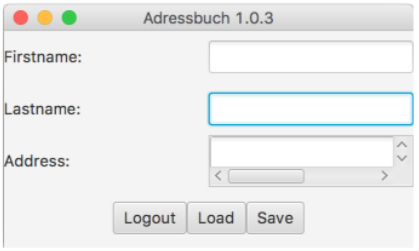
GUI

1. Welche Layout-Manager würden Sie verwenden, um das folgende Fenster zu realisieren (Abbildung 1)? Zeichnen Sie die Layout-Manager direkt in die Abbildung. Tipp: Folgende Layout-Manager wurden in der Vorlesung besprochen: BorderPane, HBox, VBox, StackPane, GridPane, FlowPane, TilePane.



Hier bietet sich an, ein GridPane zu verwenden. Diesen könnte man zudem in ein BorderPane integrieren.

2. Eventhandler

a) Geben Sie ein Beispiel für die zwei Formen von JavaFX Eventhandlern (alt und seit Java8)

```
btn1.setOnAction(e -> Event)
Button btn2 = new Button("3");
btn3.setOnAction(event -> btn2.setText("Active"));

Button btn3 = new Button("Bye");
btn1.setOnAction(new EventHandler<ActionEvent>() {
@Override
public void handle(ActionEvent event) {
btn3.setText("Active");
}
});
```

b) Welche Möglichkeit gibt es in JavaFX, nach einer bestimmten Zeit einen Handler-Callback zu bekommen? Wozu könnte man das verwenden?

Eine mögliche Verwendung könnte so aussehen, dass man im Hintergrund Datenbankabfragen laufen lässt um den GUI Thread dann zu informieren, wenn die Daten geladen wurden.

3. GUI: 1. Zeichnen Sie das Fenster mit Inhalt, das durch den JavaFX Code beschrieben wird. 2. Was macht der Handler?

```
public class LayoutExample extends Application
{
private TextField inputArea = new TextField();
private TextArea outputArea = new TextArea();
public static void main(String[] args)
{ Application.launch(args); }
@Override
public void start(Stage stage)
{
Label headerLbl = new Label("Please insert Message in
TextArea!");
Label inputLbl = new Label("Input: ");
Label outputLbl = new Label("Output: ");
Button okBtn = new Button("OK");
HBox output = new HBox();
output.getChildren().addAll(outputLbl, outputArea);
```

```
okBtn.addEventHandler(MouseEvent.MOUSE_CLICKED,
event -> outputArea.appendText("You: " +
inputArea.getText() + "\n"));
BorderPane root = new BorderPane();
root.setTop(headerLbl);
root.setRight(okBtn);
root.setBottom(output);
root.setLeft(inputLbl);
root.setCenter(inputArea);
Scene scene = new Scene(root);
stage.setScene(scene);
stage.setTitle("SE2 Nachdenkzettel GUI");
stage.show();
}
}
```

1.

2. Der Handler “kümmert” sich um die Eingaben des Users per bspw. Tastatur oder Maus.


DocScape – publizieren mit Zukunft

6 + + + einfach in der Anwendung

DocScape bringt Ihre Kommunikationsprojekte nach vorne – individuell, innovativ, international

DocScape ist eine regelbasierte Software, die auf vorhandene Datenbanken oder Warenwirtschaftssysteme in Ihrem Unternehmen zugreift.

Ihr Corporate Design für Dokumente ist nichts anderes als eine umfangreiche und präzise Sammlung an Layoutregeln. DocScape übersetzt diese Gestaltungsregeln in eine XML-Notation und führt sie „auf Knopfdruck“ exakt und vollautomatisch aus.

Die Arbeit mit DocScape bedeutet: Änderungen einer Designregel werden in allen Dokumenten und Dokumenttypen automatisch umgesetzt. Auch ältere Versionen von Dokumenten werden von diesen Änderungen erfasst. Damit ist DocScape prädestiniert, die dynamische und nicht zuletzt durch die Expansion in neue Märkte getriebene Entwicklung eines Corporate Designs nachhaltig zu unterstützen. Die Arbeit mit DocScape verhindert darüber hinaus Flüchtigkeitsfehler oder das „bewusste“ Übersehen von CD-Regeln.

