Laboratorium 1: Pierwszy projekt w Angular

https://angular.io/

https://www.jetbrains.com/webstorm/

Ćwiczenie 1. Tworzenie środowiska

Praca z aplikacjami frontendowymi wymaga instalacji:

- Node.js: https://nodejs.org/en/download/
- Angular-cli: https://angular.io/

Poniżej znajduje się opis procesu instalacji dla komputerów z uczelni. Na własnym komputerze trzeba będzie również zainstalować powyższe narzędzia, jednak proces może się różnić w zależności od systemu.

Na systemach Windows zawsze trzeba uaktualnić ścieżkę PATH dla Node.js.

https://angular.io/guide/quickstart#devenv

Pominąć na uczelni ----------

1. Aby rozpocząć pracę z Angularem niezbędna nam będzie biblioteka **node.js**. Jeśli jeszcze nie masz jej zainstalowanej, dokonaj instalacji: https://nodejs.org/en/download/

2. Następnie korzystając z **konsoli systemowej cmd** należy sprawdzić wersję node.js:

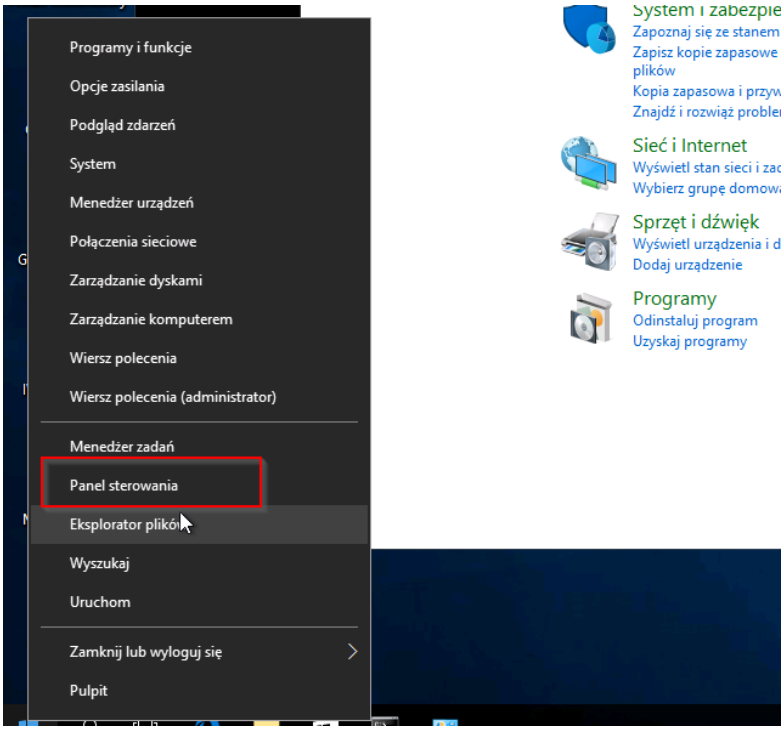
   ```
   npm –v
   ```

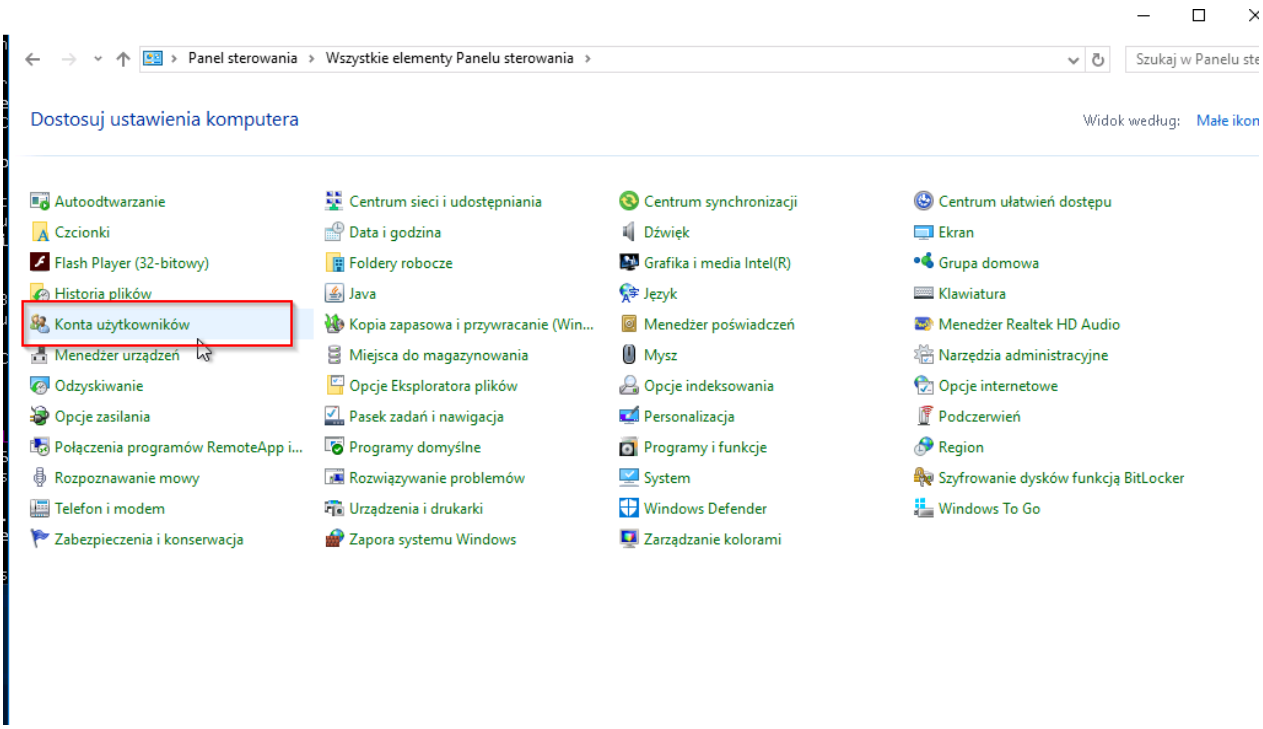
---

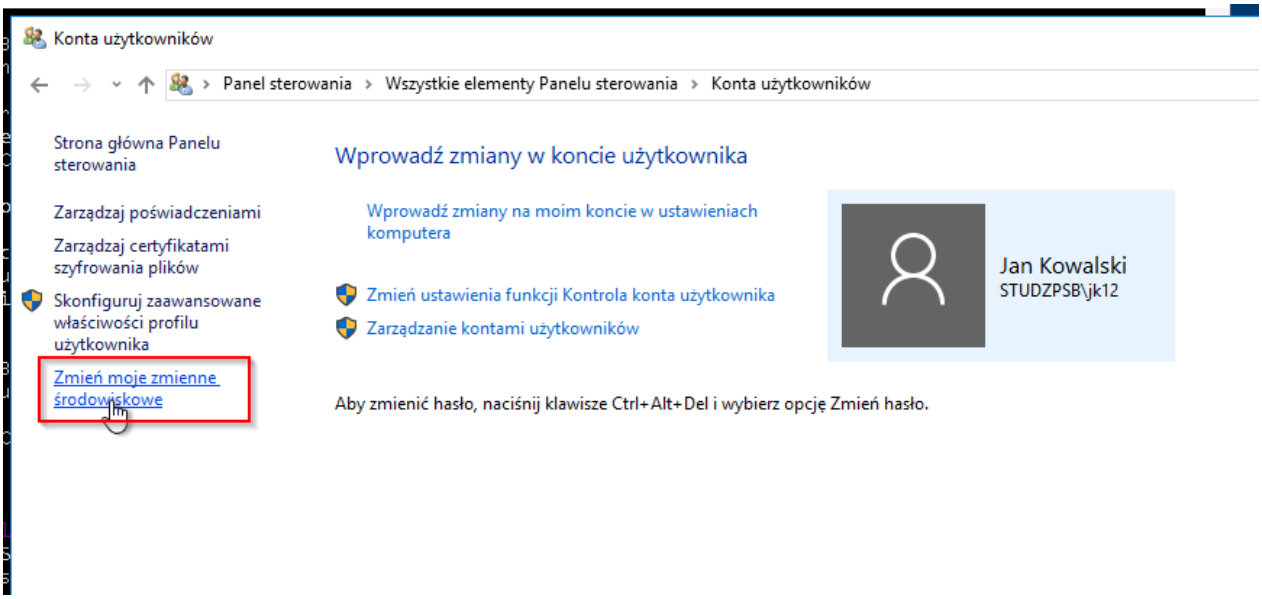
3. W kolejnym kroku trzeba zainstalować bibliotekę do obsługi Angular:

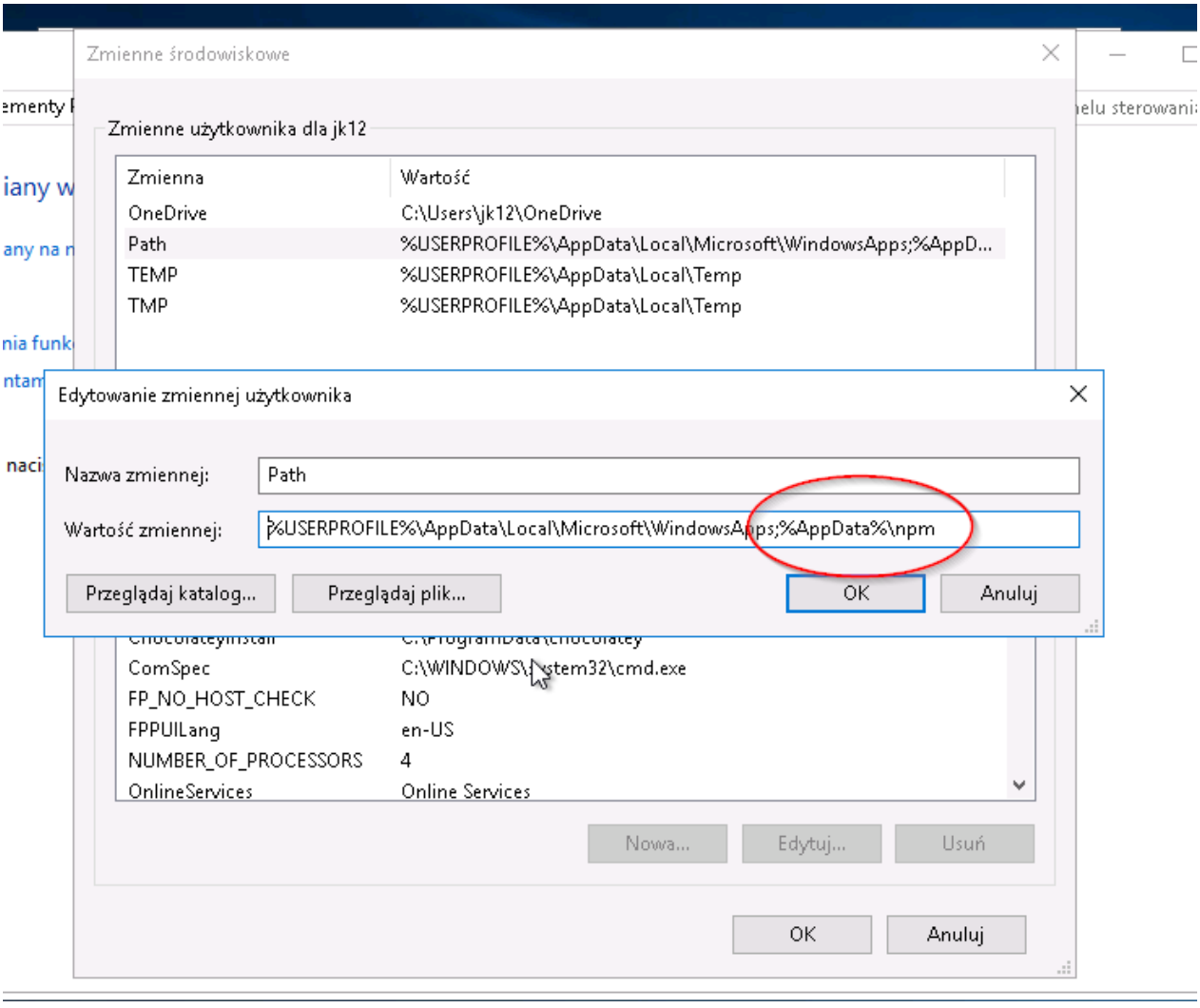
   ```
   npm install -g @angular/cli
   ```

4. Do zmiennej środowiskowej PATH trzeba dodać ścieżkę do npm: **%AppData%\npm**


Programy i funkcje
Opcje zasilania
Podgląd zdarzeń
System
Menedżer urządzeń
Połączenia sieciowe
Zarządzanie dyskami
Zarządzanie komputerem
Wiersz polecenia
Wiersz polecenia (administrator)
Menedżer zadań
Panel sterowania
Eksplorator plików
Wyszukaj
Uruchom
Zamknij lub wyloguj się
Pulpit
System i zabezpie
Zapoznaj się ze stanem
Zapisz kopie zapasowe plików
Kopia zapasowa i przyw
Znajdź i rozwiąż proble
Sieć i Internet
Wyświetl stan sieci i zac
Wybierz grupę domow
Sprzęt i dźwięk
Wyświetl urządzenia i d
Dodaj urządzenie
Programy
Odinstaluj program
Uzyskaj programy


Panel sterowania › Wszystkie elementy Panelu sterowania ›
Szukaj w Panelu ste
Dostosuj ustawienia komputera
Widok według: Małe ikon
Autoodtwarzanie
Czcionki
Flash Player (32-bitowy)
Historia plików
Konta użytkowników
Menedżer urządzeń
Odzyskiwanie
Opcje zasilania
Połączenia programów RemoteApp i...
Rozpoznawanie mowy
Telefon i modem
Zabezpieczenia i konserwacja
Centrum sieci i udostępniania
Data i godzina
Foldery robocze
Java
Kopia zapasowa i przywracanie (Win...
Miejsca do magazynowania
Opcje Eksploratora plików
Pasek zadań i nawigacja
Programy domyślne
Rozwiązywanie problemów
Urządzenia i drukarki
Zapora systemu Windows
Centrum synchronizacji
Dźwięk
Grafika i media Intel(R)
Język
Menedżer poświadczeń
Mysz
Opcje indeksowania
Personalizacja
Programy i funkcje
System
Windows Defender
Zarządzanie kolorami
Centrum ułatwień dostępu
Ekran
Grupa domowa
Klawiatura
Menedżer Realtek HD Audio
Narzędzia administracyjne
Opcje internetowe
Podczerwień
Region
Szyfrowanie dysków funkcją BitLocker
Windows To Go

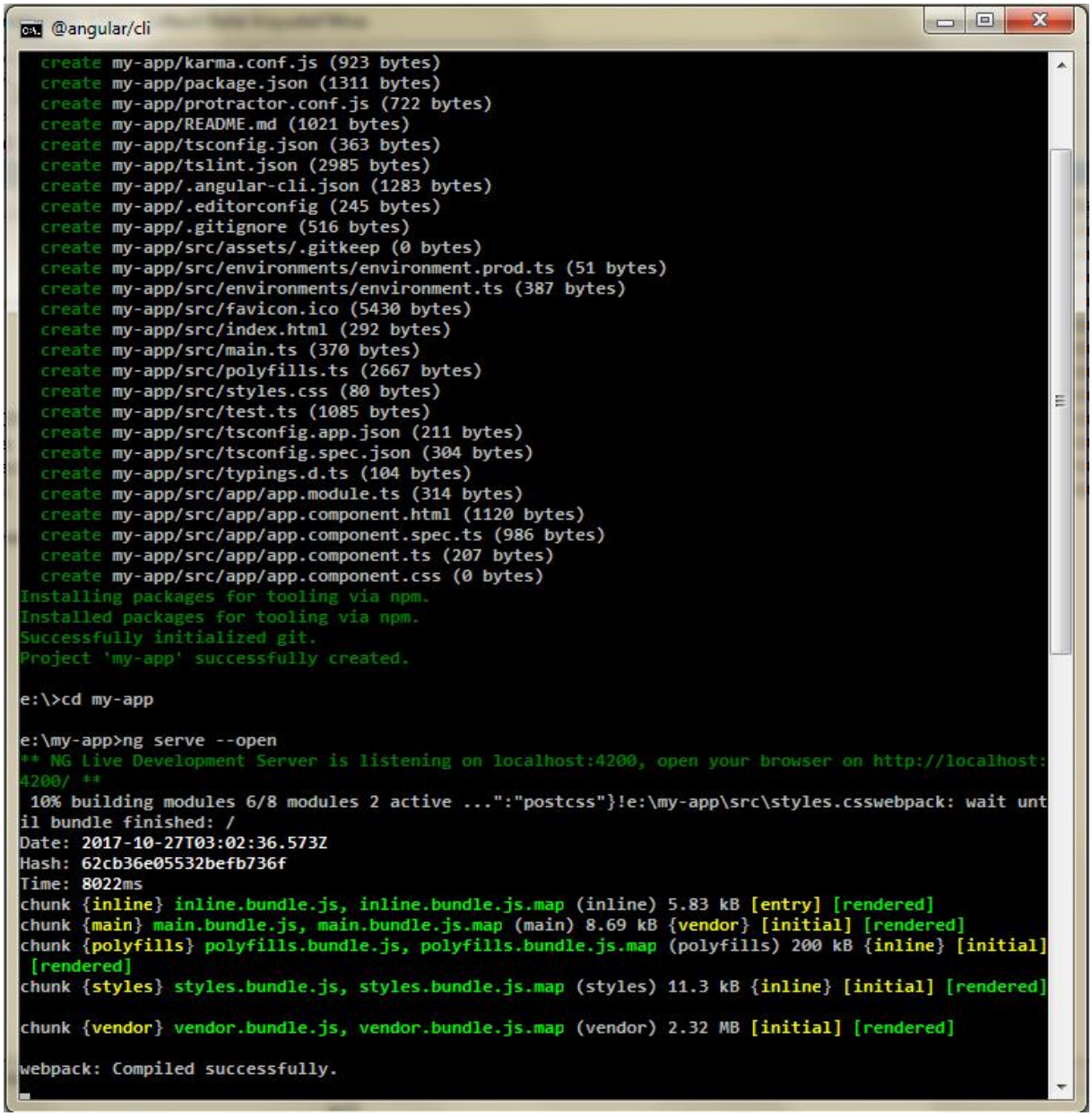

Konta użytkowników
Panel sterowania › Wszystkie elementy Panelu sterowania › Konta użytkowników
Strona główna Panelu sterowania
Zarządzaj poświadczeniami
Zarządzaj certyfikatami szyfrowania plików
Skonfiguruj zaawansowane właściwości profilu użytkownika
Zmień moje zmienne środowiskowe
Wprowadź zmiany w koncie użytkownika
Wprowadź zmiany na moim koncie w ustawieniach komputera
Zmień ustawienia funkcji Kontrola konta użytkownika
Zarządzanie kontami użytkowników
Jan Kowalski
STUDZPSB\jk12
Aby zmienić hasło, naciśnij klawisze Ctrl+Alt+Del i wybierz opcję Zmień hasło.


Zmienne środowiskowe
Zmienne użytkownika dla jk12
Zmienna
Wartość
OneDrive
C:\Users\jk12\OneDrive
Path
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;%AppD...
TEMP
%USERPROFILE%\AppData\Local\Temp
TMP
%USERPROFILE%\AppData\Local\Temp
Edytowanie zmiennej użytkownika
Nazwa zmiennej:
Path
Wartość zmiennej:
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;%AppData%\npm
Przeglądaj katalog...
Przeglądaj plik...
OK
Anuluj
ComSpec
C:\WINDOWS\system32\cmd.exe
FP_NO_HOST_CHECK
NO
FPPUILang
en-US
NUMBER_OF_PROCESSORS
4
OnlineServices
Online Services
Nowa...
Edytuj...
Usuń
OK
Anuluj

5. Pozostało już tylko utworzyć projekt. Ponownie uruchom konsolę i wpisz polecenie:

   ```
   ng new my-app
   ```

   No i nastaw się, że trochę musisz poczekać. Node.js ściąga właśnie wszystko to, czego potrzebujesz, żeby zacząć pracę z Angularem.



6. Przejdź do katalogu z projektem i uruchom serwer.

```
cd my-app
ng serve --open
```

Polecenie –open powinno automatycznie otworzyć okno przeglądarki pod adresem:

http://localhost:4200/

a Ty powinieneś zobaczyć to:

Gratulacje, właśnie utworzyłeś swój pierwszy projekt w Angular!

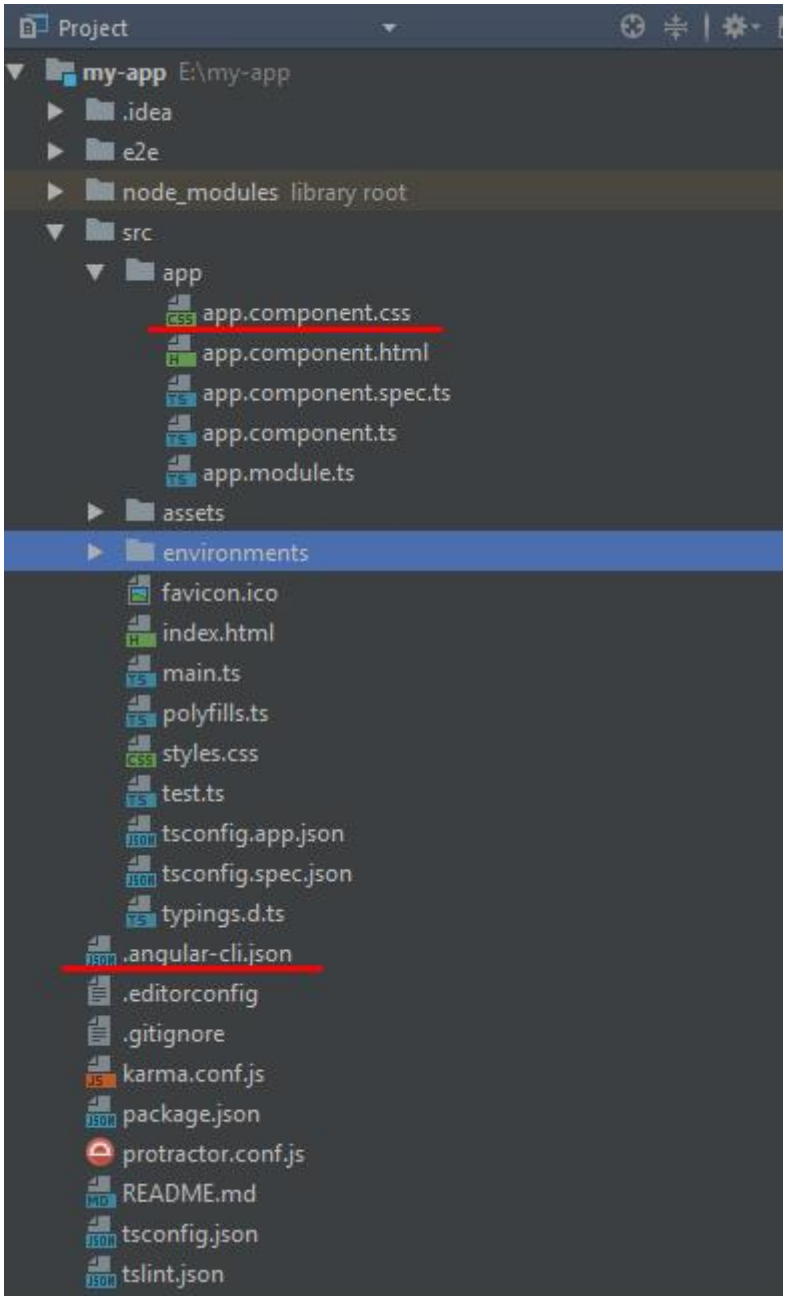
Ćwiczenie 2. Preprocesor SCSS

Ok, Angular zainstalowany, ale zanim przejdziemy do dalszej pracy skonfigurujemy sobie co nieco, żeby ułatwić sobie życie i pracę w przyszłości.

Po pierwsze skonfigurujemy Angulara tak, żebyśmy zamiast plików .css mogli używać preprocesora SASS i plików .scss.

Zaczynamy!

1. Otwórz aplikację Webstorm i wybierz opcję Open i katalog z projektem. Struktura katalogu wygląda tak:



To, co nas interesuje to katalog src – to on tak naprawdę zawiera naszą aplikację. Pozostałe katalogi i pliki to elementy środowiska wspomagające naszą pracę.

W katalogu app znajduje się główny komponent naszej aplikacji. Zanim przyjrzymy mu się bliżej, zwróćmy uwagę na *app.component.css* . To coś, czego nie chcemy – nasza aplikacja ma wykorzystywać SCSS.

Sprawmy, żeby tak się działo.

2. Na początek powiemy angularowi, że ma domyślnie korzystać z preprocesora SASS.
   W konsoli wpisz polecenie:

```
ng set defaults.styleExt scss
```

3. Sprawdźmy, czy zadziałało. Otwórzmy plik *angular-cli.json*. Na samym dole powinniśmy zobaczyć:

4. Pozostało nam jeszcze zmienić rozszerzenie już istniejącego pliku *app.component.css* na *app.component.scss*

To było proste!

To teraz coś trudniejszego. Używając preprocesora SASS zwyczajowo nie korzysta się z jednego pliku ze stylami css, ale z wielu plików podzielonych według pewnej logiki.

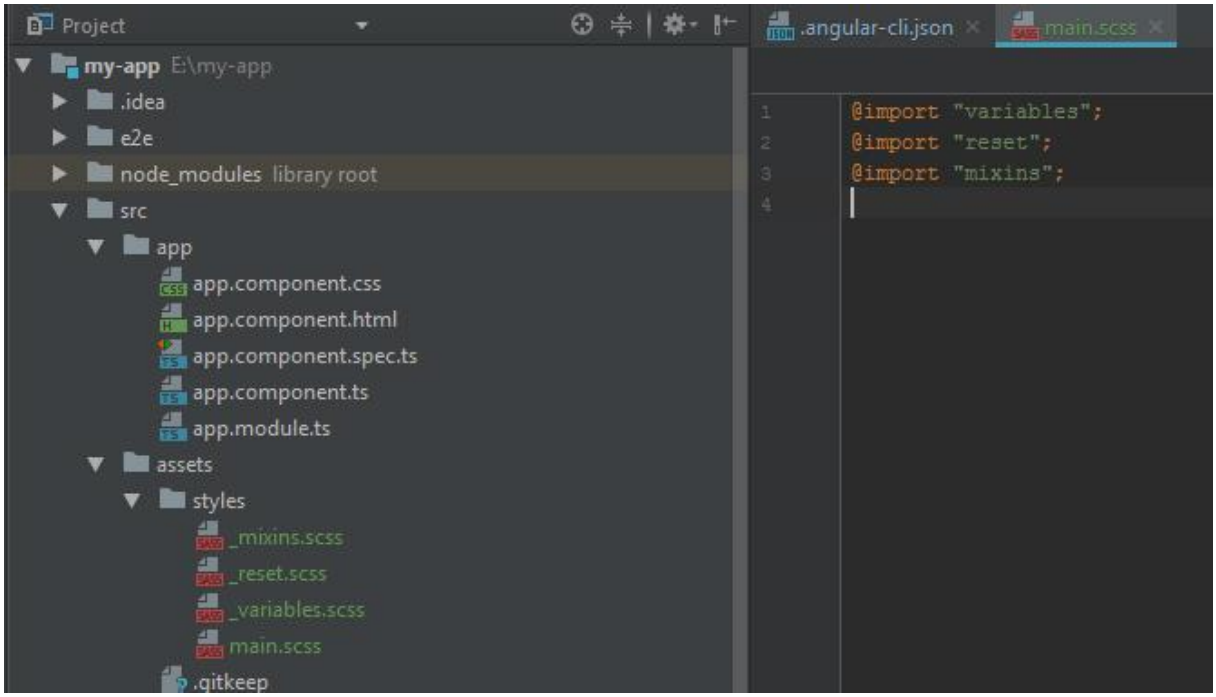
W naszym przypadku Angular sam dołącza pojedyncze pliki scss do komponentów, tworząc style lokalne, obowiązujące tylko w obrębie komponentu. Jednakże korzysta się też ze styli globalnych, gdyż to ułatwia pracę i ujednolica wygląd aplikacji.

Uparliśmy się na ten SASS przede wszystkim dlatego, że umożliwia nam tworzenie w stylach **zmiennych** oraz **mixins** – czegoś w rodzaju funkcji, które możemy dołączać w innych plikach.

5. Na początek w katalogu *assets* utwórzmy katalog *styles*, a w nim kilka plików:

   - _variables.scss – plik, w którym będziemy trzymać zmienne
   - _mixins.scss – plik na mixins
   - _reset.scss – plik czyszczący niepotrzebne style przeglądarek
   - main.scss – plik scalający wszystkie pliki scss w jedno

A następnie w pliku *main.scss* zaimportujmy pozostałe pliki.



6. Na koniec zmienimy ścieżkę w *angular-cli.json* w taki sposób, żeby wiedział, z czego ma korzystać przy tworzeniu styli aplikacji.

   Zamiast

```
"styles": [
        "styles.css"
 ],
```

   Wpisujemy

```
"styles": [
        "assets/styles/main.scss"
],
```

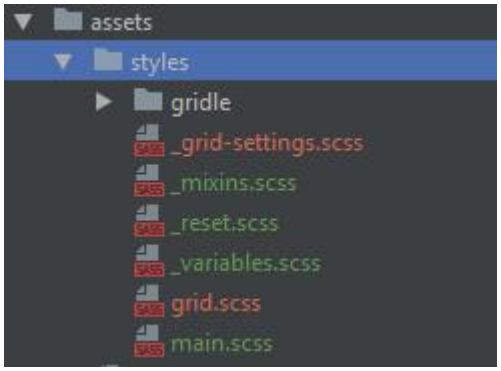
Ćwiczenie 3. Siatka grid

Na zajęciach z technologii hipertekstowych używaliśmy Bootstrapa, gdyż zapewniał nam wsparcie dla RWD (Responsive Web Design) oraz dostarczał gotowe rozwiązania JavaScript. Teraz jednak do JavaScript mamy Angulara, więc nie ma sensu obciążać się tak wielką biblioteką, skoro wszystko możemy napisać samodzielnie.

Jest jedna rzecz, której nie ma sensu pisać od podstaw: siatka grid. Wykorzystamy sobie do niej niewielką bibliotekę **gridle{.scss}**: http://gridle.org/documentation.

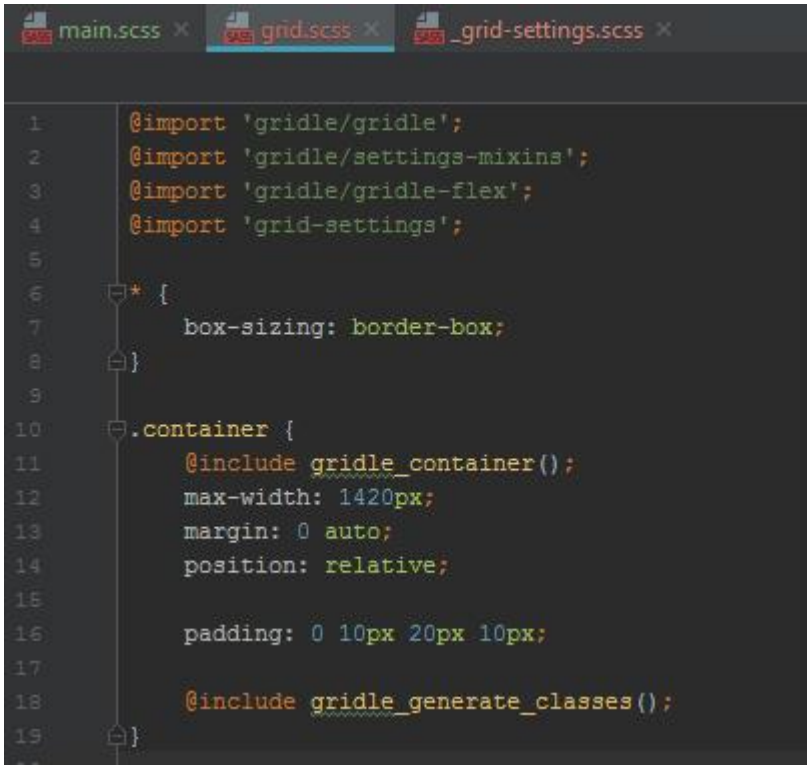
Nie będziemy samodzielnie konfigurować całego gridle. Pobierzemy paczkę ze strony laboratoriów: gridle.zip

1. Na początek należy ściągnąć bibliotekę ze strony i rozpakować ją na dysku. Skopiować całość do naszego katalogu ze stylami scss:



2. Dołączmy plik *grid.scss* do pliku – żeby był kompilowany od razu ze wszystkimi plikami scss.

3. Zajrzyjmy teraz do pliku *grid.scss*



Od góry na początek zaimportowaliśmy gridla i ustawienia reguł, następnie mamy gridle-flex. To oznacza, że będziemy korzystać z możliwości Flexbox: https://www.w3schools.com/css/css3_flexbox.asp Idźmy dalej.

Jak widać mamy tu już ustawione wartości dla głównego kontenera naszej aplikacji. Zmieńmy szerokość na 1640px, ustawmy też padding na: 10px 20px.

Tu już nic więcej nie mamy do roboty oprócz przyjrzenia się linii

```
@include gridle_generate_classes();
```

4. Otwórzmy plik *grid-settings.scss* i zobaczmy co tam się dzieje. A dzieje się dużo. Na początek mamy rejestrację kilku klas css, które będziemy wykorzystywać później.
Następnie pojawia się *gridle_setup*.

Domyślnie gridle dostarcza 12 kolumn z odstępem pomiędzy nimi 20px.
Przestawmy to na 16 kolumn i odstęp 0.

main.scss grid.scss _grid-settings.scss

```
@include gridle_set_classname_map('grid', ('grid-', '', '%column', '@', '%state'));
@include gridle_set_classname_map('grid-table', ('grid-', '', 'table', '@', '%state'));
@include gridle_set_classname_map('grid-adapt', ('grid-', '', 'adapt', '@', '%state'));
@include gridle_set_classname_map('grid-grow', ('grid-', '', 'grow', '@', '%state'));
@include gridle_set_classname_map('grid-centered', ('grid-', '', 'centered', '@', '%state'));

@include gridle_setup((
        context: 12,
        gutter-width: 20,
        debug: true,
        gridle-vendor-prefix: false,
        debug-show-class-names: true
));

@include gridle_register_state(mobile, (
        max-width : 768px
));

@include gridle_register_state(tablet, (
        min-width: 769px,
        max-width: 1024px
));

@include gridle_register_state(all-mobile, (
        max-width: 1024px
));

@include gridle_register_state(desktop, (
        min-width: 1025px,
        max-width: 1599px
));

@include gridle_register_state(all-desktop, (
        min-width: 1025px
));
```

5. Jeszcze niżej mamy klasy odpowiedzialne za punkty przełamania dla różnych urządzeń. Mamy mobile, tablet, desktop. To odpowiedniki bootstrapowych klas *col-sm, col-md, col-lg*. Podczas pracy nad aplikacją będziemy korzystać z nich w następujący sposób:

```
<div class="grid-4 grid-8@tablet grid-16@mobile"></div>
```

6. Dopiszmy sobie jeszcze jedną regułę dla urządzeń o rozdzielczości większej niż 1600px.

```
@include gridle_register_state(large, (
        min-width: 1600px
));
```

Skończyliśmy! Ale…

Żeby nie pisać wszystkich styli od podstaw dołączymy kilka przygotowanych przeze mnie plików scss znajdujących się w paczce *basic_scss.zip*. Rozpakowujemy do styli, nadpisujemy istniejące pliki i dołączamy w *main.scss*.

@import 'app' powinno się znaleźć na końcu, to będzie nasz plik do styli globalnych, jak na przykład dla body, odnośników, itd.

Dodatkowo w paczce fonts.zip znajduje się katalog z ikonami fontAwesome http://fontawesome.io/icons/ Wgrywamy go do katalogu assets.

Poza kończeniu prac w konsoli wpisujemy:

```
ng serve --open
```

Jeśli nie będzie błędów, aplikacja odpali się pod adresem http://localhost:4200/ i będzie można zobaczyć efekty.

Można zmieniać do woli!

Sprawdźmy jeszcze czy to działa.

Przebudujmy trochę plik *app.component.html*

app.component.html

```
<!--The content below is only a placeholder and can be replaced.-->
<div style="text-align:center">
  <h1>
    Welcome to {{title}}!
  </h1>
  <img width="300" src="
</div>
<h2>Here are some links to help you start: </h2>
<ul>
  <li>
    <h2><a target="_blank" rel="noopener" href="https://angular.io/tutorial">Tour of Heroes</a></h2>
  </li>
  <li>
    <h2><a target="_blank" rel="noopener" href="https://github.com/angular/angular-cli/wiki">CLI Documentation</a></h2>
  </li>
  <li>
    <h2><a target="_blank" rel="noopener" href="https://blog.angular.io/">Angular blog</a></h2>
  </li>
</ul>
```

Zamiast `<div style="text-align:center">` wstawimy głowny konterer gridle `<div class=”container”>`

Dodamy `<div class=”row”>` oraz dwie kolumny, do których przeniesiemy nagłówek h1, obrazek oraz listę.

Na koniec dodamy do nagłówka h1 ikonke font awesome

```
<span class=fa fa-caret-right”></span>
```

app.component.html

```
<!--The content below is only a placeholder and can be replaced.-->
<div class="container">
    <div class="row">
        <div class="grid-5 grid-centered">
            <h1><span class="fa fa-caret-right"></span> Welcome to {{title}}!</h1>
            <img width="300" src="
        </div>
        <div class="grid-11 grid-centered">
            <h2>Here are some links to help you start: </h2>
            <ul>
                <li>
                    <h2><a target="_blank" rel="noopener" href="https://angular.io/tutorial">Tour of Heroes</a></h2>
                </li>
                <li>
                    <h2><a target="_blank" rel="noopener" href="https://github.com/angular/angular-cli/wiki">CLI Documentation</a></h2>
                </li>
                <li>
                    <h2><a target="_blank" rel="noopener" href="https://blog.angular.io/">Angular blog</a></h2>
                </li>
            </ul>
        </div>
    </div>
</div>
```

Końcowy wynik nie jest zbyt imponujący, ale widać, że wszystko działa.