

# LaTeX Formeln in Java

---

*Ein Beispiel basierend auf JavaFX 8 mit integrierter Zoom-Komponente*

## Inhaltsverzeichnis

<b>1. Einführung.....</b>	<b>2</b>
<b>2. Aufbau des Projekts.....</b>	<b>2</b>
<b>3. Implementierung.....</b>	<b>3</b>
3.1. Darstellung von Latex Code .....	3
3.1.1. GUI Definition .....	3
3.1.2. Implementierung in der Controller Klasse .....	4
3.1.3. Implementierung der Zoom-Funktion.....	5
3.2. JavaFX Applikation.....	5
<b>4. Weiterführende Links .....</b>	<b>6</b>

## 1. Einführung

Im nachfolgenden Dokument wird das Vorgehen beschrieben, mit welchem LaTeX-Code ansprechend in einer Java Applikation dargestellt werden kann. Diese Komponente kann somit beispielsweise sehr gut verwendet werden, um Formeln in einer Erklärkomponente darzustellen.

Des weiteren dient das Projekt als eine kleine Einführung in Verwendung des GUI Frameworks JavaFX 8.

Zudem wird erläutert, wie die LaTeX Inhalte skaliert werden können.

## 2. Aufbau des Projekts

Voraussetzung für die Applikation ist Java 8. Grund hierfür ist die Verwendung von JavaFX 8 als grundlegendes GUI Framework.

Die Paket-Struktur sieht wie folgt aus:

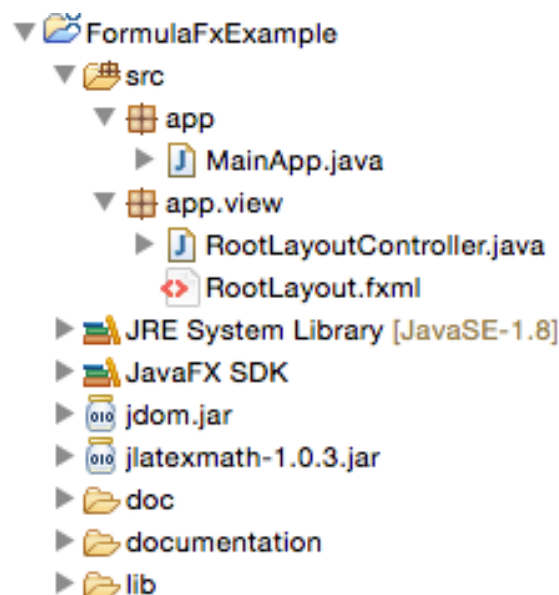


Abbildung 1: Struktur des Projekts

Das Package **app** enthält die Klasse *MainApp.java*. In dieser wird der generelle Aufbau der Applikation initialisiert, das Layout geladen und dargestellt. Der Aufbau der Klasse entspricht dem grundsätzlichen Aufbau einer Main Klasse eines JavaFX 8 Projekts.

Das Package **app.view** enthält die Definition der grafischen Oberfläche (*RootLayout.fxml*), sowie eine Controllerklasse (*RootLayoutController.java*) in der die Events der

GUI abgearbeitet und zudem die eigentliche Logik der LaTeX-Darstellung in Java realisiert ist.

Der Ordner **doc** enthält die generierte JavaDoc des Beispielprojekts. Diese kann über den Aufruf der `index.html` angesehen werden.

Der Ordner **documentation** enthält die hier vorliegende Dokumentation.

Im Ordner **lib** enthält die benötigten externen Libraries. Zur Darstellung von LaTeX-Code in Java ist eine Bibliothek notwendig. Im vorliegenden Beispiel wird die Bibliothek *JLatexMath\*.jar* verwendet. Diese hat eine weitere Abhängigkeit auf die Bibliothek *jdom\*.jar*, welche ebenfalls eingebunden ist.

## 3. Implementierung

### 3.1. Darstellung von Latex Code

Mit der hier beschriebenen Erläuterung kann LaTeX Code in einer Java Applikation angezeigt werden. Dies wird beispielhaft an der Darstellung von Text und der Mitternachtsformel erläutert.

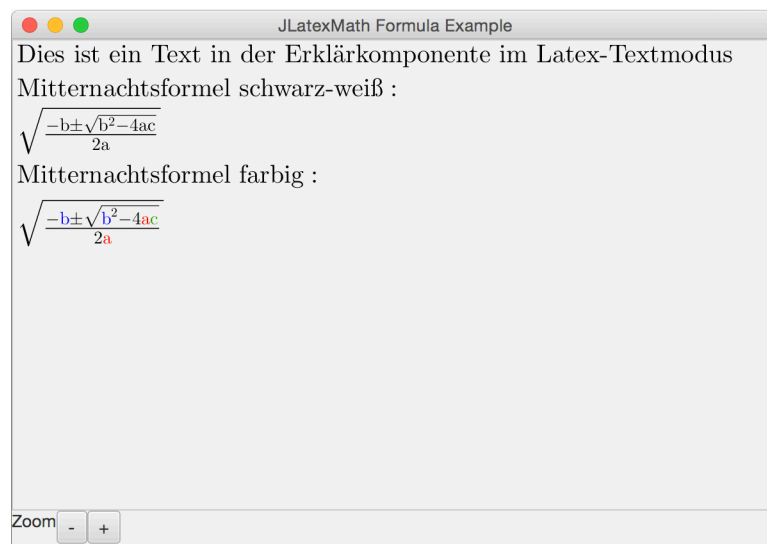


Abbildung 2: Beispielanwendung FormulaFxExample

#### 3.1.1. GUI Definition

Die LaTeX-/Formel Komponente wird in einer SwingKomponente dargestellt. Damit diese zu JavaFX kompatibel ist, wird im *RootLayoutController.fxml* eine *SwingNode* definiert, welche im Code mit Hilfe der *fx:id swingNode* angesprochen werden kann.

