

ionic3.x+ 目录结构分析、创建组件、创建页面、页面跳转

主讲教师：（大地）

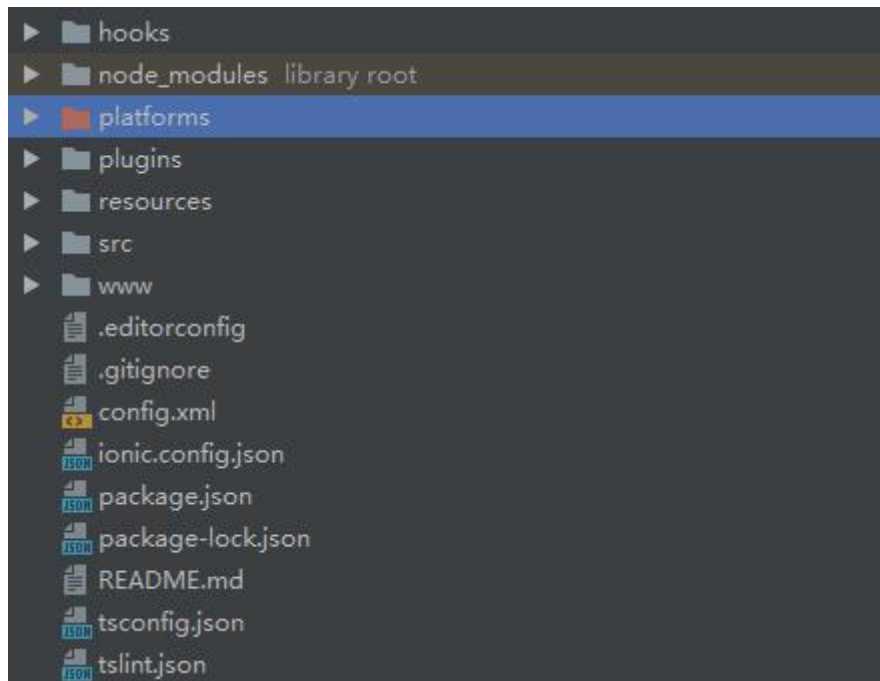
合作网站：www.itying.com （IT 营）

合作网站：www.ionic.wang （ionic 中文网）

目录

一、 Ionic 目录结构分析.....	2
二、 Ionic src 下面文件分析.....	3
三、 app.module.ts 分析.....	4
四、 Ionic 创建组件.....	6
五、 Ionic 创建页面以及页面跳转.....	8

一、Ionic3.x 目录结构分析



hooks: 编译 cordova 时自定义的脚本命令，方便整合到我们的编译系统和版本控制系统中

node_modules : node 各类依赖包

resources : android/ios 资源（更换图标和启动动画）

src: 开发工作目录，页面、样式、脚本和图片都放在这个目录下

www: 静态文件

platforms: 生成 android 或者 ios 安装包路径（`platforms\android\build\outputs\apk`: apk 所在位置）执行 `cordova platform add android` 后会生成

plugins: 插件文件夹，里面放置各种 cordova 安装的插件

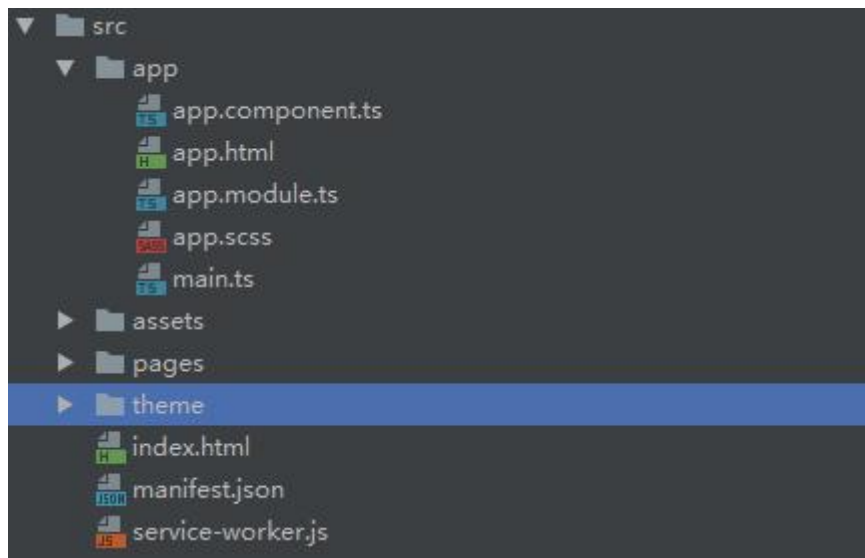
config.xml: 打包成 app 的配置文件

package.json: 配置项目的元数据和管理项目所需要的依赖

tsconfig.json: TypeScript 项目的根目录，指定用来编译这个项目的根文件和编译选项

tslint.json: 格式化和校验 typescript

src 工作目录:



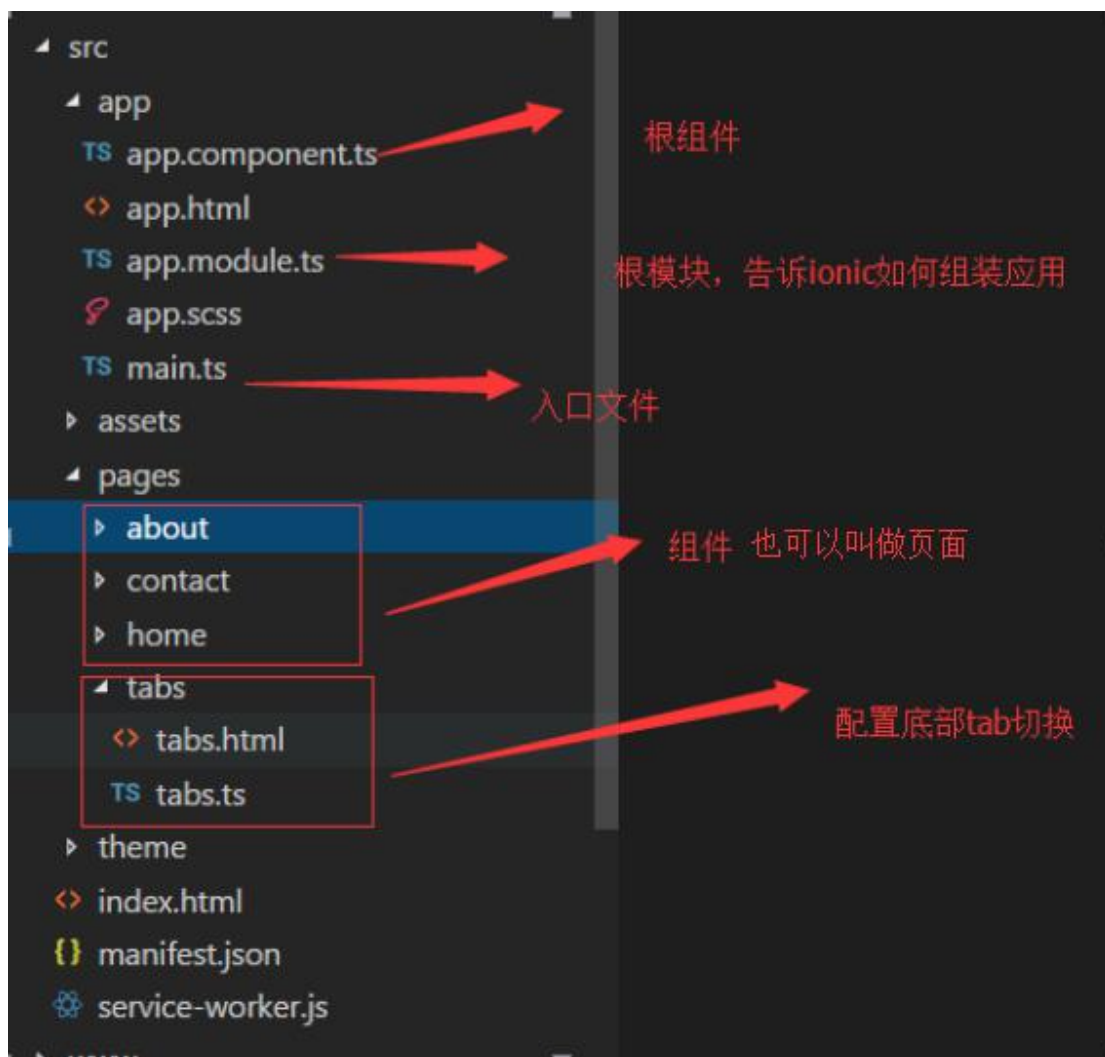
app: 应用根目录

assets: 资源目录（静态文件（图片，js 框架。。。）各

pages: 页面文件，放置编写的页面文件，包括：html，scss，ts。

theme: 主题文件，里面有一个 scss 文件，设置主题信息。

二、Ionic3.x src 下面文件分析



三、app.module.ts 分析

```
//这个根模块会告诉 ionic 如何组装该应用
import { NgModule, ErrorHandler } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { ComponentsModule } from '../components/components.module';
import { IonicApp, IonicModule, IonicErrorHandler } from 'ionic-angular';
import { MyApp } from './app.component'; /*根组件*/
//其他组件
```

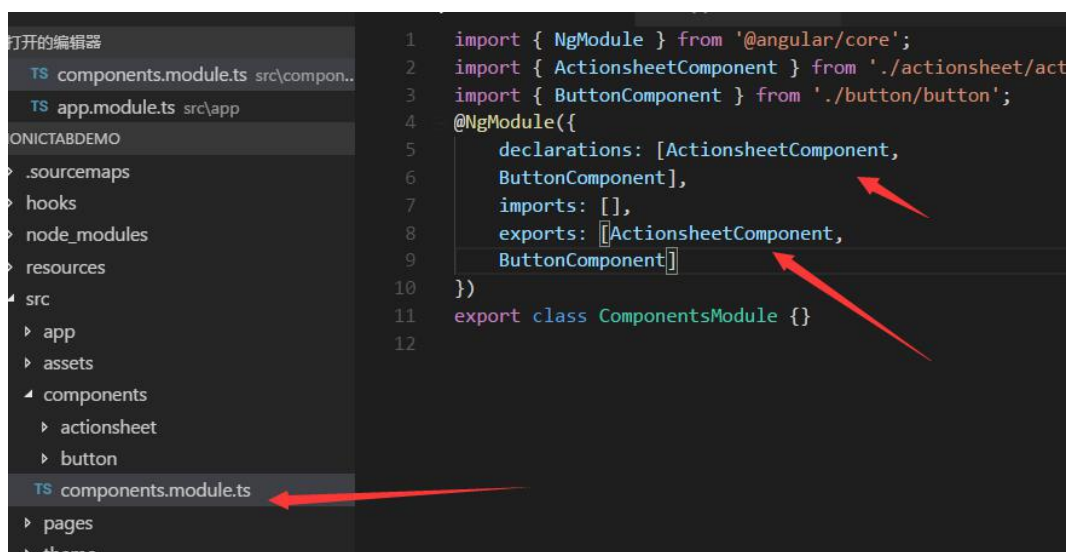
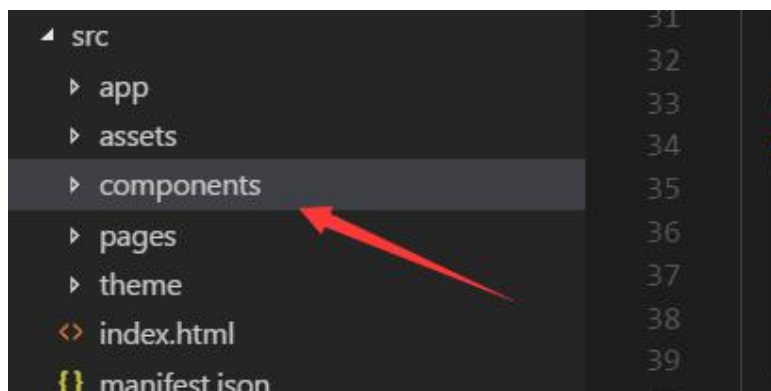
```
import { AboutPage } from '../pages/about/about';
import { ContactPage } from '../pages/contact/contact';
import { HomePage } from '../pages/home/home';
import { NewsPage } from '../pages/news/news';
//底部 tab 切换
import { TabsPage } from '../pages/tabs/tabs';
import { StatusBar } from '@ionic-native/status-bar';
import { SplashScreen } from '@ionic-native/splash-screen';
@NgModule({
  declarations: [
    MyApp,
    AboutPage,
    ContactPage,
    HomePage,
    NewsPage,
    TabsPage
  ],
  imports: [
    BrowserModule,
    ComponentsModule,
    IonicModule.forRoot(MyApp)
  ],
  bootstrap: [IonicApp],
  entryComponents: [ /*配置不会在模板中使用的组件*/
    MyApp,
    AboutPage,
    ContactPage,
    HomePage,
    NewsPage,
    TabsPage
  ],
  providers: [
    StatusBar,
    SplashScreen,
    {provide: ErrorHandler, useClass: IonicErrorHandler}
  ]
})
export class AppModule {}
```

