

# **ZPW lab**

## **laboratorium 4**

### **Wprowadzenie do Angulara**

Materiały pomocnicze do dzisiejszych zajęć: <https://angular.io/tutorial>

#### **Zadanie 1.** Weryfikacja środowiska pracy

Uwaga. Dla osób korzystających z komputera z Sali lab – proszę zalogować się na Windows i wybrać profil ostatni – Projektowanie systemów ..... .

Sprawdź czy masz na swoim komputerze zainstalowane następujące oprogramowanie:

npm - `npm -v`

Zapoznaj się z krótkim tutorialiem dotyczącym instalacji i używania Angulara  
<https://angular.io/guide/quickstart>

Na jego podstawie sprawdź czy masz zainstalowanego Angulara.

`ng version` lub `ng -v`

W przypadku gdyby nie było zainstalowanego angulara proszę o jego instalację.

W przypadku korzystania z komputerów w lab pojawia się problem rejestracji ścieżki do zainstalowanego środowiska ng (Angulara). Proszę samodzielnie ustawić lokalną zmienną systemową PATH dla zainstalowanego ng.

Potrzebujemy także edytora tekstowego obsługującego TypeScript. Proponuje korzystać z Visual Studio Code. Jeśli go nie posiadamy – zainstalować.

Zapoznaj się z możliwościami środowiska CLI - <https://cli.angular.io/>

#### **Zadanie 2.** Podstawowym językiem tworzenia oprogramowania w Angularze jest TypeScript.

Przeglądnij tutorial dotyczący TypeScript

<https://www.typescriptlang.org/docs/tutorial.html>

Zapoznając się z najważniejszymi zasadami i konstrukcjami języka.

#### **Zadanie 3.** Stwórz nowy projekt w Angularze typu HelloWorld

Zapoznaj się z wygenerowaną strukturą projektu. Przeglądnij zawartość najważniejszych plików projektu.

Generalnie interesują nas tylko te cztery pliki:

```
src/app/app.module.ts,  
src/app/app.component.ts,  
src/index.html,  
src/main.ts
```

W utworzony projekcie proszę dokonać następujących zmian, najlepiej, krok po kroku, po każdej całościowej i spójnej zmianie uruchamiając aplikację:

- zmieniamy komponent app na hello czyli:
  - zmieniamy nazwę pliku app.component.ts na hello.component.ts,
  - zmieniamy nazwę komponentu z AppComponent na HelloComponent
  - zmieniamy selektor (znacznik html) w szablonie i pliku index.html na hello-app,
  - zmieniamy treść szablonu na „Witaj Jan Kowalski”, gdzie Jan Kowalski powinno być zmiennymi imię i nazwisko przekazanymi do eksportowanej klasy wykorzystanej w szablonie: `<h1>Witaj {{imie}} {{nazwisko}}</h1>`,
- zmieniamy moduł app na main czyli:
  - zmieniamy nazwę pliku app.module.ts na main.module.ts,
  - zmieniamy nazwę modułu z AppModule na MainModule,
- **Dla ambitnych:** zmieniamy nazwę głównego katalogu aplikacji z src/app na src/hello, będzie wymagało również zmiany w pliku systemjs,config.js albo w pliku .angular-cli.json,
- zmieniamy port, na którym uruchamiana jest aplikacja na 5000:  
<http://stackoverflow.com/questions/37154813/angular-cli-server-how-to-specify-default-port>

Takie zabawy z Hello World, żeby lepiej zrozumieć, jak wczytywane są komponenty, importowane moduły, jak jest uruchamiana aplikacja. To ważne aby swobodnie poruszać się w tym środowisku.

**Zadanie 4.** Niech na stronie głównej wyświetla się obok tekstu powitalnego pole tekstowe pozwalające na wpisywanie nazwy imienia. Po wpisaniu tam wartości tekst ten powinien pojawić się od razu na ekranie w tekście powitania.

Wykorzystując interpolację zweryfikuj wynik działania następujących operacji:

```
{{2+2}}
```

```
{{ imie.length }}
```

```
{{ imie.toUpperCase() }}
```

```
{{ a = 2+3 }}
```

```
{{ windows.location.href }}
```

Wyciągnij wnioski z otrzymanych wyników i ewentualnie tak zmodyfikuj kod aby zadziałał.

### **Zadanie 5.**

Napisać funkcję tabliczka() przyjmującą dwie tablice (typu string i number), która wyświetli tabliczkę „złożenia” z zawartości dwóch tablic. Przykładowo dla tablic: ['Ala', 'ma', 'kota'] oraz [0, 1, 2, 3] otrzymamy:

Ala0 Ala1 Ala2 Ala3

ma0 ma1 ma2 ma3

kota0 kota1 kota2 kota3

Wykorzystać dwa różne rodzaje pętli z TypeScript:

<https://www.typescriptlang.org/docs/handbook/iterators-and-generators.html>

**Zadanie 6.** Stwórz nowy komponent o nazwie Produkt. Zmodyfikuj tak kod aby nowy komponent był komponentem wyświetlanym na stracie naszej aplikacji.

Komponent Produkty powinien wyświetlać listę produktów. Sama lista może być zaszyta bezpośrednio w kodzie w postaci tablicy obiektów. Pojedynczy obiekt Produkt powinien zawierać następujące pola:

Nazwa, ilość produktów, cena jednostkowa, opis oraz link do poglądowego zdjęcia.

Stwórz tablice w głównym komponencie o co najmniej 8 elementach.

Wyświetl zawartość tablicy obiektów w szablonie komponentu głównego - dyrektywa \*ngFor

Wyświetlane zdjęcia produktów proszę wyświetlać jako okrągłe.

Przy każdym produkcie powinny znajdować się 2 przyciski + i - pozwalające na dodawanie produktu do listy zakupów lub jego oddawanie.

Jeśli wartość produktu znajdującego się w tablicy będzie wynosiła 0 to należy wyświetlić inny komunikat niż gdy ilość dostępnych produktów jest większa od 0.

W przypadku gdy ilość produktu spadnie do zera przyciski - powinien zostać ukryty. Nie chcemy przecież rezerwować produktu którego już nie posiadamy. - dyrektywy ngStyle lub ngClass

Gdy ilość produktu będzie zbliżała się do 0 ( np. od 3 w dół) należy zaznaczyć to w sposób graficzny np. inne tło, kolor czcionki, wielkość fontów lub inny wizualny sposób.

Podobnie należy rozróżnić produkt o najniższej cenie jednostkowej oraz najwyższej – za pomocą dodatkowego obramowania obejmującego dany produkt - zielone – najdroższy, czerwone najtańszy.

Wypisz całą nazwę produktu dużymi literami -> skorzystaj z odpowiedniego typu pipe.

Wyświetl cenę produktu wraz z nazwa (lub znakiem płatniczym) skojarzonym z walutą np. USD - \$ , euro lub złotych.

Wyświetl również sumaryczna ilość aktualnie dostępnych produktów - jeśli wynosi on więcej niż 10 ma być wyświetlana na zielonym tle, jeśli poniżej 10 na czerwonym tle.

### **Zadanie 7. dwa niezależne komponenty**

Proszę utworzyć drugi komponent o nazwie Nagłówek. Komponent ten należy wyświetlić z poziomu komponentu Produkt. Sam komponent będzie wyświetlał informacje o nazwie firmy jej adresie oraz baner reklamowy. Przygotować odpowiedni kontent.