Iniciando com Angular 4 + Angular Cli

Lembrando, tenha instalado em sua máquina:

Visual Studio Code: https://code.visualstudio.com/

Node 6+: <https://nodejs.org/en/>

Com o Node Instalado:

npm install -g @angular-cli@latest

npm install -g typescript

ng new ‘nome do projeto’

cd ‘nome do projeto’

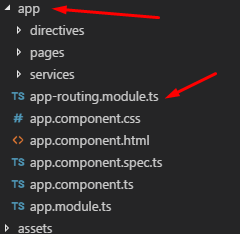
code .

Primeiro passo, vamos separar em um novo módulo as rotas para que fique de uma maneira simples de manipularmos.

Digite:

ng g module app-routing.module.ts

Note que ele irá criar uma pasta sendo seu pai, você pode tirar ela dessa hierarquia, sendo seu novo pai a pasta app.

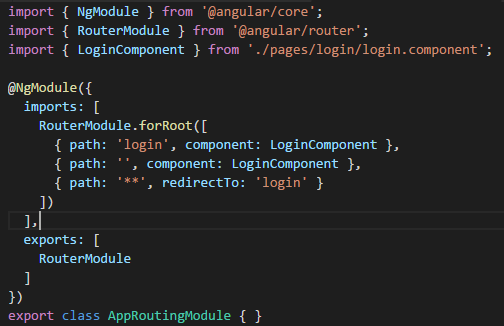


Agora vamos criar nossa primeira página, que será a de login, para isso é bem simples, use o comando.

ng g component pages/login

Logo após procure pelo arquivo criado anteriormente: app-routing.module.ts

Dentro de seu conteúdo, o código deve manter assim:



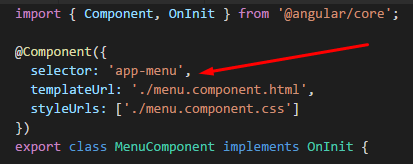
Note que tendo o path vazio ‘’ significa que será a rota Inicial do projeto, o path com dois asteriscos ‘\*\*’ significa o redirecionamento caso não bata em nenhuma rota (utilizamos muito aqui a página de 404’, e o resto segue com o padrão.

É extremamente importante, pois com a biblioteca ‘RouterModule’ vamos utilizar as rotas, utilizar o conceito de ‘partial view’ e a tag <router-outlet> será reconhecida.

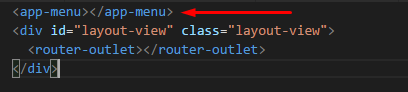
Feito isso vamos gerar um componente menu.

ng g component pages/menu

Acesso o arquivo ‘menu.component.ts’ e repare no seletor.

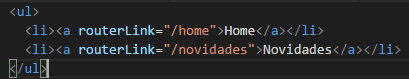


Agora acesse o arquivo ‘app.component.html’ e acrescente o seletor.

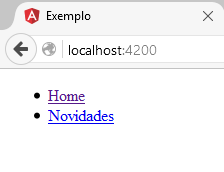


O template irá invocar o conteúdo do menu dentro da nossa ‘Layout’, por isso vamos adicionar algum conteúdo lá dentro.

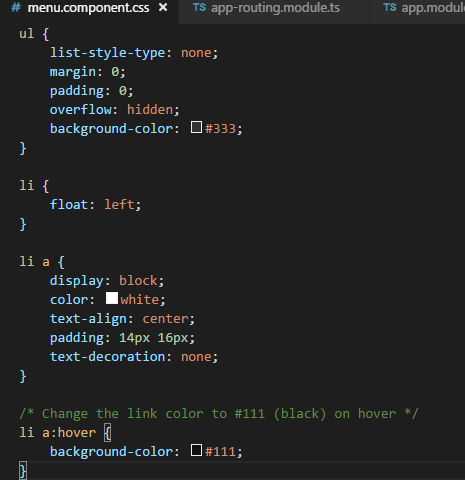
Note que essas rotas são



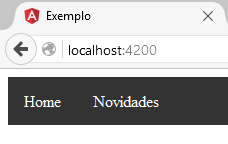
Sendo assim, renderizando vamos obter o seguinte resultado.



Mas queremos dar vida ao menu, vamos acrescentar algum css na página, é só acessar o arquivo ‘menu.component. css’, vou aplicar como exemplo:



Ficando assim:



\*Nota:

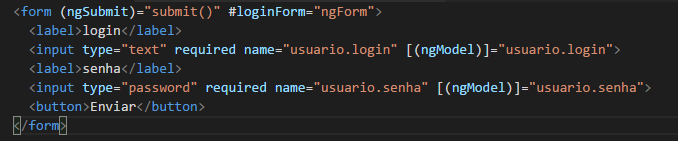
Podemos usar o SCSS e SASS, eu recomendo usar o SCSS, para isso utilize algum destes comandos na criação do projeto:

ng new My\_New\_Project --style=(sass|less|stylus)

Como no nosso caso ele já existe, podemos utilizar:

ng set defaults.styleExt (scss | less | styl)

Voltando agora a página de login, vamos dar algum conteúdo a ela.



