFX-u:
  - `BorderPane`
  - `HBox`
  - `VBox`
  - `StackPane`
  - `GridPane`
  - `FlowPane`
  - `TilePane`
  - `AnchorPane`

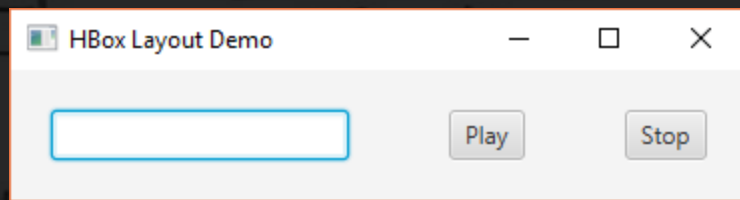
# BorderPane

- Deli prozor na pet regiona: top, left, right, bottom i center.
- Nepopunjen prostor pripada centralnom regionu.
- Ukoliko je prozor manji od prostora potrebnog za sve regione, oni se mogu preklapati, pri čemu se onaj region koji je kasnije dodat nalazi iznad regiona koji su dodati pre njega.
- Pogodan je za prozore koji imaju tool bar (u top regionu), status bar (u bottom regionu), panel za navigaciju (u left regionu), dodatne informacije (u right regionu) i prostor za rad (u center regionu).



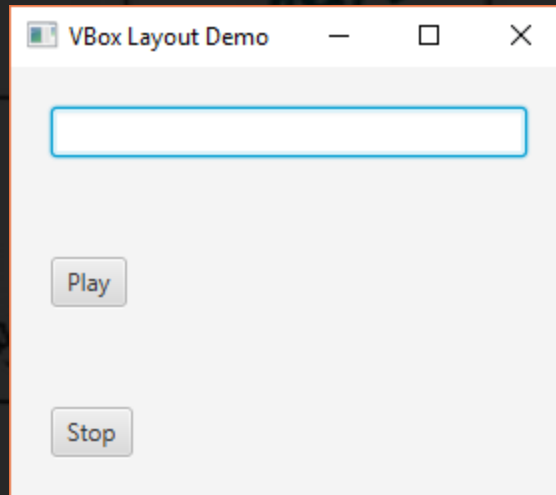
# HBox

- Postavlja elemente layout-a u jedan horizontalan red.



# VBox

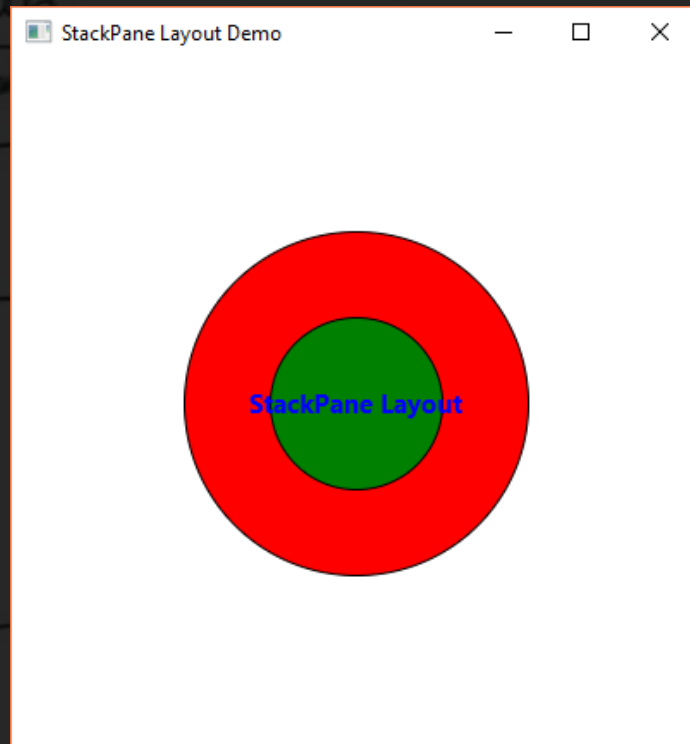
- Postavlja elemente layout-a u jednu vertikalnu kolonu.





