## Angular前段框架

<https://angular.cn/start>

## node.js相关指令

查看路径：

where node

查看版本号：

node-v

## vscode运行angular

<https://blog.csdn.net/luanxiyuan/article/details/81171618>

<https://blog.csdn.net/maiowu1001/article/details/76600558>

## Angular,ng-zorro文档学习

## 一．安装

**1.安装node.js 安装npm**

**2.使用 npm 包管理器来安装 CLI**

npm install -g @angular/cli

**3.创建新的项目并运行**

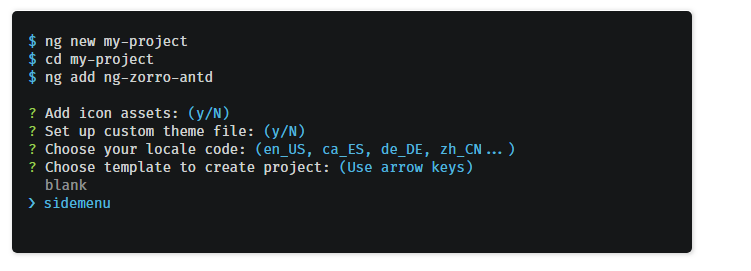
ng new my-first-project

cd my-first-project

ng serve

1. **安装ng-zorro**

**Ng add ng-zorro-anto**



## 路由



路由配置文件



## Cli指令

生成组件

ng generate component test

## 项目文件结构

路由，页面跳转，模板数据显示，接口请求

## 二．angula服务

<Http://a.itying.com/api/productlist>

1. 安装

ng g service servicename

Or ng g service /目录/servicename

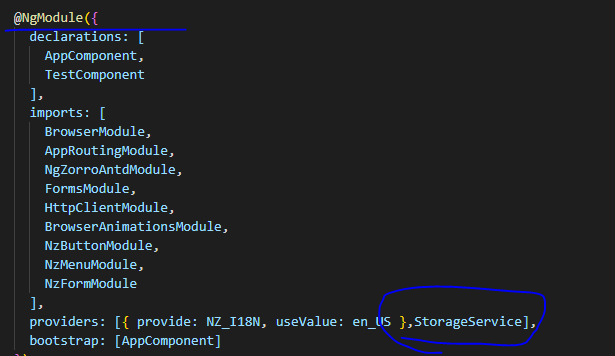
1. imort 依赖

在中引入

//引入 配置服务

import  {StorageService} from './services/storage.service';

1. 在providers中声明

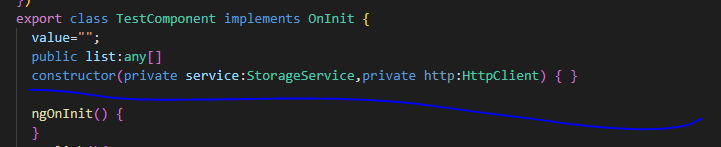


1. 在组件中使用

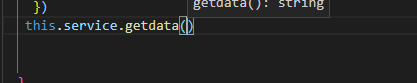
引入配置



依赖注入



调用服务中方法



## 接口请求

跨域配置：

### HttpClient

//请求放于服务

#### Get

    let url="http://a.itying.com/api/productlist";

    this.service.getrequest(url).subscribe(data=>{

       console.log(data)

       this.list=data['result']

    })

服务

  getrequest(url){

    // let url="http://a.itying.com/api/productlist";

    //

      return this.http.get(url)

  }

#### Post

 var url = "/api/angulartest";

    this.service.postrequest(url).subscribe(response => {

      console.log(response);

    });

服务

  postrequest(url){

    const httpOptions = {

      headers: new HttpHeaders({

         'Content-Type': 'application/json'

        })

    };

#### JsonP

### 第三方模块axios

Axios 使用文档

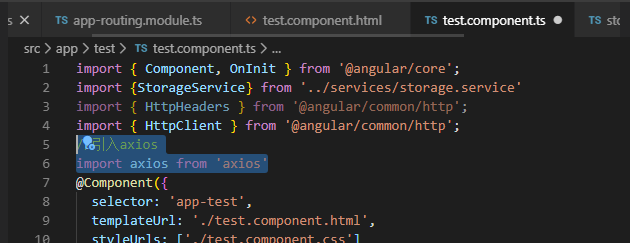
<https://www.kancloud.cn/yunye/axios/234845>

A：安装 npm install axios --save

B: 引入

//引入axios

import axios from 'axios'



C：使用

    axios.get('/user?ID=12345')

  .then(function (response) {

    console.log(response);

  })

  .catch(function (error) {

    console.log(error);

  });

法2

    axios({

      method: ’get’,

      url: url,

    }).then(res=>{

      console.log(res)

    }).catch(err=>{

      console.log(err)

    });

    }

Post 封装

 axiosPost(api){

      return new Promise((resolve,reject)=>{

              // 发送 POST 请求

              axios({

                method: 'post',

                url: '/user/12345',

                data: {

                  firstName: 'Fred',

                  lastName: 'Flintstone'

                }

              }).then(res=>{

                resolve(res)