Agora vamos criar nosso serviço que será chamado pelo typescript dessa página.

ng g service services/login

Deixe o código assim



A primeira setinha referencia o tipo do conteúdo, no caso não tipamos uma model, vamos utilizar o ‘any’, logo acessamos o atributo, e ali utilizamos a variável de ambiente.

Nota\*

Lembre-se, sempre que criar uma service, você deve injetar seu código ao módulo, como fazer isso? É Simples! Entre no arquivo ‘app.module.ts’

E adicione as seguintes linhas



Agora vamos injetar ao typescript da tela de login e fazer a chamada ao método.



Note que injetamos a classe no construtor.

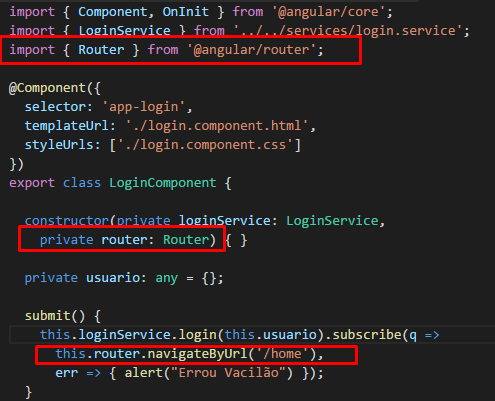
No método submit, quando chamamos o método da service, passamos como parâmetro o objeto de usuário (login), e assim chamamos a observable pegando seu retorno no primeiro paramêtro como q (sucesso), e o segundo err (erro).

Para testar a funcionalidade caso esteja usando o backend desse exemplo, coloque a senha diferente de ‘123’ para ver o erro.

Vamos criar uma nova página para brincar.

ng g component pages/home

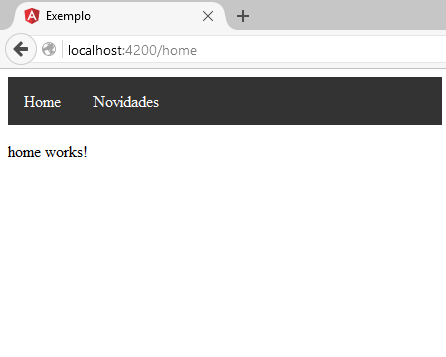
Agora vamos injetar o módulo de rotas na página de login para redirecionar o cliente a home.



Vamos criar a rota da home.



Coloque a senha como ‘123’ para visualizar sua home page.



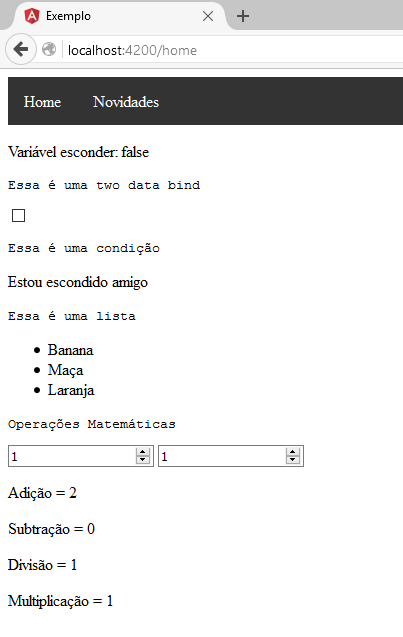
Caso esteja vendo essa mensagem significa que está tudo indo bem.

Vamos testar algumas funcionalidades nessa página.

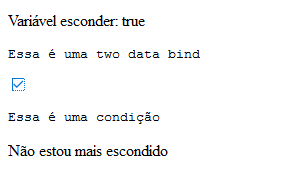
Como o If e o for



O resultado deverá ser assim



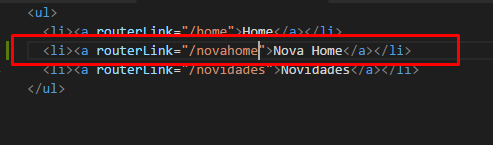
Veja a diferença ao clicar no checkbox

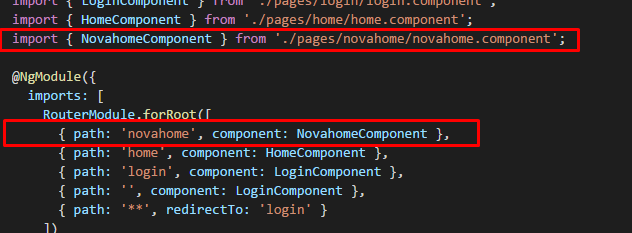


Vamos treinar novos atributos e eventos, repita o procedimento anterior no castro de rotas, porem dê o nome de novahome.

ng g component pages/novahome

Estando criado, você vai adicionar a nova home no menu com o arquivo menu.component.html





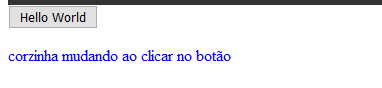
Agora no typescript da nova tela, adicione o seguinte código:



No Html:



O html retornará como:



Ao clicar:

