

Dasar-Dasar Pemrograman 2

Lab 09

JavaFX

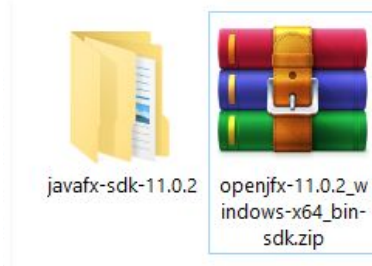


FAKULTAS
ILMU
KOMPUTER

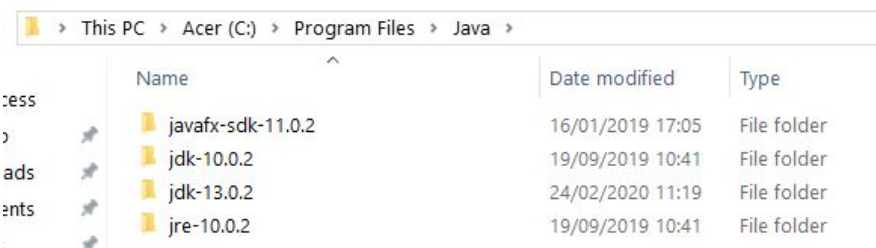
Panduan Instalasi JavaFX(Berdasarkan [Getting Started with JavaFX](#))

Text Editor:

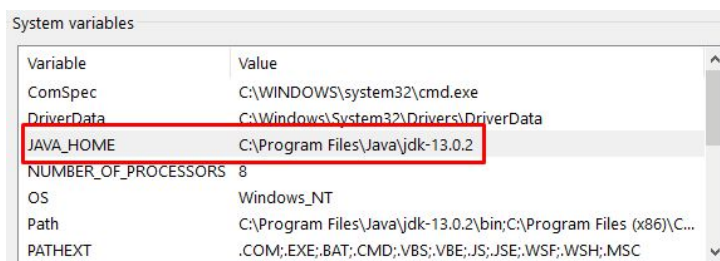
1. Download berkas JavaFX yang sesuai pada [JavaFX](#) dan Extract file Zip tersebut.



2. Letakkan folder *javafx-sdk-11.0.2* tersebut pada lokasi yang kamu inginkan. Sebagai contoh, kamu dapat meletakkannya di dalam folder java seperti berikut. (C:\Program Files\Java\jdk-13.0.2 atau sesuai versi Javamumu)



3. Pastikan kamu sudah mengeset *JAVA_HOME* pada *System Variable* di *System Environment*.



4. Jika sudah, maka langkah selanjutnya adalah melakukan perintah berikut pada terminal anda. Dalam kasus ini, **isi path dengan lokasi folder lib** yang ada pada folder *javafx-sdk-11.0.2* yang sudah di extract dan dipindahkan tadi.

Notes: set harus selalu dilakukan ketika membuka terminal baru

```
// Setting Path pada Terminal
set PATH_TO_FX="C:\Program Files\Java\javafx-sdk-11.0.2\lib"
```

5. Nah dengan demikian, kamu sudah siap untuk membuat program JavaFX pertama kamu.
6. Setelah program pertama kamu telah selesai dibuat, untuk mengcompile programnya, jalankan perintah `javac` dengan beberapa tambahan seperti berikut

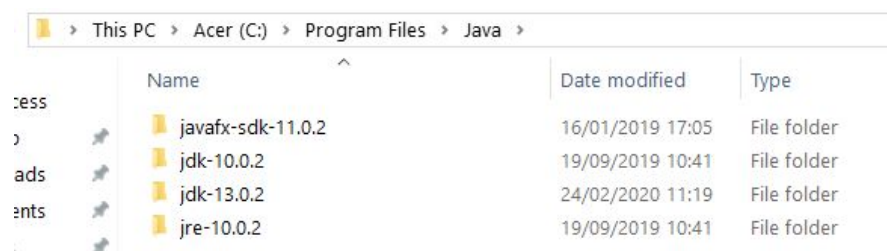
```
// Mengcompile Program
javac --module-path %PATH_TO_FX% --add-modules javafx.controls
NamaFileKamu.java
```

