

# FIAP



## OPTIMIZATION TECHNOLOGY

2º  
semestre  
2017

[www.fiap.com.br](http://www.fiap.com.br)

RENATO BORTOLIN  
[renatoboschini@gmail.com](mailto:renatoboschini@gmail.com)

## 1. AngularJs

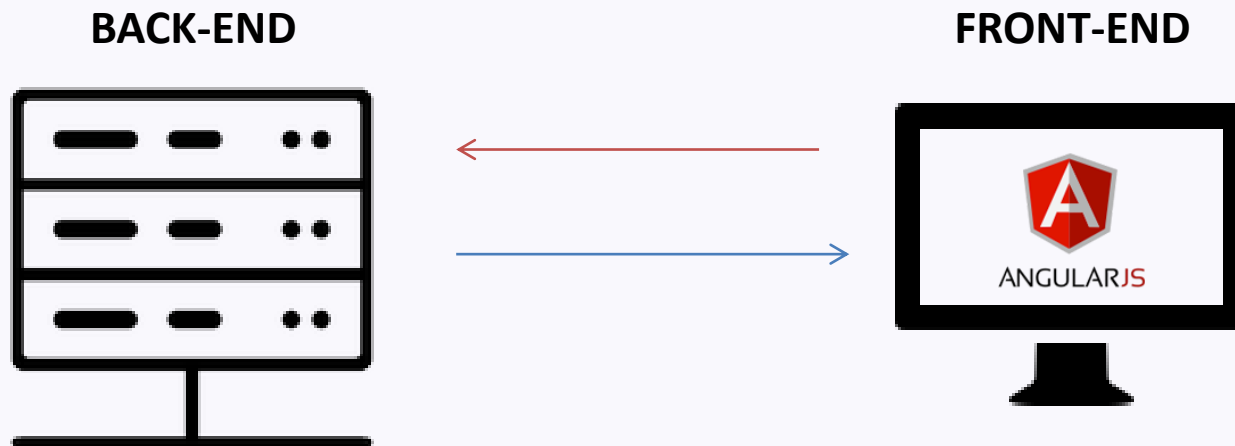
# Conceito



ANGULARJS

**AngularJS** é um framework para construir aplicações web baseada em HTML5, CSS e JavaScript, mantido pelo Google.

open-source

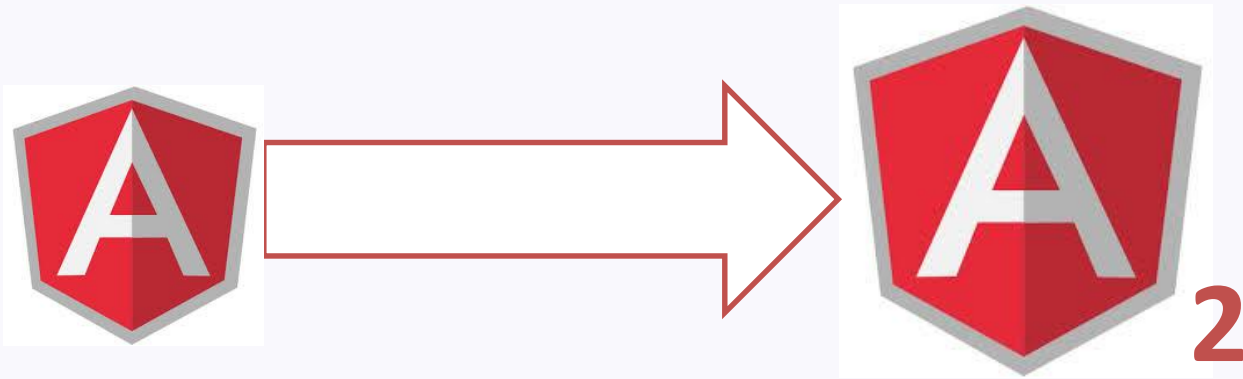


Permite entregar as tecnologias (HTML5, CSS e JS) em uma aplicação que executa no browser capaz de consumir um ou vários serviços disponibilizados por um servidor.

## Angulajs é um **Client-side**

**Client-side** é aquele que roda do lado do cliente (Ex. Javascript), ou seja, no browser do usuário;  
**Server-side** roda no servidor (Ex. PHP, JAVA, . NET...)

# Conceito



**Foi rescrito todo o framework do zero!!!!  
Quebrando a compatibilidade com a primeira versão!**







1

Fatores que levaram a nova versão foram:

**1** - Não foi projetado para desenvolver aplicações complexas. Objetivo inicial era para aplicações rápidas e para persistir formulários; Feita para designers *prototipar* mais rápida as telas;

**2** - Problemas de desempenho, pois aumentou o número de API's

**3** – Performance ruim



Fatores que levaram a nova versão foram:

- 1 – Framework mais moderno;
- 2 – Mais aderente aos padrões da web (HTML5);
- 3 – Mais simples e mais fácil de evoluir.
- 4 – Uma das principais mudanças é que na primeira versão tínhamos que trabalhar com **CONTROLLERS e outros objetos** e agora com esta versão usamos **CLASSES**, utilizando **TypeScript** do compiladas para **JavaScript**.

# O que precisa?



ANGULARJS

1º



**Node.js** é uma plataforma para desenvolvimento de aplicações server-side baseadas em rede utilizando JavaScript e o V8 JavaScript Engine, ou seja, com **Node.js** podemos criar uma variedade de aplicações Web utilizando apenas código em JavaScript.

<https://nodejs.org/>

node

HOME | ABOUT | DOWNLOADS | DOCS | FOUNDATION | GET INVOLVED | SECURITY | NEWS

node JS Interactive  
NORTH AMERICA  
Vancouver, Canada  
October 4-6, 2017

Join us for Node.js Interactive happening in Vancouver, Canada  
October 4 - 6, 2017

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, npm, is the largest ecosystem of open source libraries in the world.

Important security releases, please update now!

Download for Windows (x64)

v6.11.1 LTS  
Recommended For Most Users

v8.2.1 Current  
Latest Features

Other Downloads | Changelog | API Docs

Sempre a última versão



# O que precisa?



Depois de instalado o **NodeJs**, vamos ativar alguns pacotes via prompt de comando.

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [versão 10.0.14393]
(c) 2016 Microsoft Corporation. Todos os direitos reservados.
C:\Users\renat>
```

1 – Verificar a versão do **nodeJs**  
Instalado utilizando o comando  
> ***node -v***

2 – Instalar os pacotes do  
typescript  
> ***npm install -g typescript***

3 – Instalar os pacotes angular  
cli, utilizando o comando  
> ***npm install -g @angular/cli***

<https://www.npmjs.com/>

**NPM** é o gerenciador de pacotes que vem pré-instalado com a plataforma de servidor Node.js. É uma ferramenta que auxilia um desenvolvedor a instalar e gerenciar programas de terceiros.



# O que precisa?



Vamos verificar se os pacotes Angular cli foram instalados, utilizando o comando

**ng -v**

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [versão 10.0.15063]
(c) 2017 Microsoft Corporation. Todos os direitos reservados.

C:\Users\renat>ng -v

Angular CLI
@angular/cli: 1.2.6
node: 8.2.1
os: win32 x64

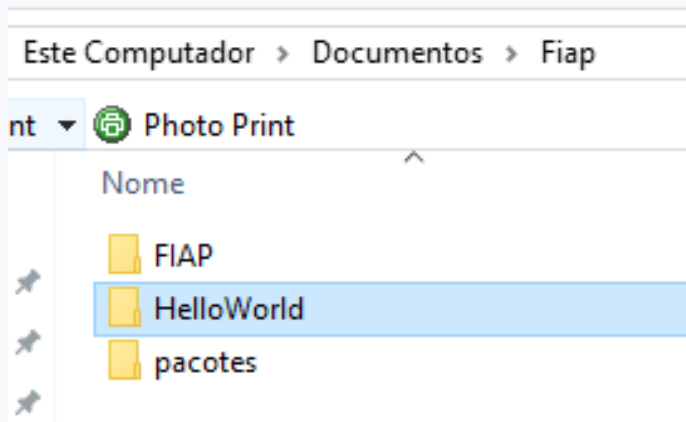
C:\Users\renat>
```



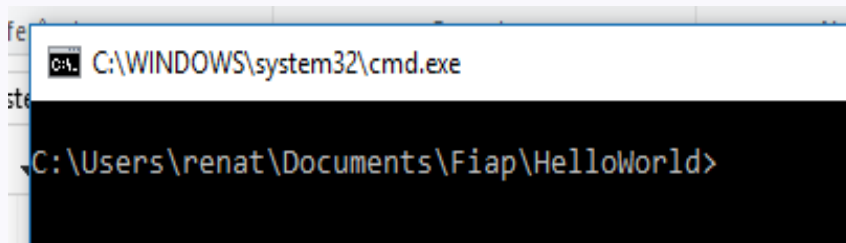
# Hello World



Vamos criar nosso primeiro projeto em AngularJs com o nome de **Hello World**, para isso criei um diretório para o seu projeto.



E depois no prompt de comando acesse localize este endereço.



Entre com o comando  
`ng new nome-da-pasta`

# Hello World



Neste passo será criada todas as bibliotecas necessárias para iniciar o seu novo projeto o AngularJS. Este processo demora um pouco.

```
C:\Users\renat\Documents\Fiap\FIAP\2017\2TINR\OPTIMIZATION TECHNOLOGY\2_Semestre\Angular\exec1>ng new ola-mundo
installing ng
  create .editorconfig
  create README.md
  create src\app\app.component.css
  create src\app\app.component.html
  create src\app\app.component.spec.ts
  create src\app\app.component.ts
  create src\app\app.module.ts
  create src\assets\.gitkeep
  create src\environments\environment.prod.ts
  create src\environments\environment.ts
  create src\favicon.ico
  create src\index.html
  create src\main.ts
  create src\polyfills.ts
  create src\styles.css
  create src\test.ts
  create src\tscconfig.app.json
  create src\tscconfig.spec.json
  create src\typings.d.ts
  create .angular-cli.json
  create e2e\app.e2e-spec.ts
  create e2e\app.po.ts
  create e2e\tscconfig.e2e.json
  create .gitignore
  create karma.conf.js
  create package.json
  create protractor.conf.js
  create tsconfig.json
  create tslint.json
Installing packages for tooling via npm.
Installed packages for tooling via npm.
Project 'ola-mundo' successfully created.
```

# Hello World



Depois que os pacotes foram instalados com sucesso, vamos agora executar no browser.

