

## Oberflächen mit FXML

### Grundsätzliches Vorgehen:

- *FXML*: Oberfläche bauen - am besten mit Designer (Scene builder)
- *JAVA*: Einlesen einer .fxml – Datei im Java-Code:

```
public class Main extends Application
{
    public static void main(String[] args)
    {
        launch(args);           // call start()
    }

    @Override
    public void start(Stage primaryStage) throws Exception
    {
        // root – Element, vom Typ des übergeordneten Layouts (hier GridPane)
        // Einlesen der FXML - Datei
        GridPane root = (GridPane)FXMLLoader.load(
            MainJavaFX.class.getResource("FilterPage.fxml"));
        Scene scene = new Scene(root);
        primaryStage.setScene(scene);
        primaryStage.show();
    }
}
```

### Events:

- *JAVA*: neue Klasse (z.B. ViewController) anlegen

```
public class ViewController
{
    @FXML
    private void btnNextClick(ActionEvent event)
    {
        System.out.println("test");
    }
}
```

- *FXML*: Den Controller zur FXML – Datei hinzufügen  
"Controller class = **ViewController**" (linke Spalte im Scene builder)
- *FXML*: Erstellen eines Events in FXML Datei (z.B. Button-Klick)  
"On Action = **#btnNextClick**"

### Auf Objekteigenschaften (von grafischen Objekten) zugreifen

- *FXML*: Erstellen eines Controls mit Namen (z.B. fx:id="**txtTest**")
- *JAVA*: Controller-Klasse muss Initializable implementieren

```
public class ViewController implements Initializable
{
    @FXML
    TextField txtTest;

    @Override
    public void initialize(URL url, ResourceBundle resourceBundle)
    {
    }

    @FXML
    private void btnNextClick(ActionEvent event)
    {
        txtTest.setText("test");
    }
}
```