Exercise (Instructions): Angular Template-driven Forms

Objectives and Outcomes

In this exercise you will learn about using Angular template-driven forms in your Angular application. At the end of this exercise you will be able to:

* Design a template-driven form in your Angular application.

1. Create a new angular project using ng new LoginProject or using existing project

2. Create a new component login:

ng generate component login

3. In the login.component.html, create a login form

(bootstrap template for login form can be found here <https://getbootstrap.com/docs/4.0/components/forms/>)

<div class="component"  >

<form novalidate class="text-center border border-light p-5"

    #loginForm="ngForm" (ngSubmit)="OnSubmit()">

    <p class="h4 mb-4">Sign in</p>

    <input class="form-control mb-4" type="text" placeholder="User Name" name="username"

         [(ngModel)]="user.username" #username="ngModel" required/>

    <input class="form-control mb-4" type="password" placeholder="Password" name="password"

        [(ngModel)]="user.password" #password="ngModel" required/>

    <button class="btn" type="submit" [disabled]="loginForm.form.invalid">Login</button>

    <br>

</form>

</div>

4. Add Boostratp to index.html

<https://getbootstrap.com/docs/4.0/getting-started/introduction/>

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <title>LoginProject</title>

  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/x-icon" href="favicon.ico">

  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

</head>

<body>

  <app-root></app-root>

</body>

</html>

5. Make sure you add bootstrap component and import FormsModule to app.module.ts

import { BrowserModule } from '@angular/platform-browser';

import { NgModule } from '@angular/core';

import { AppComponent } from './app.component';

import { LoginComponent } from './login/login.component';

import {FormsModule} from '@angular/forms';

@NgModule({

  declarations: [

    AppComponent,

    LoginComponent

  ],

  imports: [

    BrowserModule, FormsModule

  ],

  providers: [],

  bootstrap: [AppComponent, LoginComponent]

})

export class AppModule { }

Change the login.component.ts as following:

import { Component, OnInit } from '@angular/core';

@Component({

  selector: 'app-login',

  templateUrl: './login.component.html',

  styleUrls: ['./login.component.css']

})

export class LoginComponent implements OnInit {

  user =  {username:'', password:'', remembered:false};

  label = "";

  constructor() { }

  ngOnInit(): void {

  }

  OnSubmit(){

    if(this.user.username=="admin" && this.user.password=="admin"){

      this.label = "Success";

       return true;

    }

    else {

      this.label="Failed";

      return false;

    }

  }

}

Next, add the error message to the login.component.html

<div class="component"  >

<form novalidate class="text-center border border-light p-5"

    #loginForm="ngForm" (ngSubmit)="OnSubmit()">

    <p class="h4 mb-4">Sign in</p>

    <input class="form-control mb-4" type="text" placeholder="User Name" name="username"

         [(ngModel)]="user.username" #username="ngModel" required/>

    <p \*ngIf="username.errors?.required"  class="alert alert-danger">Please enter user name:</p>

    <input class="form-control mb-4" type="password" placeholder="Password" name="password"

        [(ngModel)]="user.password" #password="ngModel" required/>

        <p \*ngIf="password.errors?.required" class="alert alert-danger">Please enter password:</p>

    <button class="btn" type="submit" [disabled]="loginForm.form.invalid">Login</button>

    <br>

    <p \*ngIf="label">{{label}}</p>

</form>

</div>