7. Yay, terakhir yang tinggal dilakukan adalah menjalankan programnya dengan perintah berikut

```
// Menjalankan Program
java --module-path %PATH_TO_FX% --add-modules javafx.controls
NamaFileKamu
```

Eclipse:

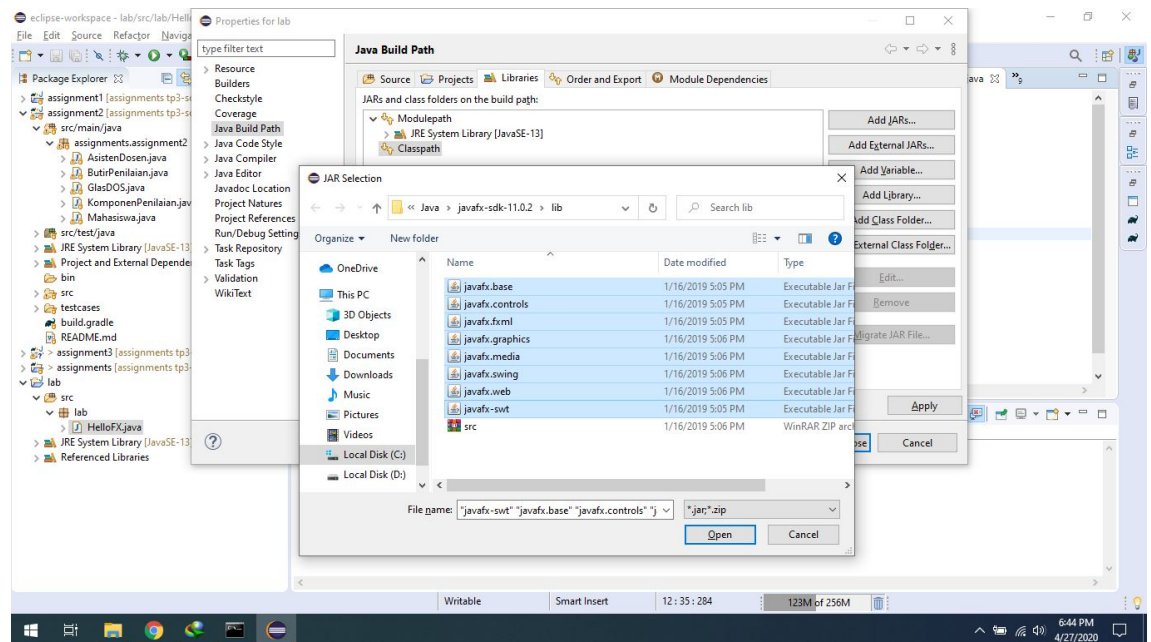
1. Sama seperti sebelumnya, unduh dahulu berkas JavaFX yang sesuai pada [JavaFX](#) dan extract file Zip tersebut.
2. Letakkan folder *javafx-sdk-11.0.2* tersebut pada lokasi yang kamu inginkan. Sebagai contoh, kamu dapat meletakkannya di dalam folder java seperti berikut. (C:\Program Files\Java\jdk-13.0.2 atau sesuai versi Javamu)



Name	Date modified	Type
javafx-sdk-11.0.2	16/01/2019 17:05	File folder
jdk-10.0.2	19/09/2019 10:41	File folder
jdk-13.0.2	24/02/2020 11:19	File folder
jre-10.0.2	19/09/2019 10:41	File folder

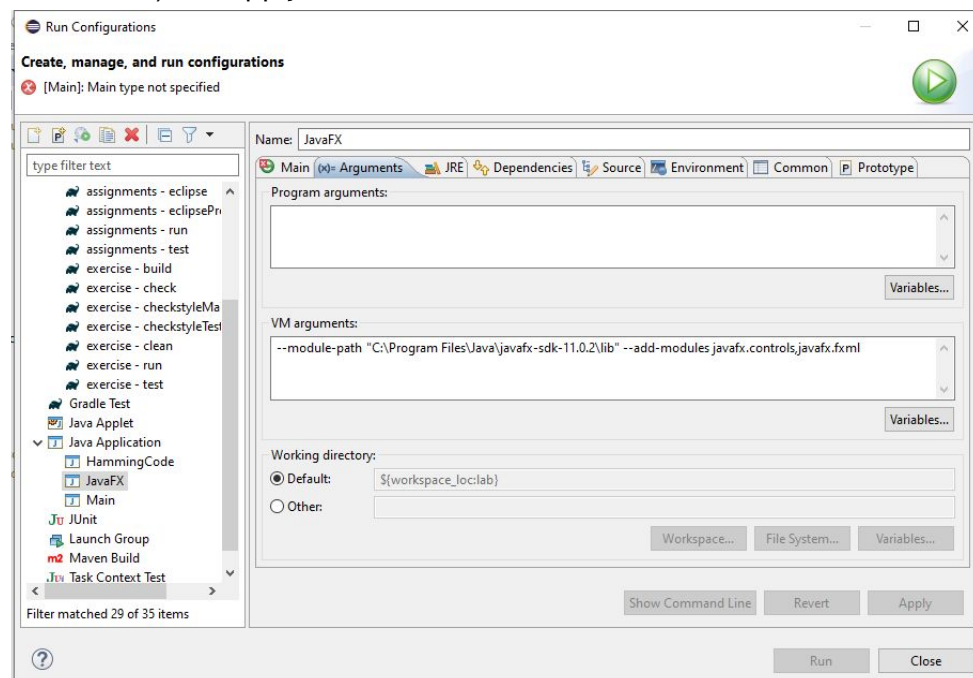
3. Buka Project -> Properties -> Java Build Path -> Libraries -> Classpath -> Add External JARs lalu pilih semua berkas jar milik JavaFX lalu pilih Open dan Apply

and Close.



4. Pilih Run -> Run Configurations... -> New launch configuration (icon kertas) -> Arguments lalu masukkan VM arguments seperti berikut:

```
--module-path "C:\Program Files\Java\javafx-sdk-11.0.2\lib"
--add-modules javafx.controls,javafx.fxml (sesuaikan dengan path
JavaFX-mu) lalu Apply dan Close.
```



5. Run program JavaFX-mu seperti biasa.

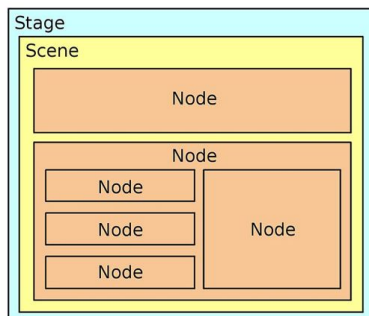
Materi

JavaFX

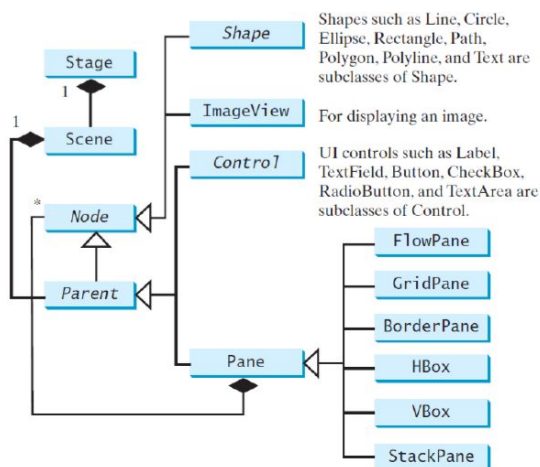
Sebuah Library untuk membuat GUI di Java yang menggunakan banyak Object Oriented Programming dalam penerapannya. Sebuah program JavaFX disebut sebagai application.

```
import javafx.application.Application;
```

JavaFX menggunakan stage dan scene untuk menampilkan berbagai Node. Stage disini bersifat sebagai top-level container yang bisa menampung scene sementara scene merupakan container yang akan menampung semua node yang kita akan buat nantinya.