              }).catch(err=>{

                reject(err)

              });

      })

    }

 axiosgetClick(){

    let url="http://a.itying.com/api/productlist";

    this.service.axiosGet(url).then(res=>console.log(res)).catch(err=>console.log(err))

  }

bug记录 laravel 安装没有开扩展报错

### angular跨域配置

[https://angular.cn/guide/build#proxying-to-a-backend-server](https://angular.cn/guide/build" \l "proxying-to-a-backend-server)



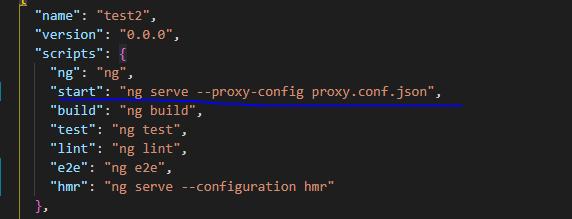




,

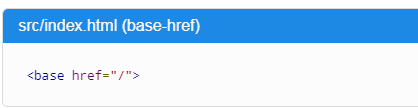
      // "changeOrigin": true

Angular 运行时读取你的代理

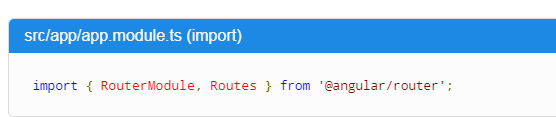


## 路由

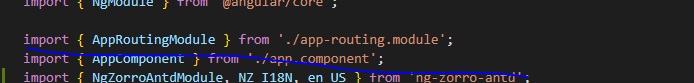
### 设置路由导航根路径

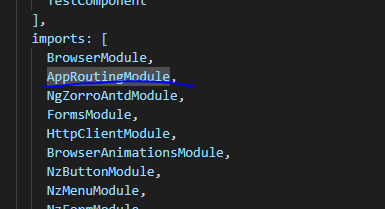


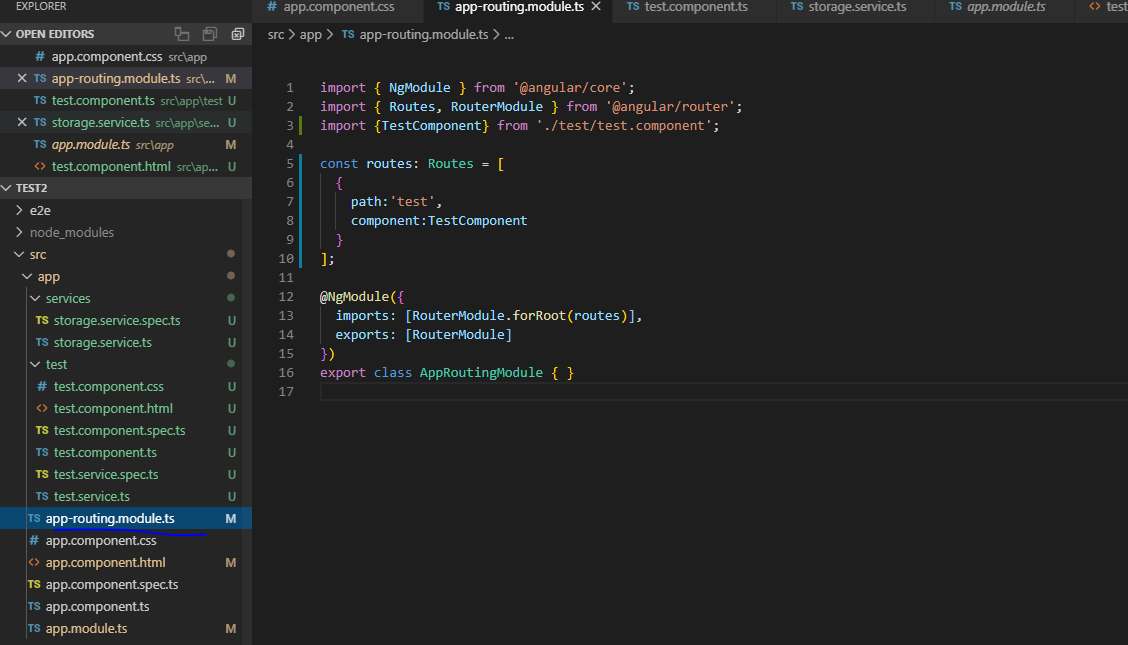
### 导入路由



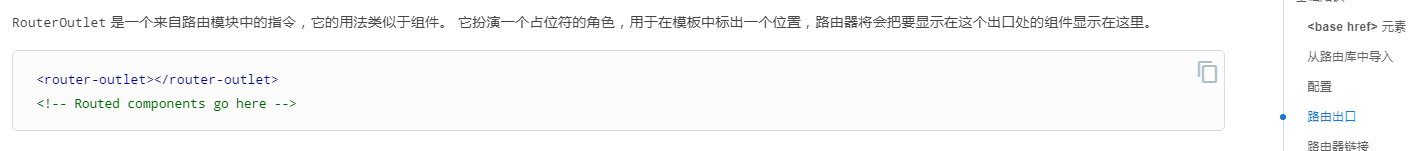
### 配置路由







### 路由出口



<[router-outlet](https://angular.cn/api/router/RouterOutlet)></[router-outlet](https://angular.cn/api/router/RouterOutlet)>

在模板中占位，将要显示的组件显示在这

### 路由链接

<[a](https://angular.cn/api/router/RouterLinkWithHref) [routerLink](https://angular.cn/api/router/RouterLink)="/crisis-center" [routerLinkActive](https://angular.cn/api/router/RouterLinkActive)="active">Crisis Center</[a](https://angular.cn/api/router/RouterLinkWithHref)> <[a](https://angular.cn/api/router/RouterLinkWithHref) [routerLink](https://angular.cn/api/router/RouterLink)="/heroes" [routerLinkActive](https://angular.cn/api/router/RouterLinkActive)="active">Heroes</[a](https://angular.cn/api/router/RouterLinkWithHref)>

### 定义路由通配符过滤非法路由

{ path: **'\*\*'**, component: 要跳转的组件的组件名}

当用户访问未定义的路由时自动跳转到预设的组件

### 路由参数

#### 7.1带参数的路由定义

{ path: 'hero/:id', component: HeroDetailComponent }

访问：

localhost:4200/hero/15

#### 从组件访问含参路由

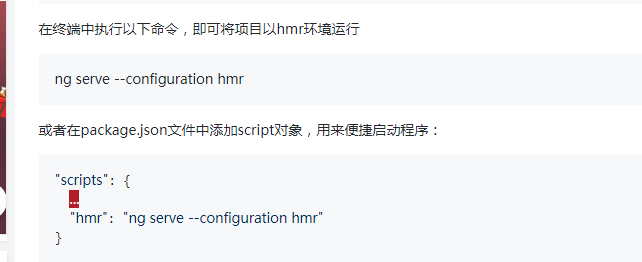
<a [routerLink]="['/hero', hero.id]">

### 子路由



## 安装配置热替换hmr

<https://blog.csdn.net/weixin_34310369/article/details/87958527>

<https://blog.csdn.net/tech_Wang/article/details/80800249>

这步不用配置

运行：

ng serve --hmr

## angular生命周期函数

### 5.1. ngOnInit

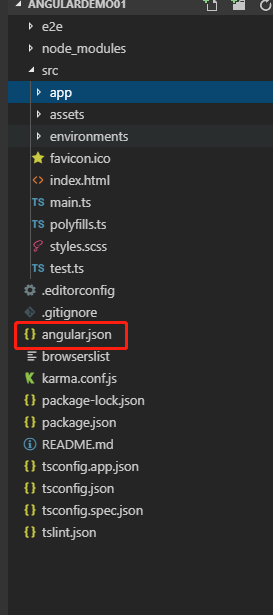
数据初始化

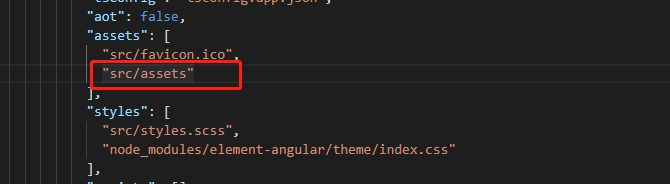
## Angular.json配置

### 6.1图片路径找不到问题

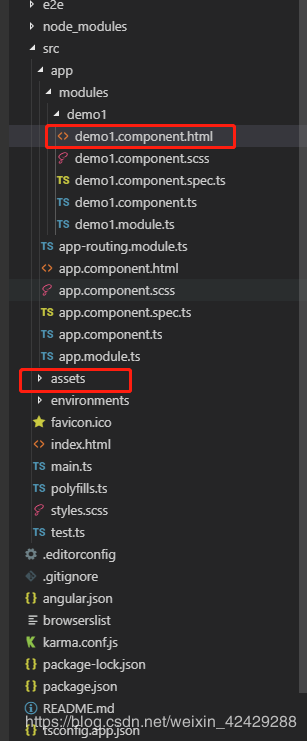
<https://blog.csdn.net/weixin_42429288/article/details/96484668>

确保你的angular.json中静态资源路径的设置如下





然后在组件中，我的项目结构如下，图片都在assets文件夹中





{

"resource": "/d:/angularProject/test2/package.json",

"owner": "\_generated\_diagnostic\_collection\_name\_#0",

"code": "768",

"severity": 4,

"message": "Problems loading reference 'https://schemastore.azurewebsites.net/schemas/json/package.json': Unable to load schema from 'https://schemastore.azurewebsites.net/schemas/json/package.json': read ECONNRESET.",

"startLineNumber": 1,

"startColumn": 1,

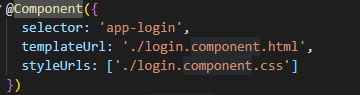
"endLineNumber": 1,

"endColumn": 2

}

## 组件

### 7.1组件嵌套

Cli命令生成组件时，会生成

当在一个组件中引用另一个组件时就直接<app-login></app-login>

## A.angular引入验证码