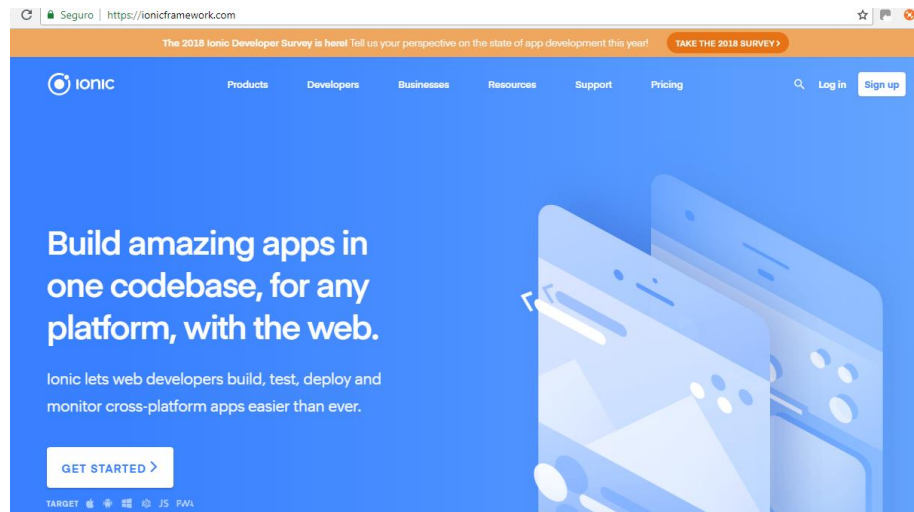


# 1

# IONIC

## MODIFICANDO O PROJETO

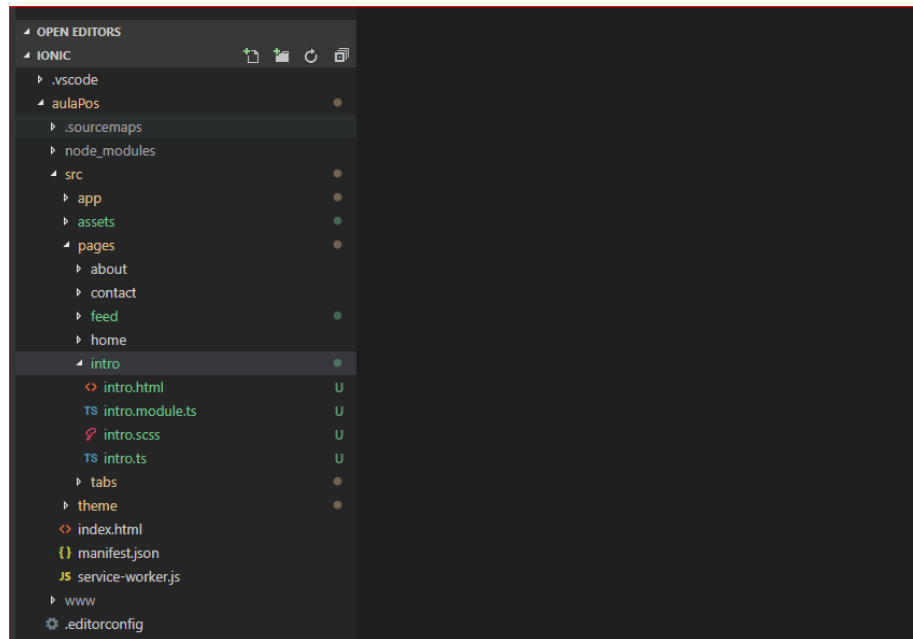


# CRIANDO PÁGINAS

# CODIFICANDO

## PASSOS

- ▶ Iremos utilizar o ionic cli para criar páginas:
- ▶ *Ionic generate page intro*



# CODIFICANDO

## PASSOS

- ▶ Ir na pagina de componentes e escolher o “SLIDES”

The screenshot displays the Ionic documentation page for the **Slides** component. On the left is a sidebar menu with categories like Icons, Inputs, Lists, Loading, Menus, Modals, Navigation, Popover, Radio, Range, Searchbar, Segment, Select, **Slides** (highlighted), Tabs, Toast, Toggle, Toolbar, API, NATIVE, STORAGE, THEMING, IONICONS, and CLI.

The main content area is titled "Slides" and includes the following text:

Slides make it easy to create galleries, tutorials, and page-based layouts. Slides take a number of configuration options on the `<ion-slides>` component.

*For more information, Check out the [Slides API docs](#) and the [Slide API docs](#).*

**Basic Usage**

```
<ion-slides pager>

  <ion-slide style="background-color: green">
    <h2>Slide 1</h2>
  </ion-slide>

  <ion-slide style="background-color: blue">
    <h2>Slide 2</h2>
  </ion-slide>

  <ion-slide style="background-color: red">
    <h2>Slide 3</h2>
  </ion-slide>

</ion-slides>
```

On the right side of the main content area, there is a link for [Demo Source](#).

At the bottom right, there is a preview of a mobile app interface. The app has a status bar at the top showing "12:34 PM" and "100%" battery. The title bar says "Slides". The main content area shows a slide with a blue eye icon and the text "Welcome to the Docs!". Below the slide, it says "The Ionic Component Documentation showcases a number of useful components that are included out of the box with Ionic." There are navigation dots at the bottom of the slide.

# CODIFICANDO

## PASSOS

- ▶ Colocar o código no html do “intro”

Icons

Inputs

Lists

Loading

Menus

Modals

Navigation

Popover

Radio

Range

Searchbar

Segment

Select

Slides

Tabs

Toast

Toggle

Toolbar

API

NATIVE

STORAGE

THEMING

IONICONS

CLI

### Slides

Slides make it easy to create galleries, tutorials, and page-based layouts. Slides take a number of configuration options on the `<ion-slides>` component.

For more information, Check out the [Slides API docs](#) and the [Slide API docs](#).

#### Basic Usage

```
<ion-slides pager>

  <ion-slide style="background-color: green">
    <h2>Slide 1</h2>
  </ion-slide>

  <ion-slide style="background-color: blue">
    <h2>Slide 2</h2>
  </ion-slide>

  <ion-slide style="background-color: red">
    <h2>Slide 3</h2>
  </ion-slide>

</ion-slides>
```

[Demo Source](#)

iOS Android Windows

# CODIFICANDO

## PASSOS

- ▶ Importar a classe de intro.module.ts “IntroPageModulo” para a app em app.module.ts

```
import { NgModule, ErrorHandler } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { IonicApp, IonicModule, IonicErrorHandler } from 'ionic-angular';
import { MyApp } from './app.component';

import { AboutPage } from '../pages/about/about';
import { ContactPage } from '../pages/contact/contact';
import { HomePage } from '../pages/home/home';
import { TabsPage } from '../pages/tabs/tabs';

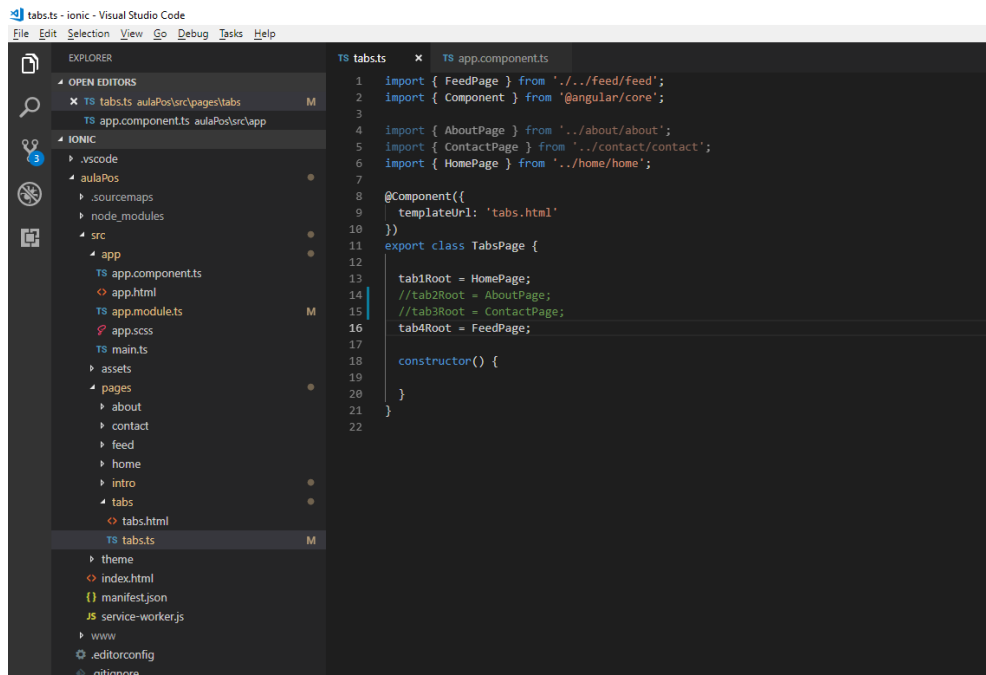
import { StatusBar } from '@ionic-native/status-bar';
import { SplashScreen } from '@ionic-native/splash-screen';
import { FeedPageModule } from '../pages/feed/feed.module';
import { IntroPageModule } from '../pages/intro/intro.module';

@NgModule({
  declarations: [
    MyApp,
    AboutPage,
    ContactPage,
    HomePage,
    TabsPage
  ],
  imports: [
    BrowserModule,
    IonicModule.forRoot(MyApp),
    FeedPageModule,
    IntroPageModule
  ],
  bootstrap: [IonicApp],
  entryComponents: [
    MyApp,
    AboutPage,
    ContactPage,
    HomePage,
    TabsPage
  ],
  providers: [
```

# CODIFICANDO

## PASSOS

- ▶ Ir na pasta em tabs.ts e remover as páginas que não vamos utilizar



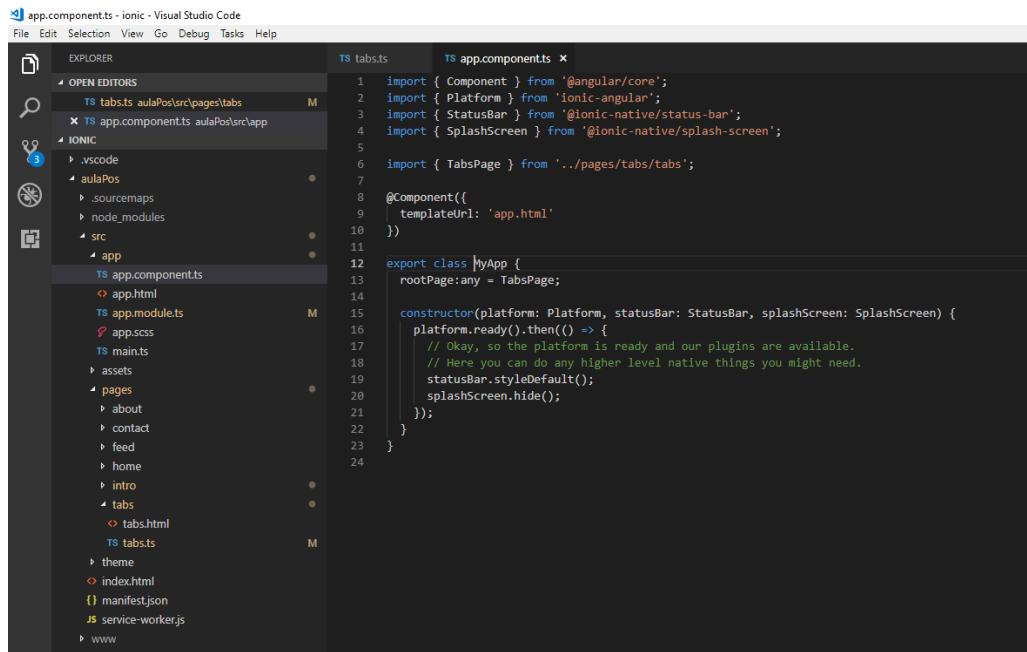
The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays the project structure. The 'src' folder is expanded, showing subfolders like 'app', 'assets', 'pages', and 'theme'. The 'app' folder is further expanded, showing 'app.component.ts', 'app.html', 'app.modules.ts', and 'app.scss'. The 'pages' folder is also expanded, showing 'about', 'contact', 'feed', 'home', 'intro', 'tabs', and 'theme'. The 'tabs' folder is selected, and its contents are shown in the main editor area. The file 'tabs.ts' is open, displaying the following TypeScript code:

```
1 import { FeedPage } from '../feed/feed';
2 import { Component } from '@angular/core';
3
4 import { AboutPage } from '../about/about';
5 import { ContactPage } from '../contact/contact';
6 import { HomePage } from '../home/home';
7
8 @Component({
9   templateUrl: 'tabs.html'
10 })
11 export class TabsPage {
12
13   tab1Root = HomePage;
14   //tab2Root = AboutPage;
15   //tab3Root = ContactPage;
16   tab4Root = FeedPage;
17
18   constructor() {
19
20   }
21 }
22
```

