

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра інформаційних систем та мереж

Звіт
до лабораторної роботи №6
з дисципліни «Екстримальне програмування»
на тему «Створення калькулятора використовуючи Angular»

Виконала
студентка групи КН-311
Мельничук М.П.

Прийняв
Щербак С.С.

Львів-2020

Мета: створення калькулятора використовуючи Angular фреймворк.

Теоретичні відомості

Angular (зазвичай так називають фреймворк Angular 2 або Angular 2+, тобто вищі версії) — написаний на TypeScript front-end фреймворк з відкритим кодом, який розробляється під керівництвом Angular Team у компанії Google, а також спільнотою приватних розробників та корпорацій. Angular — це AngularJS, який переосмислили та який був повністю переписаний тією ж командою розробників.

Спочатку переписаний AngularJS отримав назву Angular 2 від команди розробників, яка над ним працювала, але це призвело до плутанини серед інших розробників. Аби пояснити різницю між ними та наголосити, що це окремі проекти, команда вирішила для фреймворків версій 1.X застосовувати назву AngularJS, а для версій, починаючи з 2.0, — Angular без JS.

Як згадано вище, Angular — це ретельно переписаний AngularJS.

- Додано Angular CLI, що дає змогу розпочати створення нового додатка, просто написавши команду `ng new [app name]`
- Angular не використовує концепцію "області видимості" або контролерів, натомість як головну архітектурну концепцію він застосовує ієрархію компонентів
- Angular має інакший синтаксис написання виразів, застосовуючи "[]" для біндингу даних властивостей, і "()" для біндингу даних івентів
- Модульність – значна частина основного функціоналу перенесена у модулі
- Angular рекомендує та застосовує розроблену Microsoft мову — TypeScript, що містить такі можливості, як:
 - Класи, а отже Об'єктно-орієнтоване програмування
 - Система типізації
 - Узагальнене програмування

TypeScript — надмножина ECMAScript 6 (ES6), і є зворотно сумісним зі стандартом ECMAScript 5 (тобто JavaScript). Angular також має такі ES6-можливості, як:

- Анонімні функції
- Ітератори
- Цикли типу For/Of
- Python-подібні генератори
- Рефлексія

Динамічне завантаження

Асинхронна компіляція шаблонів

Заміна контролерів та $\$score$ (області видимості) компонентами та директивами – компонент є директивою з шаблоном

Ітеративні колбеки завдяки використанню RxJS. RxJS дещо обмежує видимість станів та можливості дебагінгу, але, застосовуючи такі плагіни, як ngReact та ngrx, це легко вирішується.

Хід роботи

1. Спочатку встановимо Angular/cli:

```
D:\Навчання\6 sem\ЕП\my_first_angular_project>npm install -g @angular/cli
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
C:\Users\Marynka\AppData\Roaming\npm\ng -> C:\Users\Marynka\AppData\Roaming\npm\node_modules\@angular\cli\bin\ng
> @angular/cli@9.1.7 postinstall C:\Users\Marynka\AppData\Roaming\npm\node_modules\@angular\cli
> node ./bin/postinstall/script.js

? Would you like to share anonymous usage data with the Angular Team at Google under
Google's Privacy Policy at https://policies.google.com/privacy? For more details and
how to change this setting, see http://angular.io/analytics. No
+ @angular/cli@9.1.7
added 271 packages from 206 contributors in 140.677s
```