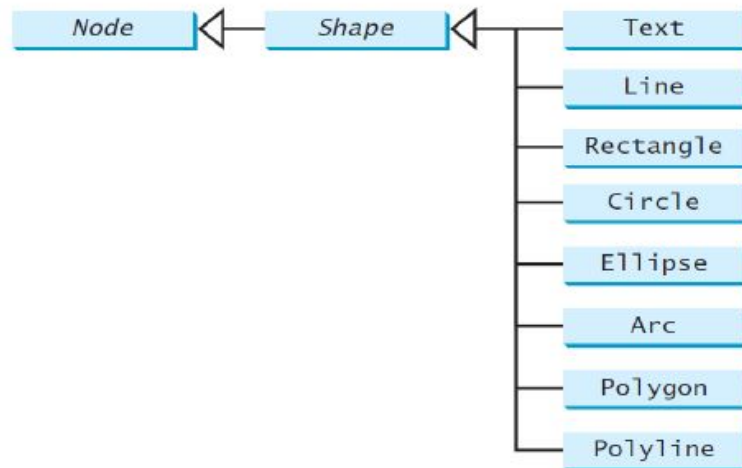
Untuk penggambaran hierarkinya bisa dilihat pada gambar berikut.



Node memiliki banyak jenis

- **Controls** (Memungkinkan Interaksi dengan User)
Contohnya adalah button, text field, label, check box, dan lain lain.
- **Shape** (Bentuk grafik 2D atau 3D)
 - **Text**(x, y , string)
 - **Line**(startX, startY, endX, endY)
 - **Rectangle**(x, y, width, height)
 - **Circle**(centerX, centerY, radius)
 - **Arc**(centerX, centerY, radiusX, radiusY, startAngle, angleExtend)

- Dan masih banyak lagi! [JavaFX - 2D Shapes](#)



- **Container/Layout**

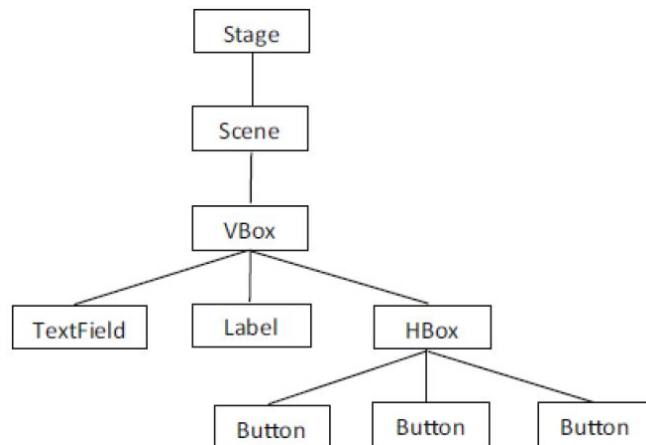
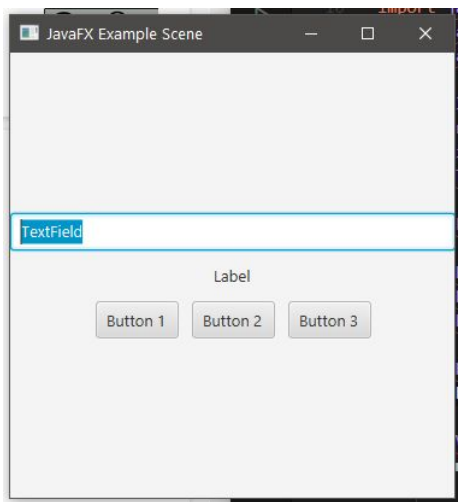
Node juga dapat merupakan container loh! Container sendiri merupakan komponen yang dapat menampung node lain. Tiap Container memiliki spesifikasi dan ciri masing-masing untuk mengatur child node didalamnya.

- [Hbox\(\)](#)
Mengatur semua child node didalamnya menjadi satu baris horizontal tunggal.
- [Vbox\(\)](#)
Mengatur semua child node didalamnya menjadi satu baris vertikal tunggal.
- [StackPane\(\)](#)
Mengatur child node dalamnya dengan menumpuk node di atas node yang lain. Node yang ditambahkan pertama ditempatkan di bagian bawah tumpukan dan node berikutnya ditempatkan di atasnya.
- [GridPane\(\)](#)
Mengatur penempatan child node didalamnya berdasarkan grid dari baris dan kolom.
- Dan lain-lain [JavaFX - Layout Panes\(Containers\)](#).

Catatan:

- Container bisa menampung banyak node (ingat, container itu sendiri adalah node).
- Kita tidak harus selalu menggunakan Container, kita bisa juga hanya menggunakan [Group](#)(node1, node2, ...) untuk menggroup node-node yang sudah kita buat.
- Kunci dari JavaFX adalah Kreativitasmu!