# CODIFICANDO

## PASSOS

- ▶ Ir na `app.componente.ts` e definir a pagina inicial em `rootPage`



The screenshot shows the Visual Studio Code interface with the file explorer on the left and the `app.component.ts` file open in the editor. The file explorer shows the project structure, including the `src` folder and the `app` component. The `app.component.ts` file is selected, and its content is displayed in the editor. The code defines the `MyApp` class, which is the root of the application. It imports necessary modules from `@angular/core`, `ionic-angular`, `@ionic-native/status-bar`, `@ionic-native/splash-screen`, and `ionic-native/tabs`. The `MyApp` class has a `rootPage` property set to `TabsPage` and a `constructor` that initializes the application, including the platform, status bar, and splash screen.

```
app.component.ts - ionic - Visual Studio Code
File Edit Selection View Go Debug Tasks Help

EXPLORER
├── OPEN EDITORS
│   ├── TS tabs.ts aulaPos\src\pages\tabs M
│   └── TS app.component.ts aulaPos\src\app
├── IONIC
│   ├── .vscode
│   ├── aulaPos
│   │   ├── sourcemaps
│   │   └── node_modules
│   └── src
│       ├── app
│       │   ├── app.component.ts
│       │   ├── app.html
│       │   ├── app.module.ts M
│       │   ├── app.scss
│       │   ├── main.ts
│       │   ├── assets
│       │   ├── pages
│       │   │   ├── about
│       │   │   ├── contact
│       │   │   ├── feed
│       │   │   ├── home
│       │   │   ├── intro
│       │   │   ├── tabs
│       │   │   ├── tabs.html
│       │   │   └── tabs.ts M
│       ├── theme
│       ├── index.html
│       ├── manifest.json
│       └── service-worker.js
└── www

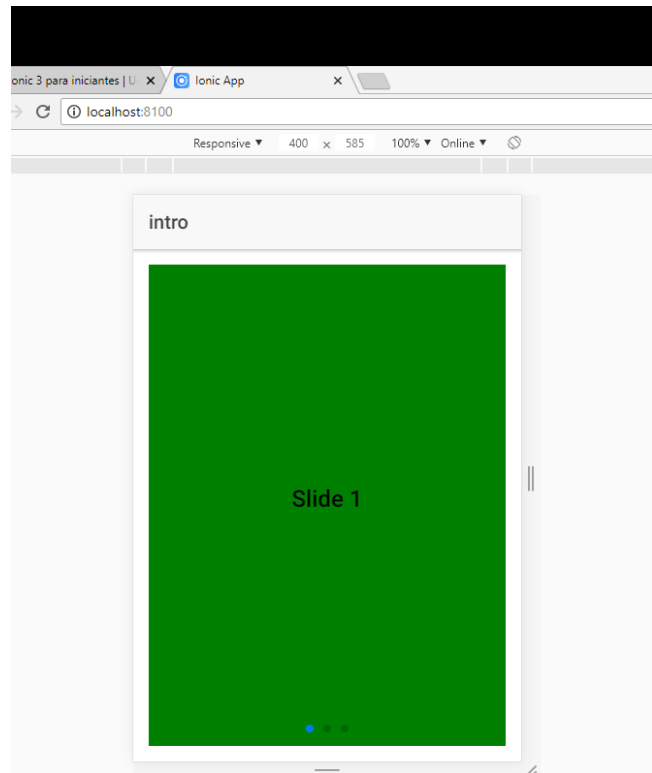
TS tabs.ts
1 import { Component } from '@angular/core';
2 import { Platform } from 'ionic-angular';
3 import { StatusBar } from '@ionic-native/status-bar';
4 import { SplashScreen } from '@ionic-native/splash-screen';
5
6 import { TabsPage } from '../pages/tabs/tabs';
7
8 @Component({
9   templateUrl: 'app.html'
10 })
11
12 export class MyApp {
13   rootPage:any = TabsPage;
14
15   constructor(platform: Platform, statusBar: StatusBar, splashScreen: SplashScreen) {
16     platform.ready().then(() => {
17       // Okay, so the platform is ready and our plugins are available.
18       // Here you can do any higher level native things you might need.
19       statusBar.styleDefault();
20       splashScreen.hide();
21     });
22   }
23 }
24
```



# CODIFICANDO

## PASSOS

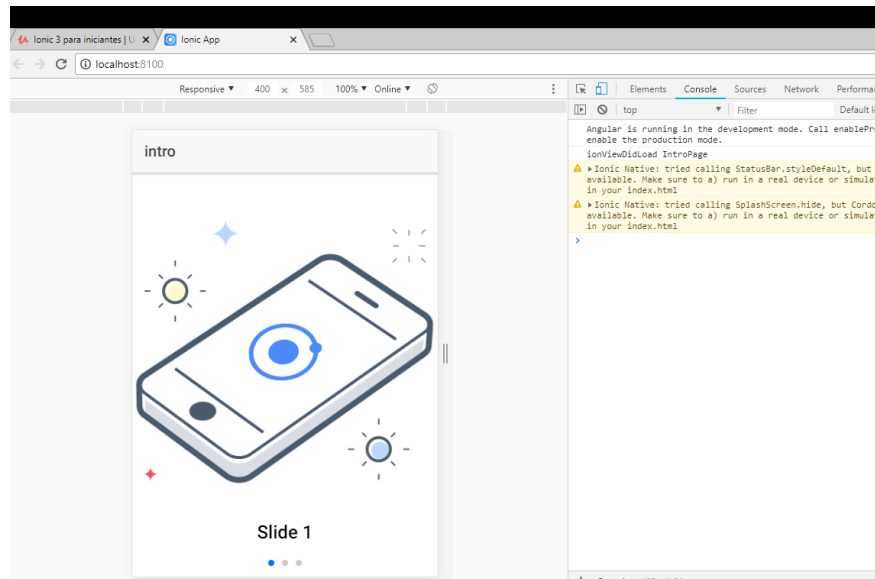
- ▶ Executar a aplicação com o ionic serve
- ▶ *(remover o padding e o title)*



# CODIFICANDO

## PASSOS

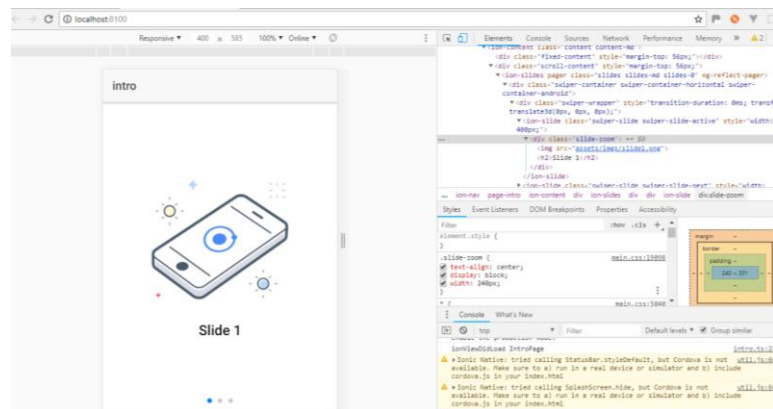
- ▶ Adicionar imagens que necessite para o sistema ionic funcionar



# CODIFICANDO

## PASSOS

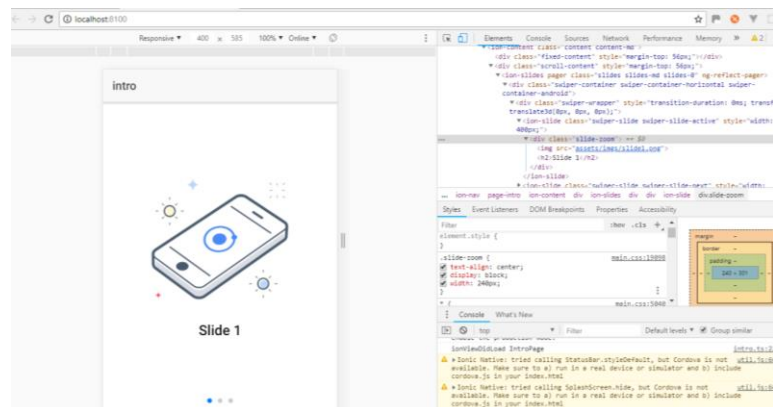
- ▶ Ajustar o tamanho da imagem, altera a classe para que altera em todos



# CODIFICANDO

## PASSOS

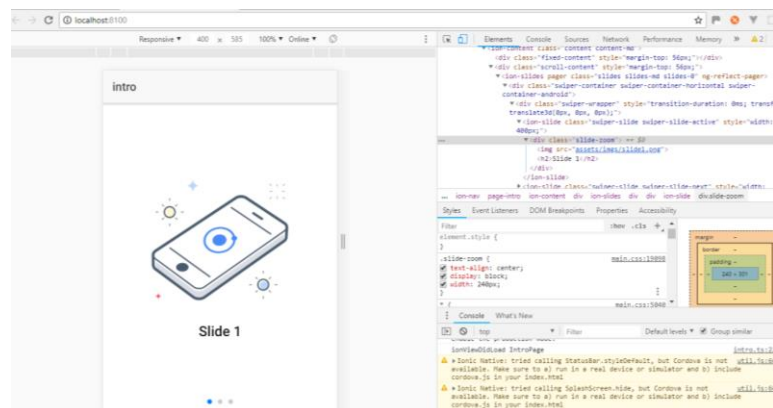
- ▶ Modificar a pagina para a forma que quiser. Conferindo as mudanças direto no celular.



# CODIFICANDO

## PASSOS

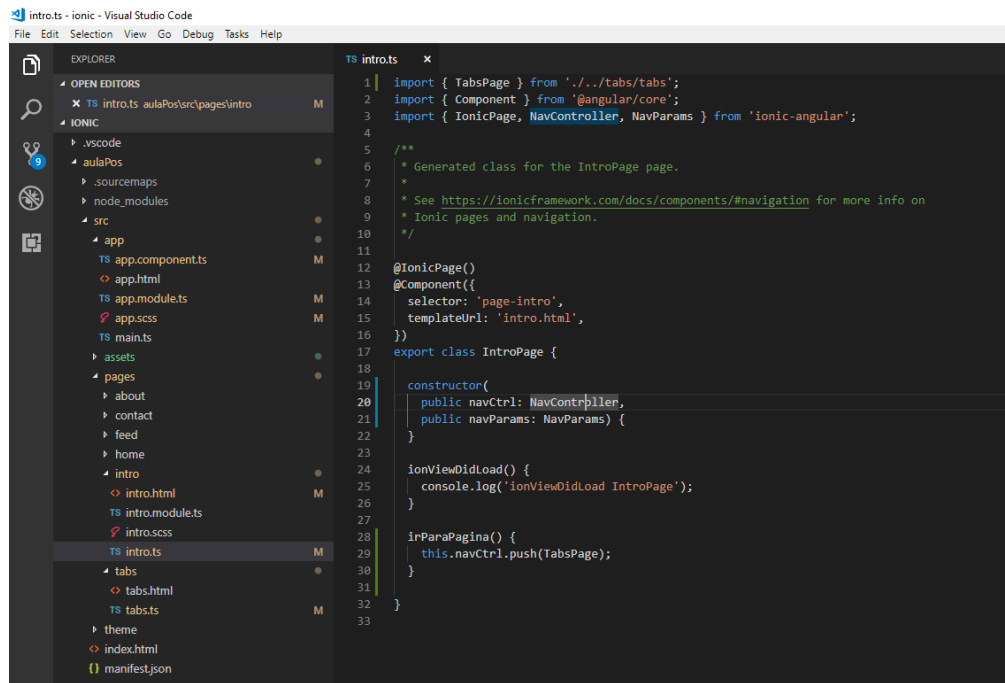
- ▶ Colocar mais um slide



# CODIFICANDO

## PASSOS

- ▶ Fazer a navegação da pagina com o Typescript



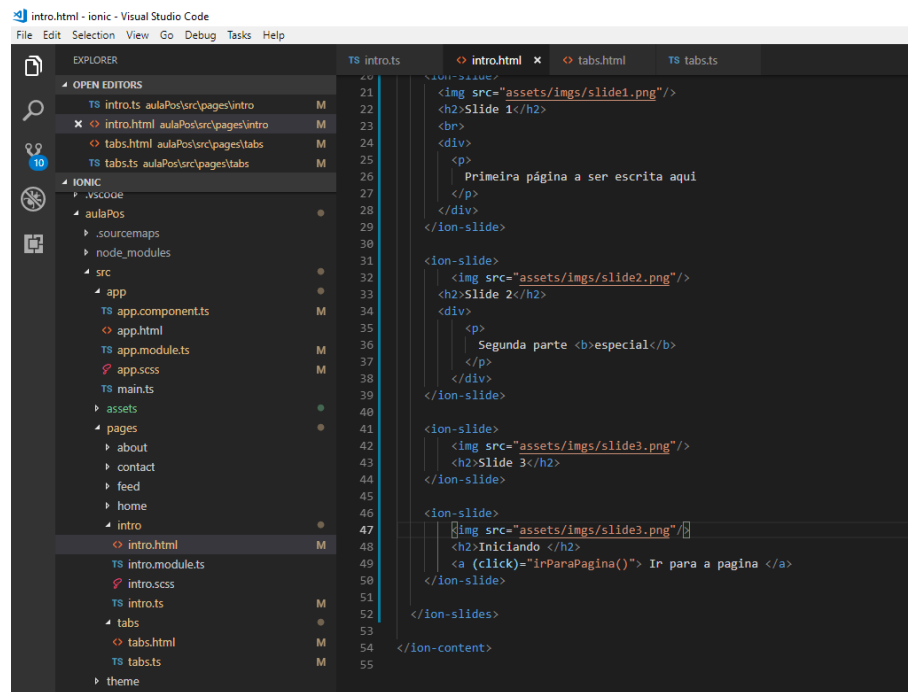
The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays the project structure for 'intro.ts - ionic - Visual Studio Code'. The file 'intro.ts' is selected under the 'pages' directory. The main editor area shows the TypeScript code for the 'IntroPage' component. The code includes imports for 'TabPage', 'Component', 'IonicPage', 'NavController', and 'NavParams'. It features a class decorator '@IonicPage()', a '@Component()' decorator with 'selector' and 'templateUrl' properties, and an 'export class IntroPage' definition. The class has a constructor with 'NavController' and 'NavParams' parameters, an 'ionViewDidLoad()' method that logs a message, and an 'ionParaPagina()' method that pushes the 'TabPage' to the navigation stack.

```
1 import { TabPage } from '../tabs/tabs';
2 import { Component } from '@angular/core';
3 import { IonicPage, NavController, NavParams } from 'ionic-angular';
4
5 /**
6  * Generated class for the IntroPage page.
7  *
8  * See https://ionicframework.com/docs/components/#navigation for more info on
9  * Ionic pages and navigation.
10 */
11
12 @IonicPage()
13 @Component({
14   selector: 'page-intro',
15   templateUrl: 'intro.html',
16 })
17 export class IntroPage {
18
19   constructor(
20     public navCtrl: NavController,
21     public NavParams: NavParams) {
22   }
23
24   ionViewDidLoad() {
25     console.log('ionViewDidLoad IntroPage');
26   }
27
28   ionParaPagina() {
29     this.navCtrl.push(TabPage);
30   }
31 }
32
33
```

# CODIFICANDO

## PASSOS

- ▶ Colocar o elemento de navegação <a>



```
intro.html - ionic - Visual Studio Code
File Edit Selection View Go Debug Tasks Help

EXPLORER
├─ OPEN EDITORS
│   ├── TS intro.ts aulaPos/src/pages/intro
│   ├── intro.html aulaPos/src/pages/intro
│   ├── tabs.html aulaPos/src/pages/tabs
│   └── TS tabs.ts aulaPos/src/pages/tabs
├─ IONIC
│   ├── .vscode
│   ├── aulaPos
│   │   ├── .sourcemaps
│   │   └── node_modules
│   ├── src
│   │   ├── app
│   │   │   ├── TS app.component.ts
│   │   │   ├── app.html
│   │   │   ├── TS app.module.ts
│   │   │   ├── app.scss
│   │   │   └── TS main.ts
│   │   ├── assets
│   │   ├── pages
│   │   │   ├── about
│   │   │   ├── contact
│   │   │   ├── feed
│   │   │   ├── home
│   │   │   └── intro
│   │   │       ├── intro.html
│   │   │       ├── TS intro.module.ts
│   │   │       ├── TS intro.scss
│   │   │       ├── TS intro.ts
│   │   │       ├── tabs
│   │   │       │   ├── tabs.html
│   │   │       │   └── TS tabs.ts
│   │   └── theme
```

```
20 </ion-slide>
21 
22 <h2>Slide 1</h2>
23 <br>
24 <div>
25   <p>
26     Primeira página a ser escrita aqui
27   </p>
28 </div>
29 </ion-slide>
30
31 <ion-slide>
32   
33   <h2>Slide 2</h2>
34   <div>
35     <p>
36       Segunda parte <b>especial</b>
37     </p>
38   </div>
39 </ion-slide>
40
41 <ion-slide>
42   
43   <h2>Slide 3</h2>
44 </ion-slide>
45
46 <ion-slide>
47   
48   <h2>Iniciando </h2>
49   <a (click)="irParaPagina()"> Ir para a pagina </a>
50 </ion-slide>
51 </ion-slides>
52
53 </ion-content>
54
55
```