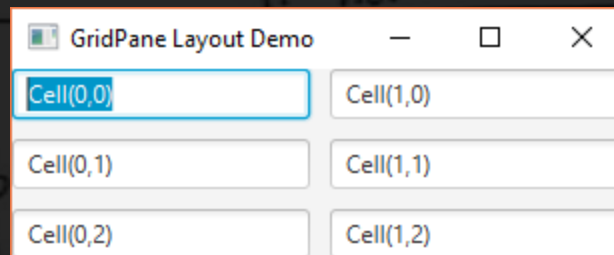
# StackPane

- Postavlja elemente layout-a jedan preko drugog, u stek.
- Element koji je prvi dodat nalazi se na dnu steka elemenata.
- Pozicija elementa na ovom layout-u može se podešavati postavljanjem odgovarajućeg alignment-a (utiče na sve elemente layout-a) i margin-a (za svaki element pojedinačno).



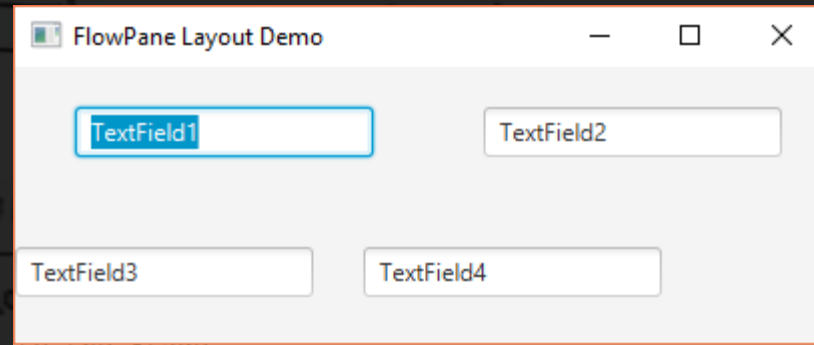
# GridPane

- Postavlja elemente layout-a u mrežu redova i kolona.
- Pogodan je za kreiranje formi.



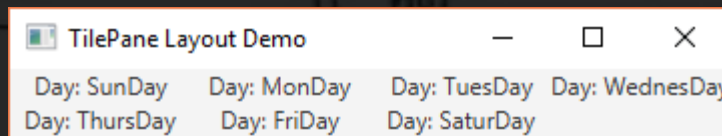
# FlowPane

- Obavija sve sadržane elemente – horizontalan po visini elementa, vertikaln po širini elementa.



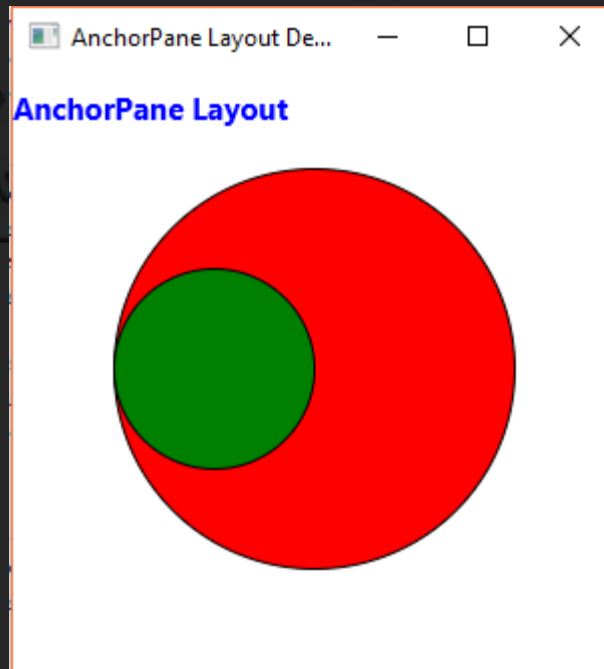
# TilePane

- Sličan je FlowPane-u, osim što su svi sadržani elementi uniformne veličine.