Рис.6.1 Встановлення Angular/cli

2. Створимо новий проект my-calculator за допомогою Angular CLI:

```

C:\Users\Marynka\IdeaProjects\my_first_angular_project>ng new my-calculator
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE my-calculator/angular.json (3622 bytes)
CREATE my-calculator/package.json (1246 bytes)
CREATE my-calculator/README.md (1029 bytes)
CREATE my-calculator/tsconfig.json (489 bytes)
CREATE my-calculator/tslint.json (3125 bytes)
CREATE my-calculator/.editorconfig (274 bytes)
CREATE my-calculator/.gitignore (631 bytes)
CREATE my-calculator/browserslist (429 bytes)
CREATE my-calculator/karma.conf.js (1025 bytes)
CREATE my-calculator/tsconfig.app.json (210 bytes)
CREATE my-calculator/tsconfig.spec.json (270 bytes)
CREATE my-calculator/src/favicon.ico (948 bytes)
CREATE my-calculator/src/index.html (298 bytes)
CREATE my-calculator/src/main.ts (372 bytes)
CREATE my-calculator/src/polyfills.ts (2835 bytes)
CREATE my-calculator/src/styles.css (80 bytes)
CREATE my-calculator/src/test.ts (753 bytes)
CREATE my-calculator/src/assets/.gitkeep (0 bytes)
CREATE my-calculator/src/environments/environment.prod.ts (51 bytes)
CREATE my-calculator/src/environments/environment.ts (662 bytes)
CREATE my-calculator/src/app/app-routing.module.ts (246 bytes)
CREATE my-calculator/src/app/app.module.ts (393 bytes)
CREATE my-calculator/src/app/app.component.html (25757 bytes)
CREATE my-calculator/src/app/app.component.spec.ts (1080 bytes)
CREATE my-calculator/src/app/app.component.ts (217 bytes)
```

Рис.6.2 Створення проекту

3. Створимо нову компоненту calculator:

```
C:\Users\Marynka\IdeaProjects\my_first_angular_project\my-calculator>ng g c calculator
CREATE src/app/calculator/calculator.component.html (25 bytes)
CREATE src/app/calculator/calculator.component.spec.ts (656 bytes)
CREATE src/app/calculator/calculator.component.ts (291 bytes)
CREATE src/app/calculator/calculator.component.css (0 bytes)
UPDATE src/app/app.module.ts (491 bytes)
```

Рис.6.3 Створення компоненти calculator

4. Створимо всі потрібні функції для роботи калькулятора у файлі calculator.component.ts:

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

```
@Component({
  selector: 'app-calculator',
  templateUrl: './calculator.component.html',
  styleUrls: ['./calculator.component.css']
})
```

```
export class CalculatorComponent {
```

```
  public num1:number;
  public num2:number;
  public result:number;
```

```
  add(){
    this.result = this.num1 + this.num2;
  }
```

```
  subtract(){
    this.result = this.num1 - this.num2;
  }
```

```
  multiply(){
    this.result = this.num1 * this.num2;
  }
```

```
  divide(){
    this.result = this.num1 / this.num2;
  }
```

```
}
```

5. Додамо необхідний код розмітки у файл calculator.component.html:

```
<div class="card" style="width: 40%; margin-left: 30%">
  <div class="card-header">CALCULATOR</div>
  <div class="card-body">
    <div class="form-group d-flex">
      <input [(ngModel)]='num1' type="number" name="num1" class="form-control">
      <input [(ngModel)]='num2' type="number" name="num2" class="form-control">
    </div>
    <div class="d-flex flex-row justify-content-md-between">
      <button class="btn btn-success" (click)="add()">Sum</button>
      <button class="btn btn-danger" (click)="subtract()">Subtraction</button>
```

```

<button class="btn btn-primary" (click)="multiply()">Multiplication</button>
<button class="btn btn-warning" (click)="divide()">Division</button>
</div>
<div class="card-footer">
  <h3>The result is {{ result }}</h3>
</div>
</div></div>

```

В результаті можемо побачити створений калькулятор:

A screenshot of a web-based calculator interface. At the top, there is a header bar labeled "CALCULATOR". Below this, there are two input fields: the first contains the number "12" and the second contains the number "6". Underneath the input fields, there are four colored buttons: a green "Sum" button, a red "Subtraction" button, a blue "Multiplication" button, and a yellow "Division" button. The "Multiplication" button is highlighted with a blue border. At the bottom of the interface, a light gray box displays the text "The result is 18".

Рис.6.4 Виконання функції додавання

A screenshot of the same web-based calculator interface. The input fields still contain "12" and "6". The "Division" button is now highlighted with a yellow border. The light gray box at the bottom displays the text "The result is 6".

Рис.6.5 Виконання функції додавання

A screenshot of the same web-based calculator interface. The input fields still contain "12" and "6". The "Multiplication" button is highlighted with a blue border. The light gray box at the bottom displays the text "The result is 72".

Рис.6.6 Виконання функції додавання



Рис.6.7 Виконання функції додавання

Висновок: в результаті лабораторної роботи було створено калькулятор використовуючи Angular фреймворк.