四、Ionic3.x 创建组件

- 1、cd 到我们的项目目录
- 2、通过 **ionic g component** 组件名称创建组件

```
C:\cordova\ionic_tabs_demo02>ionic g component actionsheet  
[OK] Generated a component named actionsheet!
```

- 3、创建完成组件以后会在 **src** 目录下面多一个 **components** 的目录，这个目录里面有我们用命令创建的所有的组件。



- 4、如果我们要使用这些组件必须在 **app.module.ts** 里面注册我们的模块，注册完成后就可

以在 **pages** 里面的其页面里面使用这些组件。

```
1 //这个根模块会告诉 ionic 如何组装该应用
2 import { NgModule, ErrorHandler } from '@angular/core';
3 import { BrowserModule } from '@angular/platform-browser';
4
5 import { ComponentsModule } from '../components/components.module';
6
7 import { IonicApp, IonicModule, IonicErrorHandler } from 'ionic-angular';
8 import { MyApp } from './app.component'; /*根组件*/
9
10 //其他组件
11 import { AboutPage } from '../pages/about/about';
12 import { ContactPage } from '../pages/contact/contact';
13 import { HomePage } from '../pages/home/home';
14 //底部tab切换
15 import { TabsPage } from '../pages/tabs/tabs';
16
17 import { StatusBar } from '@ionic-native/status-bar';
```

```
1 <ion-header>
2   <ion-navbar>
3     <ion-title>
4       About
5     </ion-title>
6   </ion-navbar>
7 </ion-header>
8
9 <ion-content padding>
10   <actionsheet></actionsheet>
11   about 页面
12 </ion-content>
13
```

五、Ionic3.x 创建页面以及页面跳转

```
C:\cordova\ionicTabDemo>ionic g page news  
[OK] Generated a page named news!
```

```
<button ion-button (click)='pushButton()'>执行Button</button>  
</ion-content>
```

```
import { NavController } from 'ionic-angular';  
  
import { NewsPage } from '../news/news';  
@Component({  
  selector: 'page-about',  
  templateUrl: 'about.html'  
})  
export class AboutPage {  
  
  constructor(public navCtrl: NavController) {  
  
  }  
  
  pushButton(){  
    this.navCtrl.push(NewsPage);  
  }  
}
```