Sobald Sie „den Knopf“ drücken und DocScape starten, werden die relevanten Daten (Bilder, Grafiken, Texte, etc.) abgerufen und zu 100% gemäß dem hinterlegten Regelwerk auf den Seiten platziert – inklusive Schrift- und Tabellensatz und bei Einhaltung aller definierten Stilvorlagen, in jeder beliebigen Sprache (wie z. B. Chinesisch, Arabisch, Japanisch oder Russisch). Ist etwa ein Bildstandsabgleich gewünscht, platziert DocScape die Bilder erst, nachdem alle relevanten Sprachen auf ihre Lauflänge hin überprüft und optimiert wurden. Innerhalb kürzester Zeit entsteht so das komplette Print-Produkt.

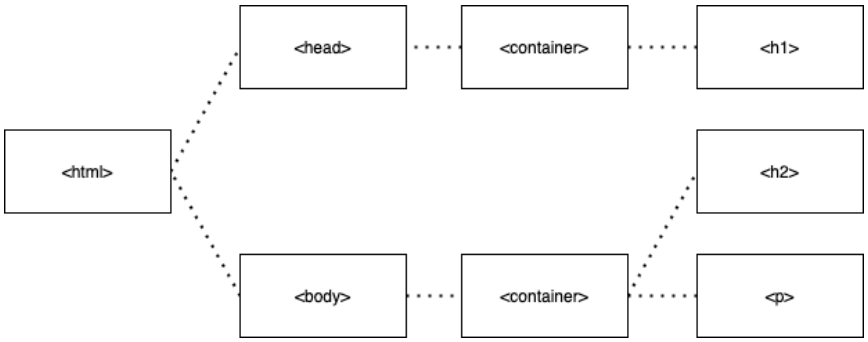
DocScape erzeugt auf einem gewöhnlichen Rechner (z. B. 1 GHz-Prozessor, 1 GB Speicher) etwa 50 bis 60 Seiten pro Minute! Das Programm arbeitet folglich so schnell, dass Sie selbst kurz vor der Drucklegung noch problemlos auch hochkomplexe Dokumente vollständig neu erzeugen können. Damit reduzieren Sie Fehlerquellen und Prozesskosten.

Mit DocScape ist auch die Produktion personalisierter Medien unkompliziert und wirtschaftlich möglich. Ob zielgruppenspezifischer Produktkatalog, individuelle Kundenzeitschrift oder persönliches Angebot – mit DocScape liefern Sie exakt diejenigen Informationen, die Ihre Zielgruppen oder Kunden wünschen. Besser lässt sich Kundenbindung nicht umsetzen. DocScape eröffnet Ihnen neue Dimensionen für das Direct-to-Customer-Marketing!

Sollen individuell gestaltete Seiten oder Objekte aus gängigen Anwendungen, wie z.B. InDesign, eingepflegt werden, ist auch dies problemlos möglich.

4. Strukturen: Beschreiben Sie das Dokument einmal als Baum aus graphischen Nodes und einmal in Form eines serialisierten Textes (wie Html/xml). Die Tags dafür können Sie frei erfinden

```
<html>
<head>
...
</head>
<body>
<container>
<h1> ... </h1>
<img> ... </img>
</container>
<container>
<h2> ... </h2>
<p> ... </p>
</container>
</body>
</html>
```



5. GUI Thread und andere…
a) Sie wollen Ihre JavaFX Application Unit Testen. Was für ein Problem tritt auf?
Tipp:
https://medium.com/information-and-technology/test-driven-development-in-javafx-with-testfx-66a84cd561e0
Die JavaFx- Library ist nicht in Java standardmäßig enthalten und muss deshalb korrekt importiert werden. Damit die Unit-Tests funktionieren benötigt man also zusätzlich die TestFX Library.

b) Sie müssen Dinge im Background machen und können den Main GUI Thread nicht dafür nehmen? Sie müssen Daten zwischen GUI und Restapplikation austauschen?
→ Was macht die Task Class in JavaFX?
Tipp: https://docs.oracle.com/javafx/2/threads/jfxpub-threads.htm
die Task-Klasse…
...implementiert die Programmlogik, die im Hintergrund-Thread laufen muss.
...erbt die Call-Methode
...versichert, dass alle mit JavaFX zusammenhängende Geschehnisse im JavaFX-Thread geschehen

```
Task starten: Thread th = new Thread(task);
th.setDaemon(true);
th.start();
```