



Übung 10 – Web GUIs Teil 2

Beschreibung

Wir verwenden das HTML der letzten Übung um eine kleine lauffähige Web-Version des Formulars zu erzeugen. Als Framework verwenden wir dazu Angular.

Vorbereitung – Startprojekt

Öffnen und starten Sie das Starter-Projekt.

Schritte:

1. Öffnen Sie das Projekt *u10-web2-starter* in der IDE
2. Öffnen Sie eine Konsole im Ordner des Projekts
3. Starten sie den Webserver mit
`npm install`
dann
`npm run ng serve`
4. Starten sie einen Browser und öffnen sie die Anwendung (die URL steht in der Konsole, normalerweise `http://localhost:4200`)
5. Ändern Sie den Text in der Datei `app.component.html`
6. Speichern Sie und beobachten Sie das Resultat im Browser

Aufgabe 1 – Event Form

Erzeugen Sie eine neue Komponente für das Formular (`EventForm`).

Schritte:

1. Erzeugen Sie die folgenden drei Dateien im Ordner `src/app`:
 - a. `eventform.component.css`
 - b. `eventform.component.html`
 - c. `eventform.component.ts`
2. Kopieren Sie die Styles der letzten Übung (`web1.css`) in die neue CSS Datei.

3. Kopieren Sie die HTML-Inhalte des Formulars (`web1.html`) in die neue HTML-Datei.
Hinweis: Nur das benötigte `<div>` mit Inhalt, nicht den Rahmen der HTML Datei (kein `<head>`, `<body>`, `<script>`....).
4. Füllen Sie ggf. die Datei `eventform.component.ts` mit einer passenden leeren Komponente (siehe Folien).
5. Sie müssen jetzt noch Ihre Komponente registrieren. Machen Sie das, indem Sie das Formular analog zur `AppComponent` in der Datei `app.module.ts` aufnehmen.
Hinweis: Import nicht vergessen und die einzelnen Komponenten kommasepariert in `declarations` aufnehmen.
6. Verwenden Sie die Form in der `AppComponent`.

Aufgabe 2 – Data & event binding

Speichern Sie die Eingabe des Benutzers in einer Modelklasse `EventModel` und binden Sie einen Click-Handler an.

Schritte:

1. Verwenden Sie die Klasse `EventModel` im Code der `EventFormComponent` (Import nicht vergessen)
2. Erzeugen Sie eine neue Instanz der Klasse
3. Binden Sie mind. 3 der Eingabefelder (Text) an die Attribute des Models
4. Geben Sie bei einem Click auf den „Save“-Button das Model aus

Aufgabe 3 – Komponenten

Machen Sie aus dem Formular eine wiederverwendbare Komponente mit Ein-/Ausgabe. Verschicken sie beim Click auf den Save-Button ein Event an die Elternkomponente und behandeln Sie das Event dort, statt direkt im Formular.

Schritte:

1. Importieren Sie in der `EventFormComponent` die benötigten Angular-Elemente:

```
import { Component, EventEmitter, Output } from '@angular/core';
```
2. Führen Sie eine `@Output()` Property vom Typ `EventEmitter<EventModel>` ein.
3. Erzeugen Sie eine Instanz.
4. Bauen sie die bisherige `onSave` Methode so um, dass sie selbst nichts mehr ausgibt, sondern ein Event verschickt.
5. Führen sie in der `AppComponent` eine Methode ein die etwas ausgibt.
6. Passen Sie das Template der `AppComponent` so an, dass bei Events vom Formular die neue Methode in der `AppComponent` ausgeführt wird