# AnchorPane

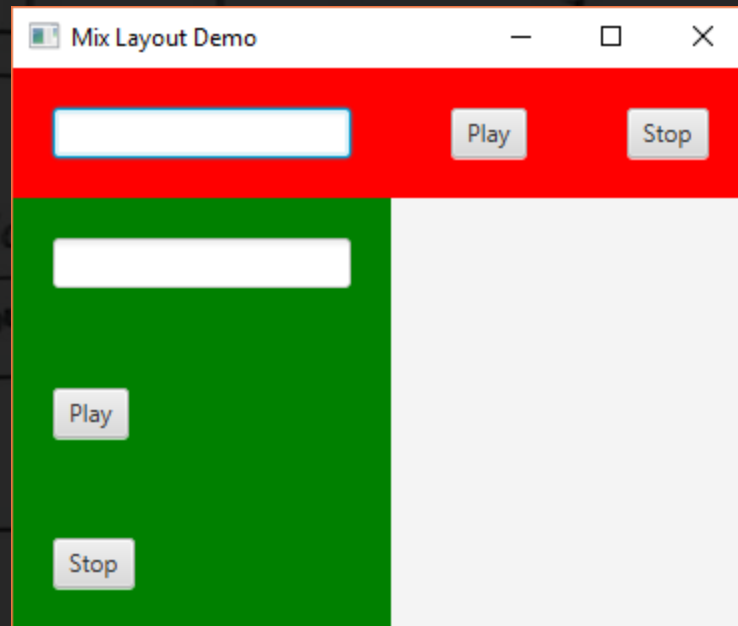
- Postavlja elemente layout-a na odgovarajuću udaljenost (offset) od njegovih ivica (top, bottom, left i right).
- Prilikom dodavanja elementa ovom layout-u neophodno mu je postaviti bar jedan offset.





# Kombinacija Layout-ova

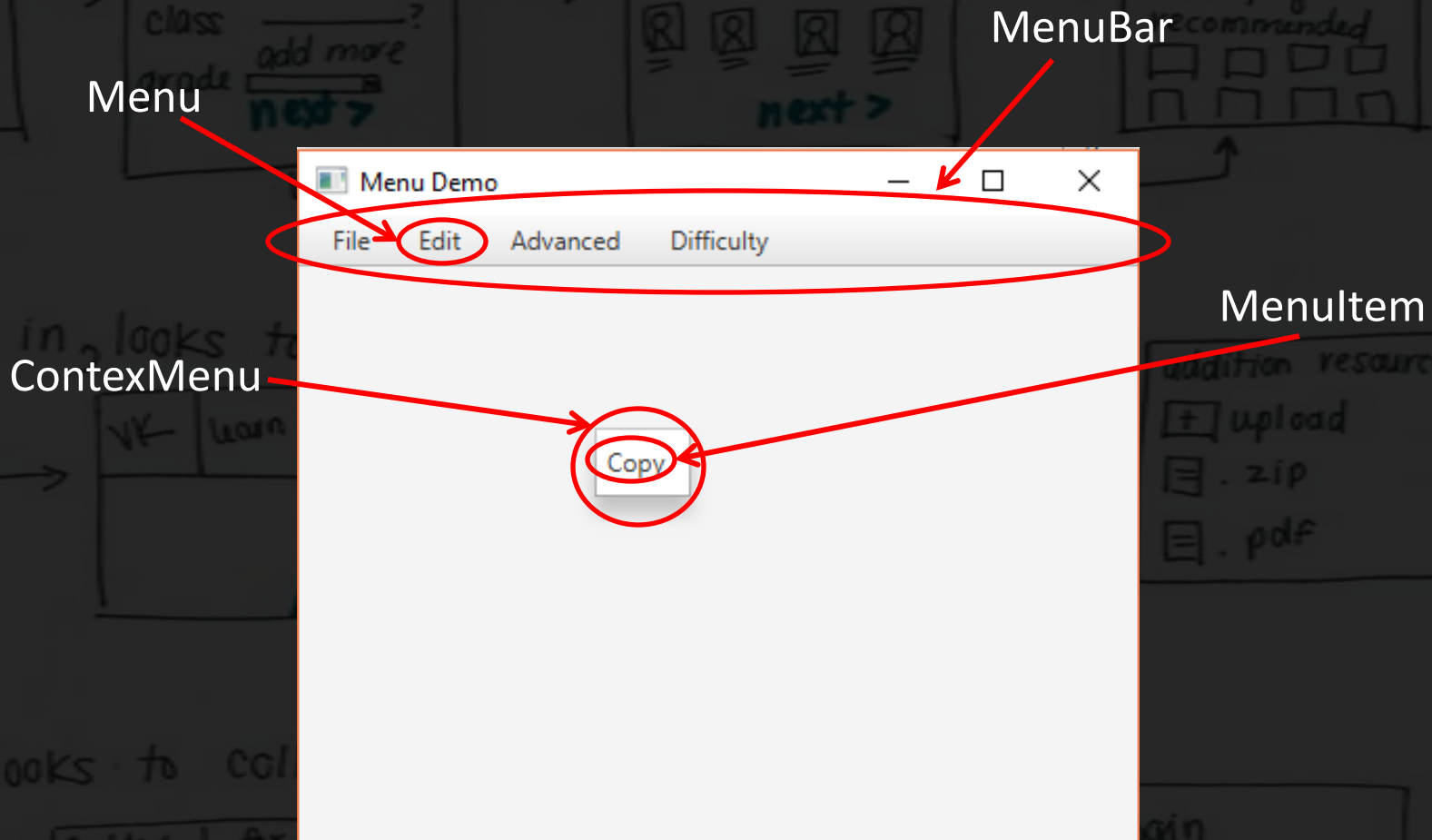
- HBox layout u top regionu BorderLayout layout-a, VBox layout u left regionu BorderLayout layout-a.



