UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering

Program Systems Development practice

Practice 10

Angular 2

Data binding

- Synchronization of data between the Model and the View
  - Display attributes of a component in a template with the **`{{variable}}`** sign
  - To change the value of an attribute use the *Two-way data binding*
    - **`[(ngModel])="variable"`**, where the **`variable`** is an attribute in the component
      - `It works only with the FormsModule.`

Dependency injection

- It's a software design pattern that implements inversion of control for resolving dependencies
- A dependency is an object that can be used (e.g.: a service)
- An injection is the passing of a dependency to a dependent object (e.g.: to the client) which uses it

Providers

- A provider describes what the injector should instantiate a given token.
- Mainly the Services are signed with the `@Injectable()` decorator to make the class available to the injector for instantiation.

HTTP Requests - Observable

- import {Http} from '@angular/http';
- A http.get returns an RxJS Observable object.
  - Observable helps to manage the asnychronous data flows
  - An Observable is like a stream. It handles series of events (can be 0, 1, 2, more).
- At the end of the request an HTTP response is received, which is processed in an asynchronous way
  - .map() (in a Service)
  - .subscribe() (in a Component)

HTTP Requests - Promise

- import {Http} from '@angular/http';
- A http.get can return a Promise object as well.
  - Promise also helps to manage asynchronous operations.
  - Promise handles a single event.
- Since it makes the **`subscribe()`** in the Service, the Component processes the data with **`.then()`**.