Contoh 1:



Pada contoh diatas, **Scene** mempunyai root node berupa sebuah **VBox**— yaitu sebuah container yang mengatur child nodes secara vertikal. **VBox** memiliki 3 child yaitu sebuah **TextField**, sebuah **Label** dan sebuah **HBox**. **HBox** adalah sebuah container yang mengatur child nodes secara horizontal. **HBox** memiliki 3 child node berupa **Button**. Berikut adalah Program untuk menjalankan application diatas.

```
import javafx.application.Application;
import javafx.geometry.Pos;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.TextField;
import javafx.scene.paint.Color;
import javafx.scene.text.Font;
import javafx.scene.control.Label;
import javafx.scene.layout.VBox;
import javafx.scene.layout.HBox;
import javafx.stage.Stage;

public class CobaJavaFX extends Application {
    @Override
    public void start(Stage stage){
        TextField field = new TextField("TextField");

        Label label = new Label("Label");

        Button button1 = new Button("Button 1");
        Button button2 = new Button("Button 2");
```

```

        Button button3 = new Button("Button 3");

        HBox horhor = new HBox(10, button1, button2);
        horhor.getChildren().add(button3);
        horhor.setAlignment(Pos.CENTER);

        VBox mainBox= new VBox(10, field, label);
        mainBox.getChildren().add(horhor);
        mainBox.setAlignment(Pos.CENTER);

        Scene scene= new Scene(mainBox, 350, 350);
        stage.setTitle("JavaFX Example Scene"); //membuat judul stage
        stage.setScene(scene);
        stage.show();
    }

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

```

Contoh 2:

Jalankan Program Berikut

```

import java.io.FileInputStream;
import java.io.FileNotFoundException;
import javafx.application.Application;
import javafx.geometry.Pos;
import javafx.scene.paint.Color;
import javafx.scene.control.Label;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.layout.VBox;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.stage.Stage;

public class CobaJavaFX extends Application {
    @Override

```

```

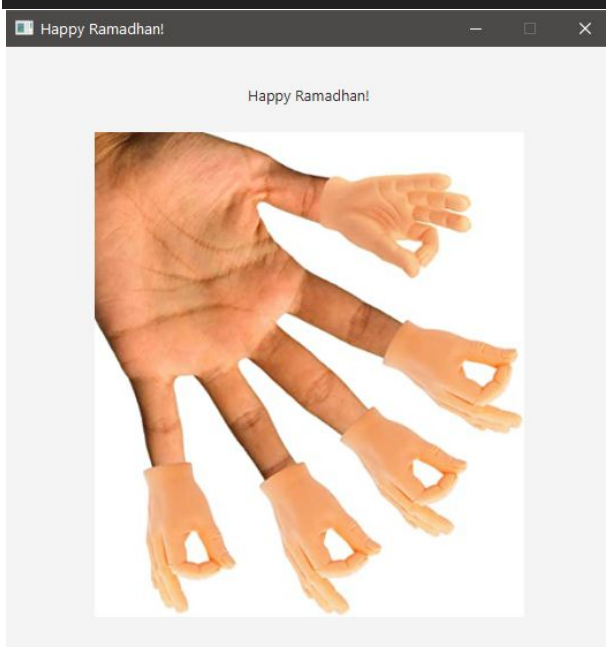
public void start(Stage stage) throws FileNotFoundException{
    VBox pn = new VBox();
    pn.setAlignment(Pos.CENTER);
    pn.setSpacing(20);
    pn.getChildren().add(new Label("Happy Ramadhan!"));

    FileInputStream fis = new FileInputStream("ramadhan.jpg");
    Image img = new Image(fis);
    ImageView iv = new ImageView(img);
    iv.setFitHeight(400);
    iv.setPreserveRatio(true);
    pn.getChildren().add(iv);

    Scene scn = new Scene(pn, 500, 500);
    stage.setTitle("Happy Ramadhan!");
    stage.setScene(scn);
    stage.setResizable(false);
    stage.show();
}

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

```



haha gotchu

Pergunakanlah slide materi kuliah sebagai bahan latihan, cobalah jalankan code yang terdapat pada slide kuliah!
Ingat, dalam JavaFX, yang paling penting adalah kemauan untuk mencoba!

Semangat :D
-DRK