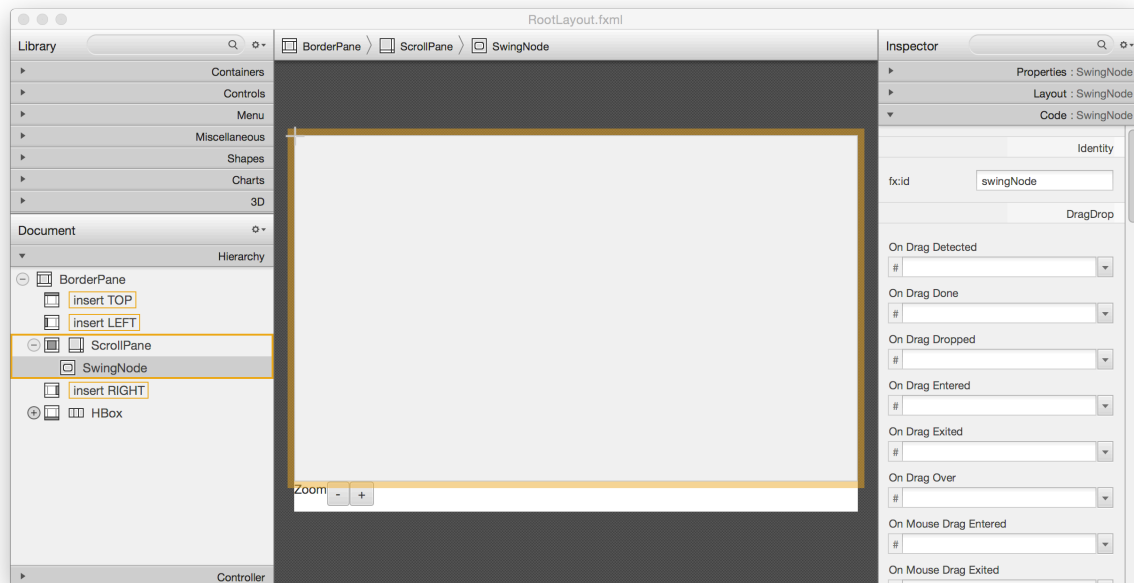


Abbildung 3: FXML Definition der GUI im SceneBuilder

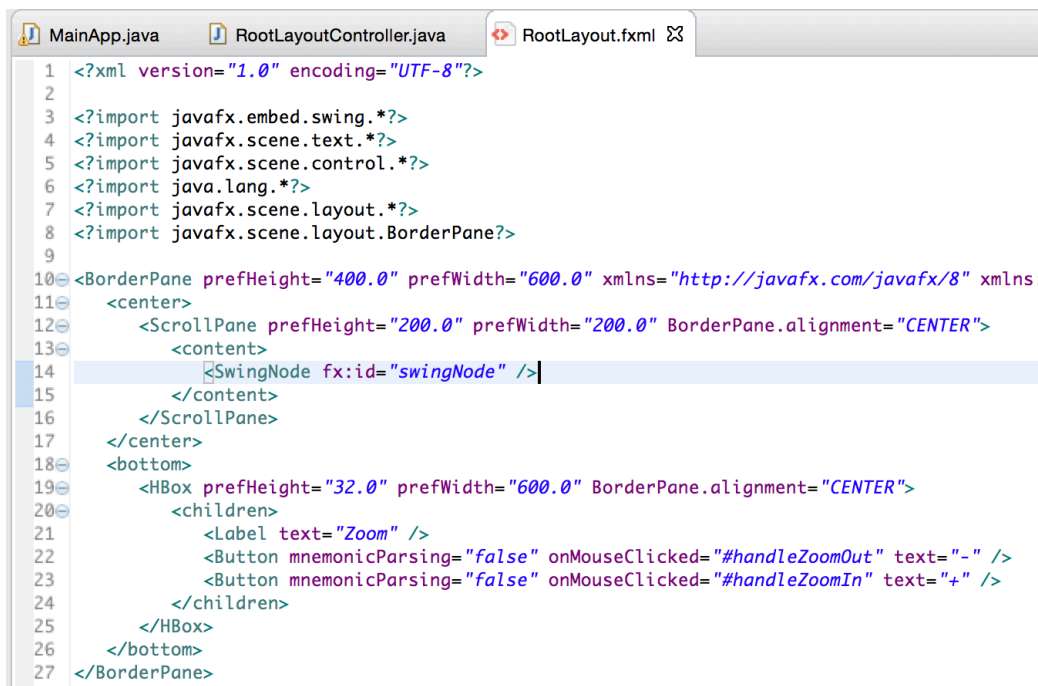


Abbildung 4: Code des FXMLs der GUI Definition

### 3.1.2. Implementierung in der Controller Klasse

In der Initialisierung des Controllers wird die Methode *showLatexFormula()* aufgerufen, mit welcher der LaTeX-Code anschließend dargestellt wird.

```

50  /**
51   * Show a sample Latex Formula in a SwingNode
52   */
53  public void showLatexFormula() {
54
55      // get a sample String with Latex Code for the Formula
56      String latexString = getSampleLatexFormula();
57
58      // generate a TexFormula using the latexString
59      TexFormula tex = new TexFormula(latexString);
60
61      // generate a new Icon using the TexFormula and setting a fontsize
62      Icon icon = tex.createTexIcon(TexConstants.ALIGN_CENTER, fontsize);
63
64      // generate a JTextPane that will be displayed in a SwingNode in JavaFX
65      JTextPane pane = getJTextPaneContainingFormula(icon);
66      swingNode.setContent(pane); // set Formula in the SwingNode on the the GUI
67
68  }

```

Abbildung 5: Codeausschnitt - Erstellung und Darstellung der LaTeX Formel

Zunächst muss der LaTeX Code implementiert werden, welcher Dargestellt werden soll. Dieser wird in einem String *latexString* gespeichert. Über diesen String *latexString* wird mit Hilfe der Klasse *TexFormula* eine LaTeX-Formel generiert. Die Klasse *TexFormula* ist in der Library *JLatexMath* enthalten. Aus dieser *TexFormula* wird mit der Methode *createTexIcon()* ein *Icon* (=Bilddatei) erstellt. Dieses wird wiederum mit Hilfe der Methode *getJTextPaneContainingFormula()* in ein *JTextPane* eingebettet. Dieses *JTextPane* wird zur Ausgabe in der JavaFX Applikation mit der Methode *setContent()* in der *swingNode* eingebettet.

### 3.1.3. Implementierung der Zoom-Funktion

Um die Größe der dargestellten Inhalte skalieren zu können, wird das *Icon* mit einer definierten Schriftgröße *fontsize* generiert. Die *fontsize* kann mit Hilfe der + und – Buttons in der GUI skaliert werden. Das jeweilige *onMouseClicked*-Event ruft die Methoden *handleZoomIn()* und *handleZoomOut()* auf. In diesen Methoden wird die Schriftgröße angepasst und anschließend die Grafik mit der *TexFormula* mit Hilfe der neuen Schriftgröße neu generiert und in der GUI gesetzt.

## 3.2. JavaFX Applikation

Der allgemeine Aufbau einer JavaFX Applikation ist nicht Teil dieser Dokumentation, kann aber anhand des Beispiels einfach nachvollzogen werden. Im Anhang wird auf ein ausführliches und gutes Beispiel zu JavaFX 8 verlinkt.

## 4. Weiterführende Links

Anbei ist eine Liste mit weiterführenden Links aufgeführt, welche einen vertieften Einblick ermöglichen:

Homepage des JLatexMath Projekts:

<http://forge.scilab.org/index.php/p/jlatexmath/>

Weitere Beispiel des JLatexMath Projekts:

<http://forge.scilab.org/index.php/p/jlatexmath/source/tree/master/examples/Basic>

Alternative zu JLatexMath: JMathTeX

<http://jmathtex.sourceforge.net>

Tutorial zu JavaFX:

<http://code.makery.ch/library/javafx-8-tutorial/>