
Inhaltsverzeichnis

1	Einleitung & Grundlagen	1
1.1	Was behandeln wir in dem einleitenden Kapitel?	1
1.2	Die Welt von Java und JavaFX	2
1.2.1	Was ist Java?	2
1.2.2	Etwas zur Historie von Java	3
1.2.3	Wo findet Java heutzutage Anwendung?	5
1.2.4	Die Java-Plattform	6
1.3	Was ist JavaFX?	7
1.3.1	Die Architektur	8
1.3.2	JavaFX ohne Java programmieren zu müssen	9
1.4	Was benötigen Sie?	10
1.4.1	Die Hardware und das Betriebssystem	10
1.4.2	Die Java-Basisumgebung	11
1.4.3	Integrierte Entwicklungsumgebungen für JavaFX	13
1.4.4	Der Scene Builder	15
1.5	Ein erstes JavaFX-Beispiel mit dem Scene Builder	16
1.5.1	Ein erster Blick auf den Scene Builder	17
1.5.2	Bauen wir uns ein JavaFX-Programm	19
1.6	Ein erstes JavaFX-Programm mit NetBeans	21
1.6.1	Eine echte JavaFX Application erzeugen	23
1.6.2	Ein Blick auf die generierten Code-Strukturen	27
1.7	Eine JavaFX-Applikation mit Eclipse	29
1.7.1	Die JavaFX-Applikation	30
1.7.2	Eine JavaFX-Applikation nur mit dem JDK und einem Editor	33
1.8	Verteilen und Ausführen von JavaFX-Applikationen	38
1.8.1	Erstellen von JAR-Dateien	39
1.8.2	Java Web Start und JNLP	41
1.8.3	Mit NetBeans die Java-Applikation zum Web Start fertig machen	46
1.8.4	JavaFX-Applets	49

2	Hilfe und Basiswissen	51
2.1	Was behandeln wir in diesem Kapitel?	51
2.2	Die Dokumentationsseiten von Oracle	51
2.2.1	Die API-Dokumentation	52
2.3	Standardbeispiele	54
2.4	Support durch Ihre IDE	55
2.5	Wichtige grundlegende Regeln für Java	59
2.6	CSS – Style Sheets für das bessere Aussehen	60
2.6.1	Regeln, immer wieder Regeln	60
3	FXML und der JavaFX Scene Builder	63
3.1	Was behandeln wir in diesem Kapitel?	63
3.2	Eine JavaFX-Applikation mit NetBeans erstellen	63
3.2.1	Eine FXML-Applikation anlegen	64
3.2.2	Die Struktur eines FXML-Projekts	64
3.3	Das Konzept einer grafischen Oberfläche – die Bedeutung von Bäumen und der JavaFX Scene Graph	68
3.3.1	Komponenten	69
3.3.2	Fenster und Container	69
3.4	Container in FXML	69
3.4.1	Das AWT – Großvater GUI	70
3.4.2	Das Konzept der Layoutmanager	71
3.4.3	Swing	75
3.4.4	Das grundsätzliche GUI-Konzept bei JavaFX	76
3.5	Die verschiedenen Container bei JavaFX	76
3.5.1	Absolute Layouts – AnchorPane und Pane	77
3.5.2	Accordion und TitlePane	78
3.5.3	Der BorderPane-Container	78
3.5.4	Fließende Inhalte – FlowPane, HBox und VBox	79
3.5.5	Tabellen beziehungsweise Gitter – der GridPane-Container	81
3.5.6	Stapelverarbeitung mit StackPane	83
3.5.7	Scrollbare Bereiche – ScrollPane	83
3.5.8	Gesplittete Ansichten – SplitPane	84
3.5.9	Tab-Strukturen – TabPane und Tab	85
3.5.10	Ein TilePane	85
3.5.11	Platz da – TitledPane	85
3.5.12	Die Toolbar	86
3.6	Die Controls	87
3.6.1	Beginn eines Praxisprojekts – ein Taschenrechner	89
3.6.2	Ein weiteres Praxisprojekt – ein Bildbetrachter	91

3.7	Grafik und Zeichnen unter JavaFX	94
3.7.1	Allgemeines zum Zeichnen und Malen auf grafischen Java-Oberflächen	95
3.8	Java-2D und dessen Erbe in JavaFX – gutes Aussehen, Effekte, komplexe Formen und Transformationen	97
3.8.1	Wer hat's erfunden?	97
3.8.2	Der Koordinatenraum und Transformationen	98
3.8.3	Füllungen und Rahmen	98
3.8.4	Transparenz	101
3.8.5	Diagramme – Charts	102
4	Gestalten mit dem Inspector des JavaFX Scene Builders	105
4.1	Was behandeln wir in dem Kapitel?	105
4.2	Grundsätzliche Konfigurationen im Inspector	105
4.3	Das Properties-Register	106
4.3.1	Texteigenschaften	109
4.3.2	Knoten und Extras	109
4.3.3	Effekte	109
4.4	JavaFX CSS zur Gestaltung	111
4.4.1	Selektoren	112
4.4.2	Die spezifischen JavaFX-Bezeichner	112
4.4.3	Das Default Style Sheet	113
4.4.4	Eigene Style Sheets und die Anwendung im Scene Builder	115
4.5	Die Layout-Kategorie im Inspector	118
4.6	Weiterentwicklung der Praxisprojekte	119
4.6.1	Taschenrechner 2.0	119
4.6.2	Bildbetrachter 2.0	121
5	Behind the scene – der Aufbau von FXML	123
5.1	Was behandeln wir in diesem Kapitel?	123
5.2	Warum FXML?	123
5.3	XML-Grundlagen	124
5.3.1	Wo kommt XML her und was macht man damit?	124
5.3.2	Erstellen und Anzeigen von XML-Dokumenten	124
5.3.3	Der Aufbau von XML-Dokumenten	125
5.3.4	Bestandteile eines XML-Dokuments	126
5.3.5	Die Syntax eines XML-Dokuments – Wohlgeformtheit	130
5.4	Die Struktur von FXML	132
5.4.1	FXML-Elemente	132
5.4.2	FXML-Attribute	135
5.4.3	Static Properties – Statische Eigenschaften	138
5.5	Weiterentwicklung des Bildbetrachters	138

6	JavaScript und der JavaFX Scene Builder	143
6.1	Was behandeln wir in diesem Kapitel?	143
6.2	Einige Hintergründe zu JavaScript	143
6.2.1	JavaScript im Web	144
6.2.2	Syntax und Aufbau	144
6.2.3	Funktionen	147
6.2.4	JavaScript und Objekte	149
6.3	Ereignisbehandlung	150
6.3.1	Vorüberlegungen zur Ereignisbehandlung	150
6.3.2	Ereignisse und die Reaktion darauf	150
6.3.3	Das Code-Register im Inspector	151
6.3.4	Reaktion im Java-Controller	151
6.3.5	Mit Skriptsprachen reagieren	154
6.4	Weiterentwicklung des Taschenrechners	160
7	Java – die Schnellbahn	165
7.1	Was behandeln wir in diesem Kapitel?	165
7.2	Das JavaFX-API	165
7.2.1	Die wichtigsten Pakete	166
7.3	Die grundsätzliche Java-Struktur einer JavaFX-Applikation	167
7.3.1	Das Hauptprogramm – ein Objekt vom Typ <code>javafx.application.Application</code>	167
7.4	Die Layout-Panes	169
7.4.1	Elemente in Panes	170
7.5	Die Größe und Position von Elementen	172
7.6	Style Sheets in Java verwenden	172
7.6.1	Vorgabeklassen und keine Vorgabeklassen	173
7.6.2	Zuordnung von Style Sheet-Klassen	173
7.7	Die JavaFX UI Controls	174
7.7.1	Ein praktisches Beispiel für eine GUI	174
7.8	Funktionalität hinzufügen – Überlegungen und Hintergrundinformationen zum Eventhandling	176
7.8.1	Aktions-Setter	177
7.8.2	Die EventHandler-Schnittstelle und die <code>handle()</code> -Methode	177
7.8.3	Dem Beispiel Funktionalität hinzufügen	179
7.8.4	Eventhandling nur mit Java	182
7.8.5	Einen Event-Controller verwenden	184
7.8.6	Weiterentwicklung des Bildbetrachters	184
7.9	Umgang mit Canvas-Elementen	192
7.9.1	Der Grafikkontext	193
7.9.2	Das Koordinatensystem und die möglichen Ausgabemethoden	193

7.10	Preloader	196
7.10.1	Einen Preloader mit NetBeans erstellen	197
7.10.2	Einen Preloader verwenden	199
8	Ereignisse und mehr	201
8.1	Was behandeln wir in diesem Kapitel?	201
8.2	Ereignisbehandlung	201
8.2.1	Die JavaFX- Events	202
8.2.2	Noch tiefere Hintergründe – vom Blubbern, Binden, Callbacks und Triggern	202
8.2.3	Die Auswahl des Ziels	204
8.2.4	Das Auffangen oder Filtern des Ereignisses	205
8.3	Konkrete Reaktionen auf Ereignisse	206
8.3.1	Eventhandler	207
8.3.2	Einige praktische Beispiele	211
8.3.3	Eventfilter	217
8.4	Die Controllerklasse bei FXML	220
8.4.1	Details zur Controllerklasse	220
8.5	JavaFX Properties und Binding	225
8.5.1	Abhängigkeiten und das API	225
8.5.2	Definition und Namenskonventionen bei Properties	226
8.5.3	Listener bei Properties	228
8.5.4	Anwendung des High-Level Binding API	230
9	JavaFX und Swing	237
9.1	Was behandeln wir in diesem Kapitel?	237
9.2	JavaFX in Swing	237
9.2.1	Synchronisation und fremde Welten	238
9.2.2	Die Methoden <code>runLater()</code> und <code>invokeLater()</code>	238
9.2.3	Ein Beispiel für die Integration von JavaFX in Swing	239
9.3	Swing in JavaFX	244
10	HTML-Content	247
10.1	Was behandeln wir in diesem Kapitel?	247
10.2	HTML-Content in JavaFX-Applikationen	247
10.2.1	Das eingebettete Browser-API	247
10.3	Hyperlinks und Adresszeilen	250
11	Diagramme	255
11.1	Was behandeln wir in diesem Kapitel?	255
11.2	Grundlagen	255
11.3	Einführung in JavaFX Charts	255

11.3.1	Die verfügbaren Typen	256
11.3.2	Die Datenbasis	257
11.4	Diagramme ohne Achsen	257
11.4.1	Ein praktisches Beispiel für ein Kuchendiagramm mit Java	257
11.4.2	Konfiguration und Einstellungen von Diagrammen	258
11.4.3	Das praktische Beispiel für ein Kuchendiagramm mit FXML	261
11.5	Diagramme mit Achsen	263
11.5.1	Datenserien	263
11.5.2	Ein praktisches Beispiel für ein Diagramm mit Achsen: ein Balkendiagramm	263
11.6	Dynamik und Ereignisbehandlung in Diagrammen	267
11.6.1	Dynamisches Kuchendiagramm – Teil 1	267
11.6.2	Datenpunkte aktualisieren – die Methode set()	273
11.6.3	Ereignisverarbeitung für Diagrammelemente	282
11.6.4	Ein Beispiel für ein Kuchendiagramm mit der Behandlung von Ereignissen	282
12	Spezialitäten	289
12.1	Was behandeln wir in diesem Kapitel?	289
12.2	JavaFX Collections	289
12.2.1	Grundlagen zu dynamischen Datenstrukturen	290
12.2.2	Details zu JavaFX Collections	293
12.2.3	Beispiel 1 – eine dynamische Liste zur Erweiterung von Inhalten in einem mehrzeiligen Textfeld	293
12.2.4	Beispiel 2 – dynamisch Steuerelemente hinzufügen	297
12.2.5	Beispiel 3 – Verwenden einer ObservableMap	300
12.2.6	Beispiel 4 – die Klassenmethoden von FXCollections nutzen	303
12.3	Multithreading	306
12.3.1	Threads laufen lassen – run() und start()	306
12.3.2	Threads abbrechen	307
12.3.3	Multithreading in JavaFX – javafx.concurrent	307
12.4	Drag & Drop	326
12.4.1	Hintergrundinformationen	327
12.4.2	Ein einfaches Drag & Drop-Beispiel	327
13	OOP- und Java-Crashkurs	333
13.1	Syntaxfragen	333
13.1.1	Datentypen	333
13.2	Java und die OOP	334
13.2.1	Kernkonzepte der OOP	334
13.2.2	Objekte und Klassen	335
13.2.3	Identifizieren Sie sich – Botschaften	337
13.2.4	Klassen	338

13.3 Variablen und Eigenschaften	339
13.3.1 Deklaration von Variablen und Eigenschaften	340
13.3.2 Die Methodendeklaration	340
13.3.3 Konstruktoren und Destruktoren – das Speichermanagement	341
13.4 Lokale und anonyme Klassen	343
13.4.1 Klassendeklarationen in Methoden	343
13.4.2 Anonyme Klassen	344
13.5 Pakete und die import-Anweisung	345
13.5.1 Die Zuordnung einer Klasse zu einem Paket und das Default-Paket	345
13.5.2 Die Suche nach Paketen	346
13.5.3 Die import-Anweisung	347
13.6 Vererbung	348
13.6.1 Superklasse und Subklasse	348
13.6.2 Die technische Umsetzung einer Vererbung	349
13.7 Überschreiben und Überladen	350
13.8 Information Hiding und Zugriffsschutz	351
13.8.1 Indirekte Zugriffe über Getter und Setter	352
13.9 Abstrakte Klassen und Schnittstellen	352
13.9.1 Was ist eine abstrakte Klasse?	352
13.9.2 Was ist eine Schnittstelle?	353
Sachverzeichnis	355