Mas primeiro vamos acessar via linha de comando o diretório criado e depois executar

```
C:\WINDOWS\system32\cmd.exe

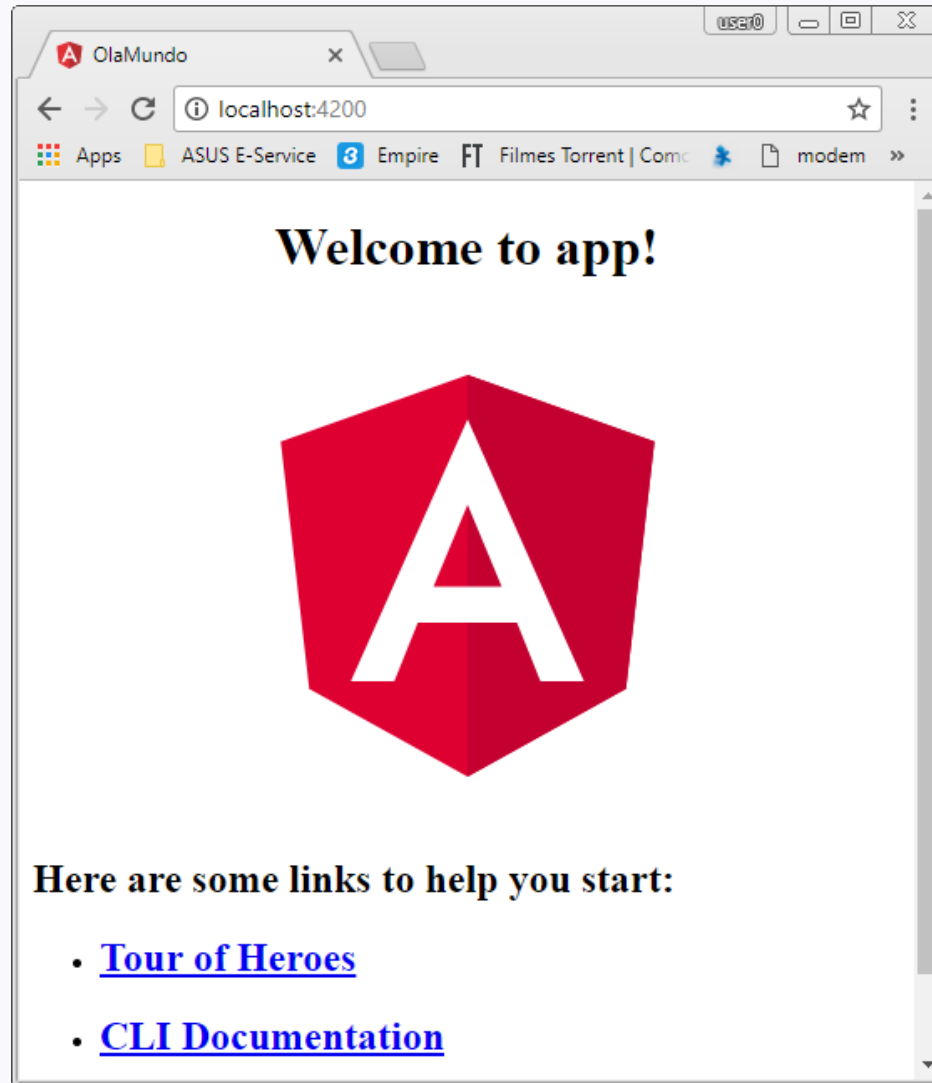
C:\Users\renat\Documents\Fiap\HelloWorld>
C:\Users\renat\Documents\Fiap\HelloWorld>cd ola-mundo
C:\Users\renat\Documents\Fiap\HelloWorld\ola-mundo>

C:\Users\renat\Documents\Fiap\HelloWorld\ola-mundo>ng serve
```

```
** NG Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200 **
Hash: a896d51d1d1e06b45d50
Time: 11893ms
chunk    {0} polyfills.bundle.js, polyfills.bundle.js.map (polyfills) 177 kB {4} [initial] [rendered]
chunk    {1} main.bundle.js, main.bundle.js.map (main) 5.34 kB {3} [initial] [rendered]
chunk    {2} styles.bundle.js, styles.bundle.js.map (styles) 10.5 kB {4} [initial] [rendered]
chunk    {3} vendor.bundle.js, vendor.bundle.js.map (vendor) 2.19 MB [initial] [rendered]
chunk    {4} inline.bundle.js, inline.bundle.js.map (inline) 0 bytes [entry] [rendered]
webpack: Compiled successfully.
```



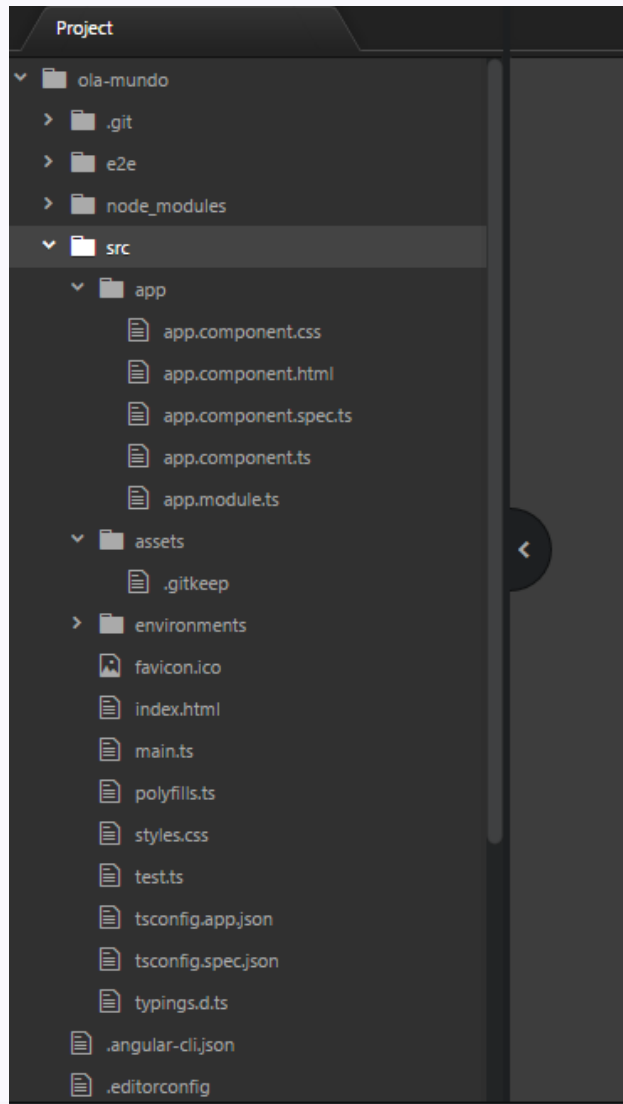
# Hello World



# Entendendo a Estrutura



ANGULARJS



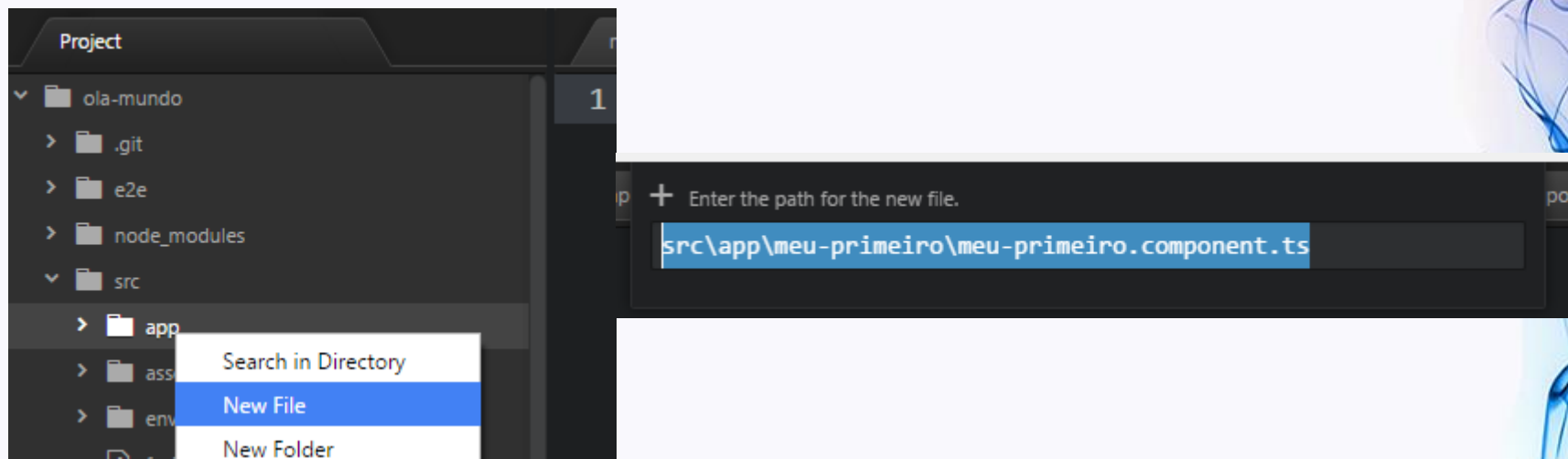
# Criando novo Component



ANGULARJS

## Manualmente

No Aton, clique com o botão direito encima da pasta APP e vamos criar uma nova página com o nome de “**meu-primeiro-component**”



src\app\meu-primeiro\meu-primeiro.component.ts

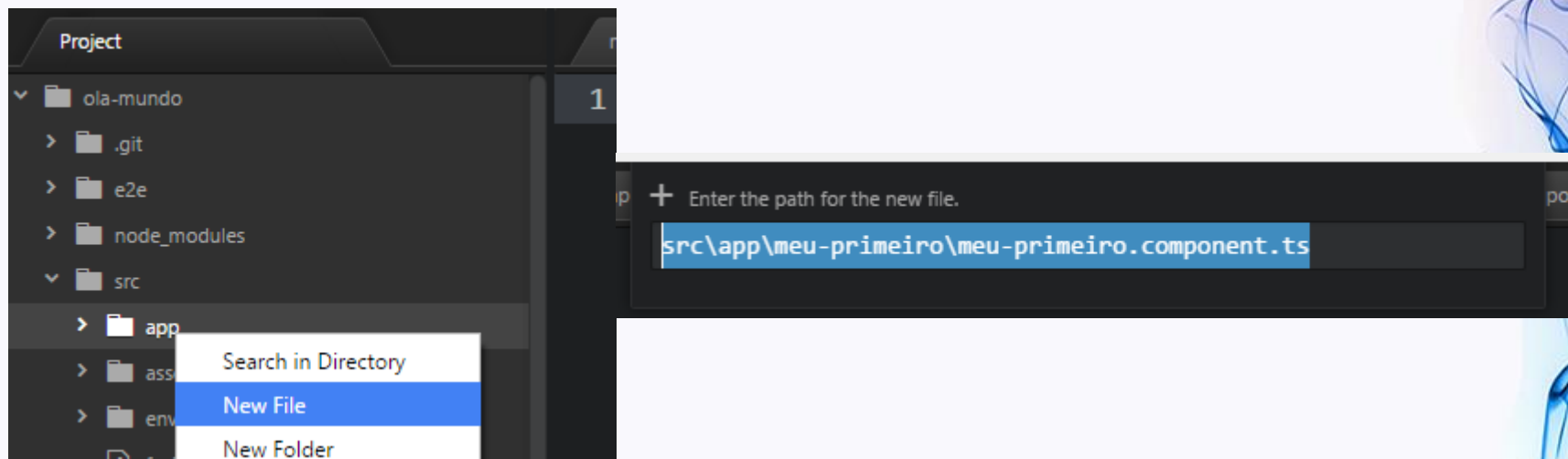
# Criando novo Component



ANGULARJS

## Manualmente

No *Aton*, clique com o botão direito encima da pasta **APP** e vamos criar uma nova página com o nome de “**meu-primeiro-component**”



src\app\meu-primeiro\meu-primeiro.component.ts

# Criando novo Component



ANGULARJS

## Manualmente

Arquivo criado, vamos criar uma **classe**, utilizando os padrões corretos de nomenclatura de classes do Java, para isso criei.

A screenshot of an IDE interface. On the left, a 'Project' sidebar shows a file tree with folders like 'ola-mundo', '.git', 'e2e', 'node\_modules', 'src', 'app', 'meu-primeiro', and 'minha-lista'. The file 'meu-primeiro.component.ts' is selected. On the right, the code editor shows the file 'meu-primeiro.component.ts' with line numbers 1 through 5. The code contains a class definition: 'class MeuPrimeiroComponent{' on line 3 and '}' on line 4. The tabs at the top are 'meu-primeiro.component.ts', 'app.module.ts', and 'app.component.html'.



# Criando novo Component



ANGULARJS

## Manualmente

Agora vamos criar uma “anotação dos metadados” o **seletor** e o **template** deste novo componente , para isso entre com os comandos

```
meu-primeiro.component.ts  app.module.ts  app.component.html
1
2  @Component({
3    selector: 'meu-primeiro-component',
4    template: '<p>meu primeiro componente com angular2</p>'
5
6  })
7
8  class MeuPrimeiroComponent{
9  }
< 0
```

# Criando novo Component



ANGULARJS

## Manualmente

A classe Component precisa ser importada das bibliotecas do Angular2 no início do código

```
1
2 import {Component} from '@angular/core';
3
4 @Component({
5   selector: 'meu-primeiro-component',
6   template: `<p>meu primeiro componente com angular2</p>`
7
8 })
9
10 class MeuPrimeiroComponent{
11 }
```

# Criando novo Component



ANGULARJS

## Manualmente

Para que este componente possa ser reutilizado em outros arquivos, precisamos colocar um **EXPORT** na frente da classe criada

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

@Component({
  selector: 'meu-primeiro-component',
  template: '<p>meu primeiro componente com angular2</p>'
})

export class MeuPrimeiroComponent{
}
```

# Criando novo Component



ANGULARJS

## Manualmente

Criamos nosso primeiro componente, mas agora precisamos declarar ele no módulo do Angular, para isso acesse o arquivo “app.module.ts”

E na **@NgModule** vamos inserir o novo componente

```
@NgModule({  
  declarations: [  
    AppComponent,  
    MeuPrimeiroComponent  
  ],  
  imports: [  
    BrowserModule  
  ],  
  providers: [],  
  bootstrap: [AppComponent]  
})
```

# Criando novo Component



ANGULARJS

## Manualmente

O **@NgModule** não conhece da onde vem o classe `MeuPrimeiroComponent`, Para isso precisamos importar ele no começo do código.

Nota: Utilize o auto completar do seu edito.

```
import {MeuPrimeiroComponent} from './meu-primeiro/meu-primeiro.component';
```



# Criando novo Component



ANGULARJS

## Manualmente

Agora vamos adicionar este novo componente na página “**app.component.html**”

app.component.html — C:\Users\renat\Documents\Fiap\FIAP\2017\2TINR\OPTIMIZATION TECHNOLOGY\2\_Semestre\Angular\exec1\ola-mundo — Atom

File Edit View Selection Find Packages Help

Project

- src
  - app
    - meu-primeiro
      - meu-primeiro.component.ts
    - minha-lista
    - app.component.css
    - app.component.html**
    - app.component.spec.ts
    - app.component.ts
    - app.module.ts
  - assets

```
1  <!--The content below is only a placeholder and can be re
2  <div style="text-align:center">
3    <h1>
4      Welcome todddd {{title}}!
5    </h1>
6  <meu-primeiro-component></meu-primeiro-component>
7
8  <img width="300"
  src="data:image/svg
```

**Welcome todddd app!**

meu primeiro componente com angular2



# Criando novo Component



ANGULARJS

## Automático - Angular CLI

Vamos parar o nosso **NG Serve** utilizando o **CTRL+C**

```
Time: 339ms
chunk    {0} polyfills.bundle.js, polyfills.bundle.js.map (polyfills) 177 kB {4} [initial]
chunk    {1} main.bundle.js, main.bundle.js.map (main) 7.83 kB {3} [initial]
chunk    {2} styles.bundle.js, styles.bundle.js.map (styles) 10.5 kB {4} [initial]
chunk    {3} vendor.bundle.js, vendor.bundle.js.map (vendor) 2.19 MB [initial]
chunk    {4} inline.bundle.js, inline.bundle.js.map (inline) 0 bytes [entry]
webpack: Compiled successfully.
webpack: Compiling...
Hash: 31c12c05dc86b0c14f3f
Time: 426ms
chunk    {0} polyfills.bundle.js, polyfills.bundle.js.map (polyfills) 177 kB {4} [initial]
chunk    {1} main.bundle.js, main.bundle.js.map (main) 7.83 kB {3} [initial]
chunk    {2} styles.bundle.js, styles.bundle.js.map (styles) 10.5 kB {4} [initial]
chunk    {3} vendor.bundle.js, vendor.bundle.js.map (vendor) 2.19 MB [initial]
chunk    {4} inline.bundle.js, inline.bundle.js.map (inline) 0 bytes [entry]
webpack: Compiled successfully.
^CDeseja finalizar o arquivo em lotes (S/N)?
```

# Criando novo Component



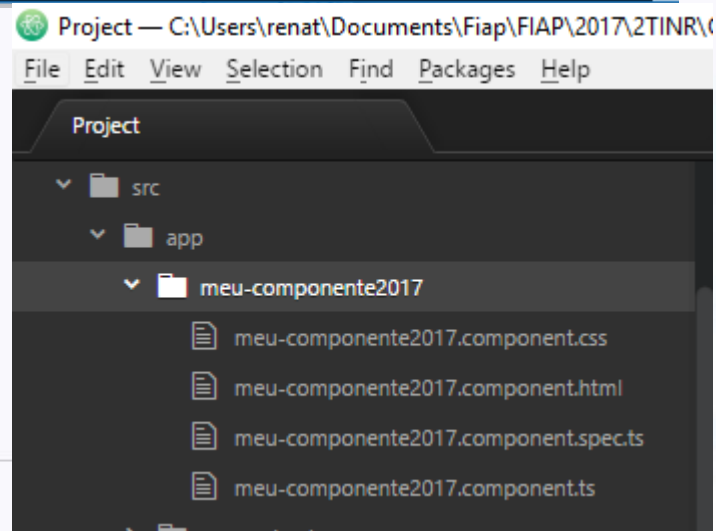
ANGULARJS

## Automático - Angular CLI

Vamos utilizar o comando **> ng g c meu-componente2017**

```
C:\Users\renat\Documents\Fiap\FIAP\2017\2TINR\OPTIMIZATION TECHNOLOGY\2_Semestre
\Angular\exec1\ola-mundo>ng g c meu-componente2017
installing component
  create src\app\meu-componente2017\meu-componente2017.component.css
  create src\app\meu-componente2017\meu-componente2017.component.html
  create src\app\meu-componente2017\meu-componente2017.component.spec.ts
  create src\app\meu-componente2017\meu-componente2017.component.ts
  update src\app\app.module.ts

C:\Users\renat\Documents\Fiap\FIAP\2017\2TINR\OPTIMIZATION TECHNOLOGY\2_Semestre
\Angular\exec1\ola-mundo>
```



# Criando novo Component



ANGULARJS

## Automático – Angular CLI

Vamos analisar os arquivos gerados e vamos acrescentar o novo componente na página “**app.component.html**”

```
5     </h1>
6     <meu-primeiro-component></meu-primeiro-component>
7
8     <app-meu-componente2017></app-meu-componente2017>
9
10    <img width=
11    <img width="300"
```

## Welcome todddd app!

meu primeiro componente com angular2

**meu-componente2017 works!**



# FIAP



Copyright © **2017** Prof. Renato Bortolin Boschini

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).