# JavaFX Meniji

- JavaFX omogućava kreiranje raznih vrsta menija i njihovih sadržaja:
  - MenuBar
  - Menu
  - MenuItem
  - CheckMenuItem
  - RadioMenuItem
  - SeparatorMenuItem
  - ContextMenu

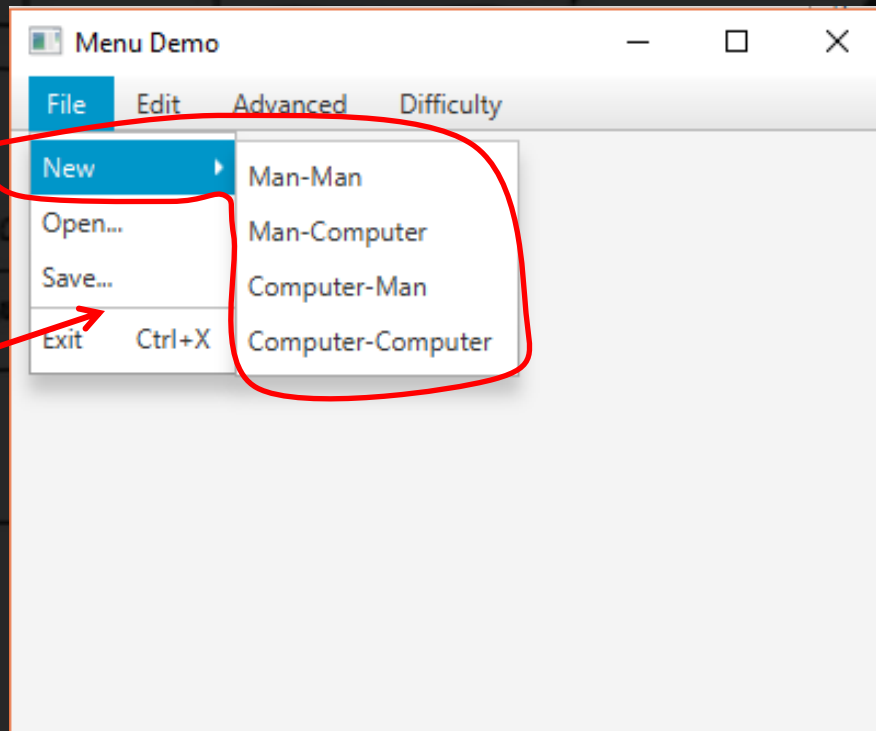
# JavaFX Meniji



# JavaFX Meniji

Menu (podmeni)

SeparatorMenuItem

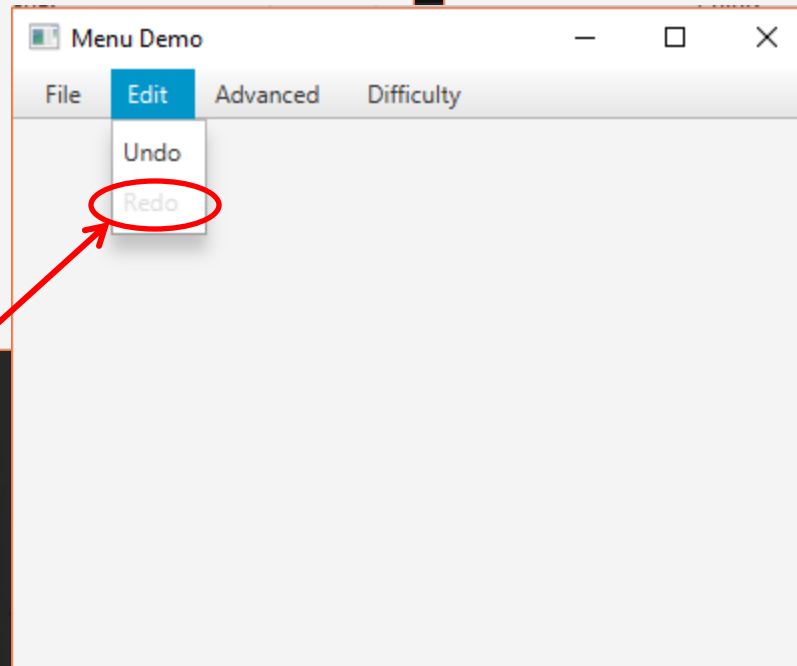
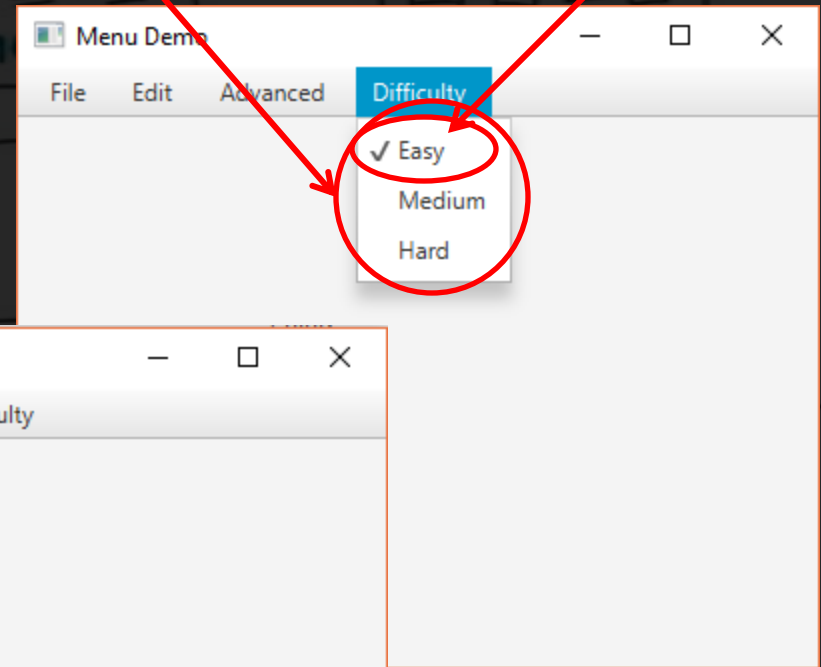
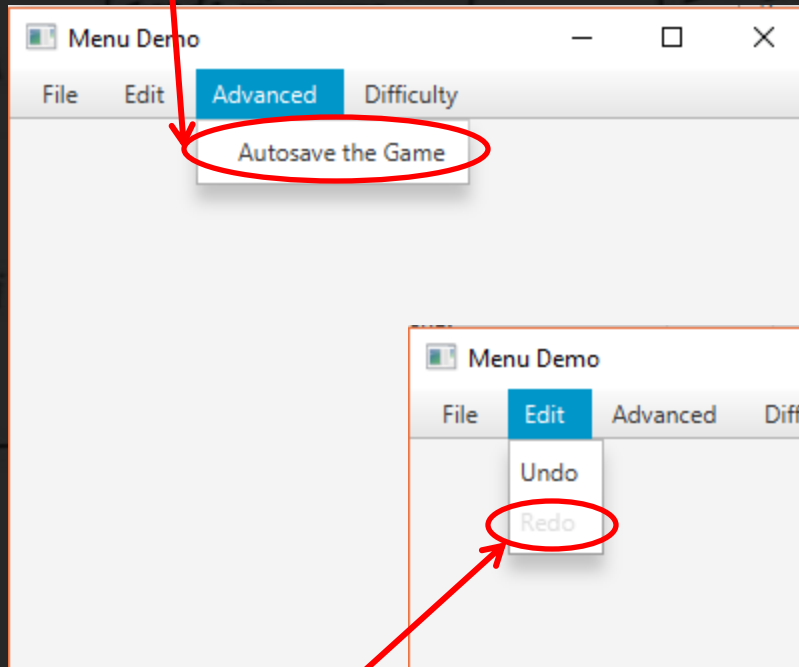


# JavaFX Meniji

CheckMenuItem

ToggleGroup

RadioMenuItem



MenuItem  
(onemogućeno - disabled)