

# Tutorial Spring Boot, JPA, Hibernate, H2, Rest, Angular e Boot Strap.

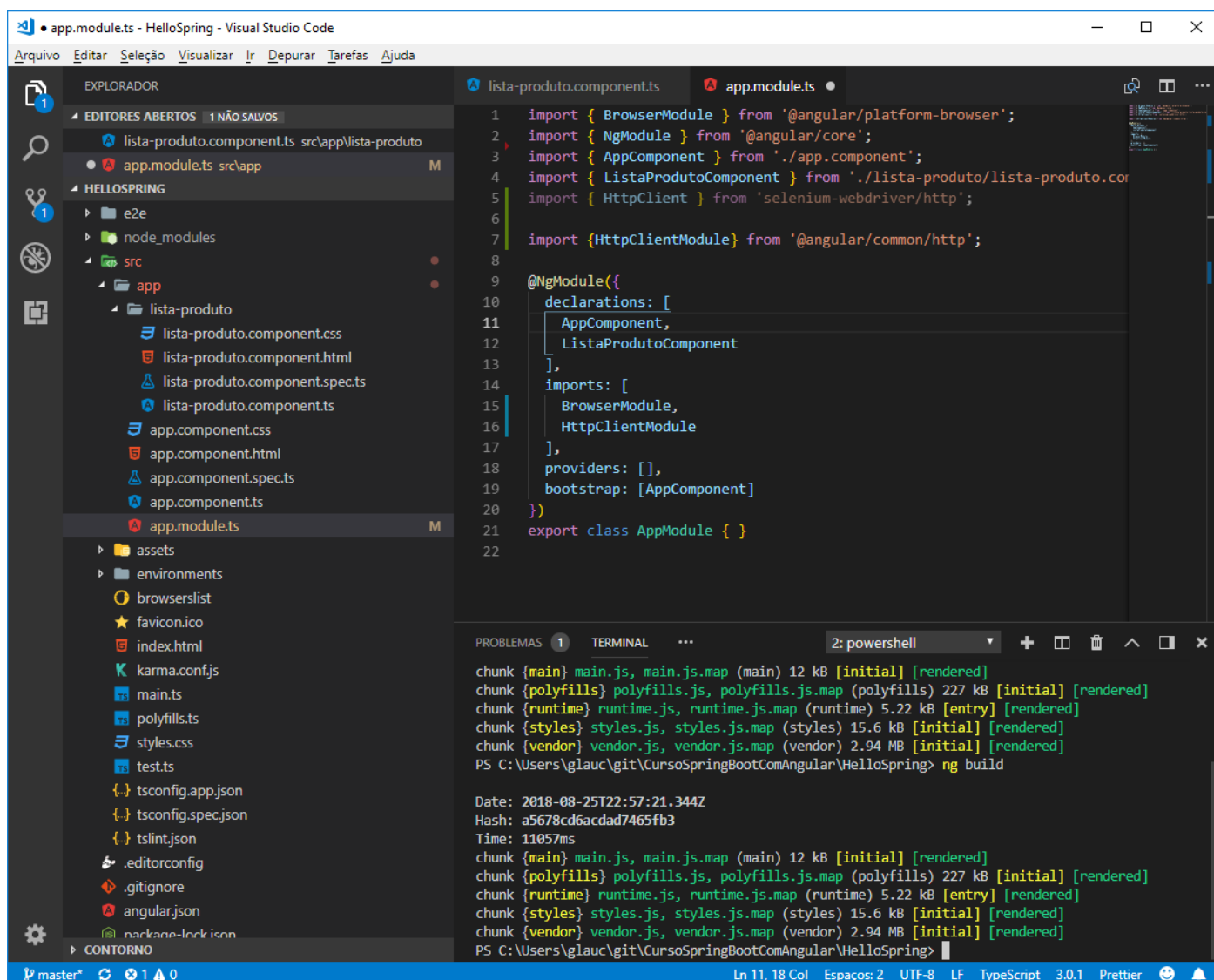
## Parte 6

Glauco Todesco

Download:

<https://github.com/glaucotodesco/CursoSpringBootComAngular/tree/Tutorial-Parte6>

1. Com o componente definido precisamos agora consumir o serviço REST já criado. A primeira etapa é inclusão do módulo HTTP no nosso projeto. Altere o arquivo app.module.ts para fazer isso. As linhas 7 e 16 foram inseridas para fazer essa configuração:



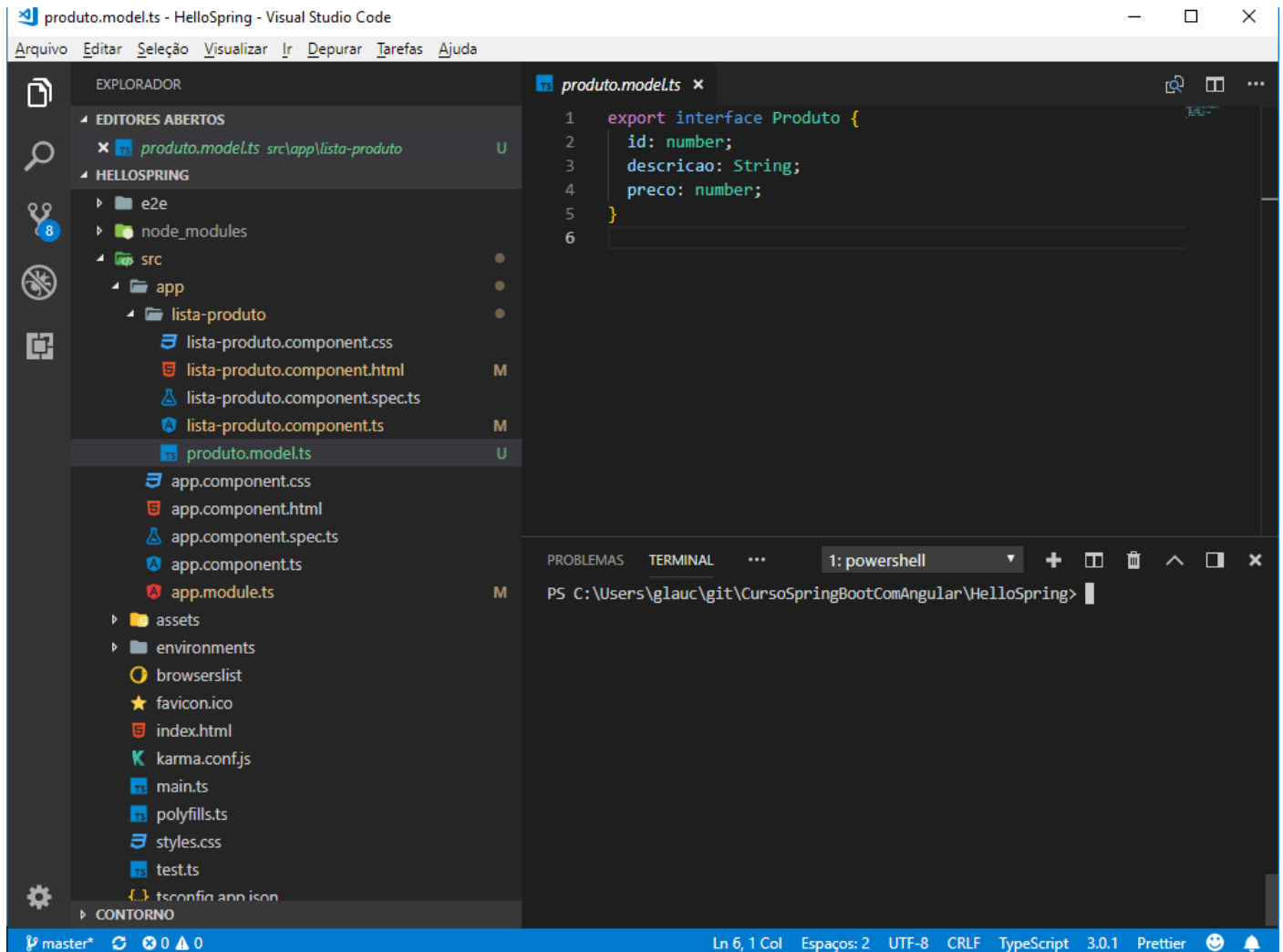
The screenshot shows the Visual Studio Code interface with the file explorer on the left, the editor in the center, and the terminal at the bottom. The file explorer shows the project structure with the 'app' folder expanded, highlighting 'app.module.ts'. The editor displays the content of 'app.module.ts' with the following code:

```
1 import { BrowserModule } from '@angular/platform-browser';
2 import { NgModule } from '@angular/core';
3 import { AppComponent } from './app.component';
4 import { ListaProdutoComponent } from './lista-produto/lista-produto.component';
5 import { HttpClientModule } from '@angular/common/http';
6
7 import { HttpClientModule } from '@angular/common/http';
8
9 @NgModule({
10   declarations: [
11     AppComponent,
12     ListaProdutoComponent
13   ],
14   imports: [
15     BrowserModule,
16     HttpClientModule
17   ],
18   providers: [],
19   bootstrap: [AppComponent]
20 })
21 export class AppModule { }
22
```

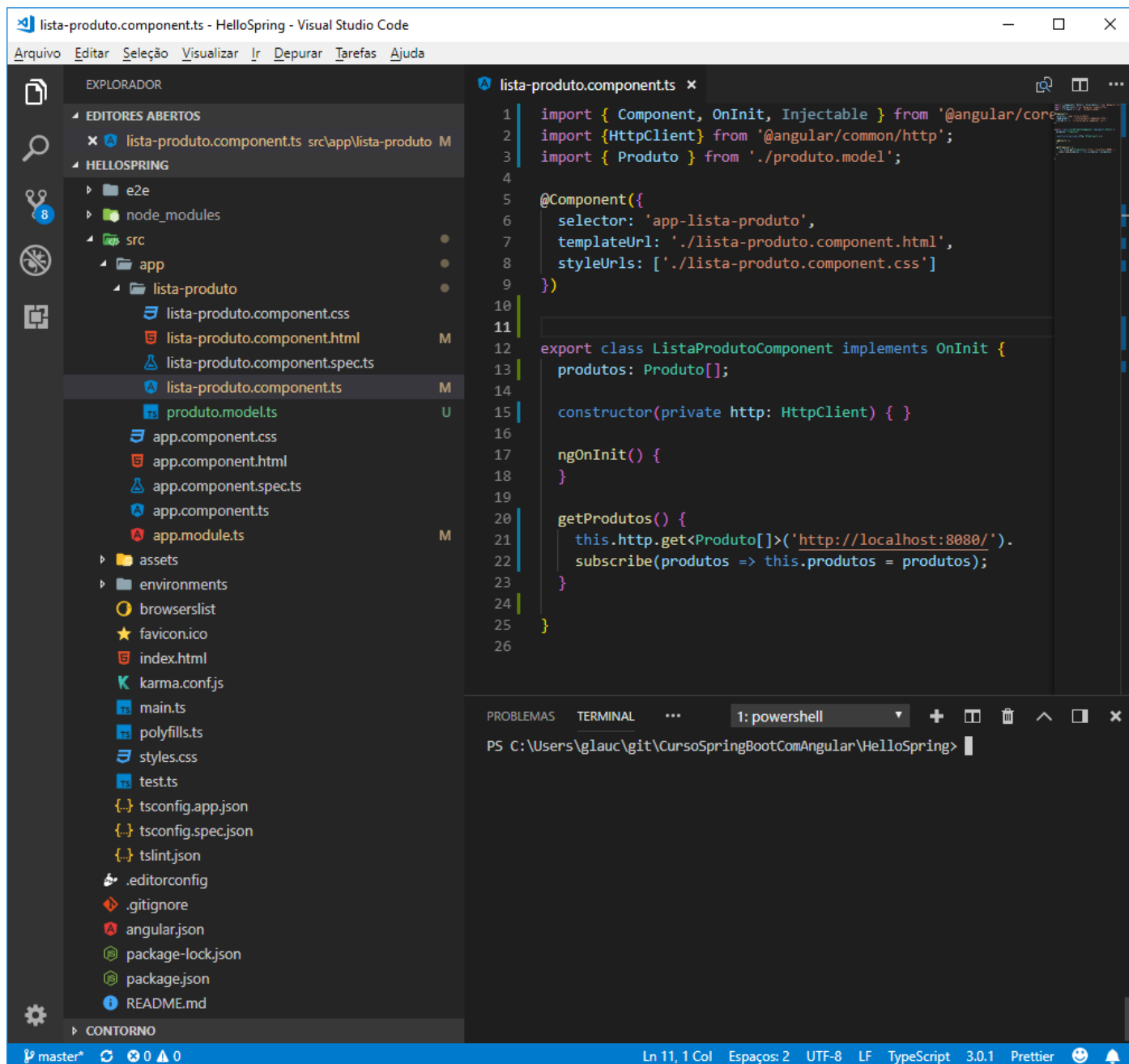
The terminal at the bottom shows the output of the 'ng build' command, indicating a successful build. The status bar at the bottom indicates the file is at line 11, column 18, and the editor is using TypeScript 3.0.1 with the Prettier formatter.

2. Vamos criar uma interface que representa um produto, criando o arquivo **produto.model.ts**. Esse arquivo deve ficar dentro da pasta do nosso componente:

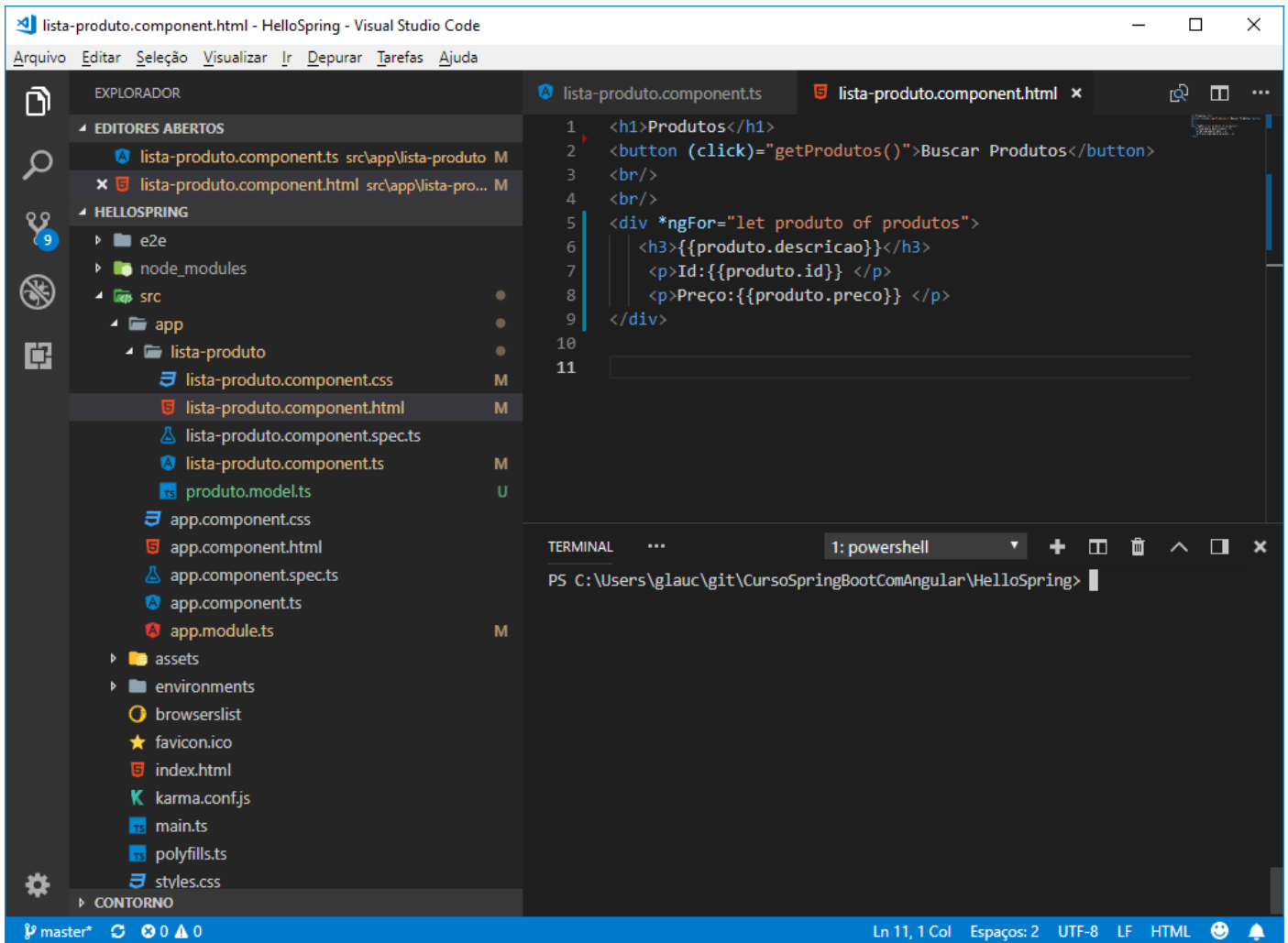
```
export interface Produto {  
  id: number;  
  descricao: String;  
  preco: number;  
}
```



3. A próxima etapa é usar esse a interface de produto dentro da classe no nosso componente (linhas 3 e 13), alterar o construtor do componente para receber o serviço de HTTP (linha 15) e alterar o método `getProdutos` para fazer a chamada http ao nosso serviço de rest (linhas 20, 21 e 22 e 23).



4. A última etapa é alterar o template do componente para mostrar todos os dados do produto (id, descrição e preço).



```
1 <h1>Produtos</h1>
2 <button (click)="getProdutos()">Buscar Produtos</button>
3 <br/>
4 <br/>
5 <div *ngFor="let produto of produtos">
6   <h3>{{produto.descricao}}</h3>
7   <p>Id:{{produto.id}} </p>
8   <p>Preço:{{produto.preco}} </p>
9 </div>
10
11
```

5. Faça o build da aplicação e ao fazer a chamada pelo navegador teremos o seguinte resultado:

