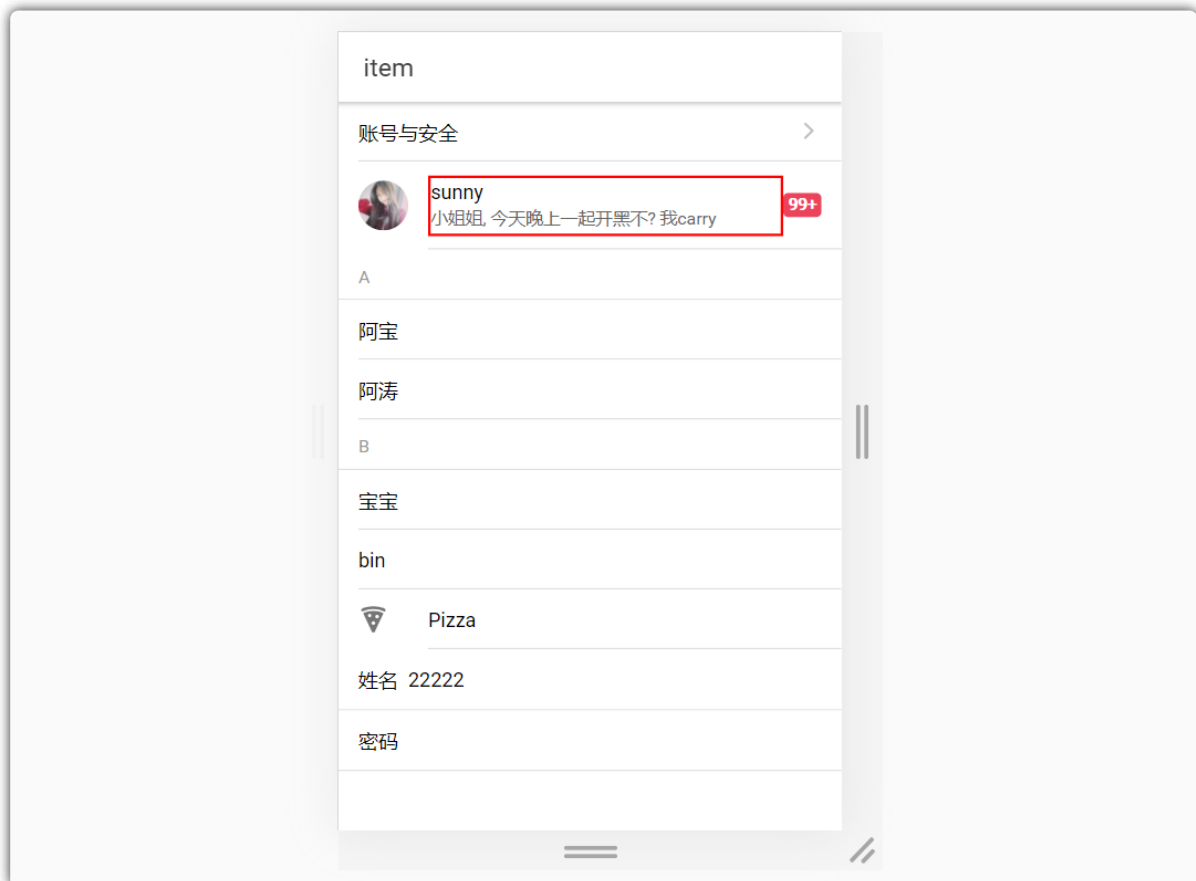


Angular05

组件

item

<https://ionicframework.com/docs/api/item>



```
1
2 <!-- item -->
3 <ion-app>
4   <ion-header>
5     <ion-toolbar>
6       <ion-title>item</ion-title>
7     </ion-toolbar>
8   </ion-header>
9
10  <ion-content>
11    <!-- detail: 添加右箭头 -->
12    <!-- button: 点击效果 -->
13    <ion-item detail button>
14      <ion-label>账号与安全</ion-label>
15    </ion-item>
16
17    <!-- i-item-avatar -->
18    <ion-item>
19      <!-- avatar: 内部图片变圆 -->
20      <!-- slot: 插槽. 左右各有一个; start 和 end -->
```

```
21     <ion-avatar slot="start">
22         
25     </ion-avatar>
26     <!-- ion-label: 是一个带有 flex:1 写法的div. 宽度与剩余空间一致! -->
27     <ion-label style="border: 2px solid red">
28         <h2>sunny</h2>
29         <p>小姐姐, 今天晚上一起开黑不? 我carry</p>
30     </ion-label>
31     <ion-badge color="danger">99+</ion-badge>
32 </ion-item>
33
34 <!-- i-item-group -->
35 <ion-item-group>
36     <ion-item-divider>
37         <ion-label>A</ion-label>
38     </ion-item-divider>
39
40     <ion-item>
41         <ion-label>阿宝</ion-label>
42     </ion-item>
43     <ion-item>
44         <ion-label>阿涛</ion-label>
45     </ion-item>
46
47     <ion-item-divider>
48         <ion-label>B</ion-label>
49     </ion-item-divider>
50
51     <ion-item>
52         <ion-label>宝宝</ion-label>
53     </ion-item>
54     <ion-item>
55         <ion-label>bin</ion-label>
56     </ion-item>
57 </ion-item-group>
58
59 <!-- i-item-icon -->
60 <ion-item>
61     <ion-icon name="pizza" slot="start"></ion-icon>
62     <ion-label>Pizza</ion-label>
63 </ion-item>
64
65 <!-- i-item-input -->
66 <ion-item>
67     <ion-label>姓名</ion-label>
68     <!-- 输入框 -->
69     <ion-input></ion-input>
70 </ion-item>
71
72 <ion-item>
73     <ion-label>密码</ion-label>
```

```

74     <!-- 输入框 -->
75     <ion-input type="password"></ion-input>
76     </ion-item>
77   </ion-content>
78 </ion-app>
79

```

路由系统

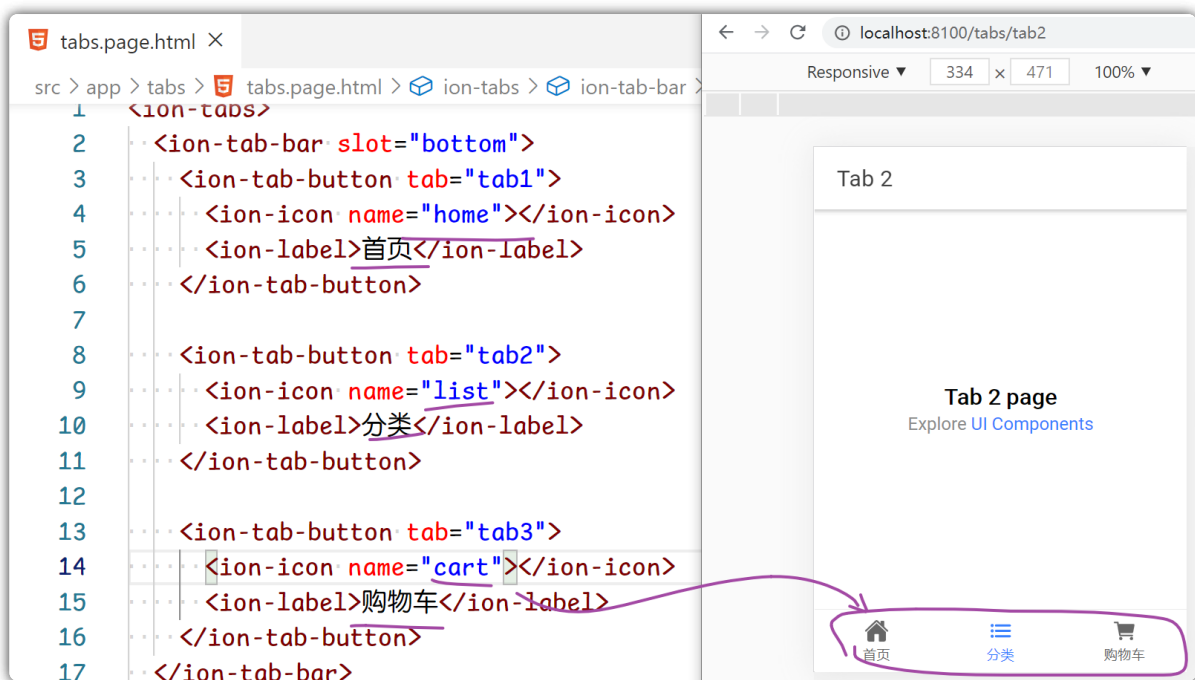
生成带有路由系统的项目包：

```
1 ionic start tabsApp tabs
```

无法自己生成的同学，按照下方操作：

- 到 FTP 的 18_angular 拿 tabs_src.zip
- 关闭vscode中所有打开的页面，然后关闭vscode，把项目包中的 src 改成 src1
- 把 tabs_src.zip 中的 src 文件夹 解压缩到 项目包中
- 重新打开vscode，使用 `ionic s` 命令启动项目即可： **注意关闭之前的服务**

配置标签栏



生成页

组件按照用途 通常分为 两类，例如在 vue 中

views: 路由切换的 页面

components: 存放 复用的 小组件

```
1 ionic g page 页面名
```

例如： `ionic g page tab4`

```
D:\webtn2008\04_Angular\Day05\ionicApp>ionic g page tab4
> ng.cmd generate page tab4 --project=app 实际上执行的命令
CREATE src/app/tab4/tab4-routing.module.ts (339 bytes) 每页都自带 子路由系统
CREATE src/app/tab4/tab4.module.ts (458 bytes) 每页都带有自己的配置文件
CREATE src/app/tab4/tab4.page.html (123 bytes)
CREATE src/app/tab4/tab4.page.spec.ts (633 bytes)
CREATE src/app/tab4/tab4.page.ts (248 bytes)
CREATE src/app/tab4/tab4.page.scss (0 bytes)
UPDATE src/app/app-routing.module.ts (526 bytes) 页面会自动更新到 主路由文件中
[OK] Generated page!
```

练习：新增 tab05

- 新建页面： `ionic g page tab05`
- 删除默认生成的全局路由配置： `app-routing.module.ts`
- 配置 tabs 路由，添加05： `tabs-routing.module.ts`
- 配置 tabs 页面，增加 商品 按钮，对应 tab05： `tabs.page.html`

生成详情页

```
1 ionic g page detail
```

```
1 import { Component, OnInit } from "@angular/core";
2 import { AlertController, NavController } from "@ionic/angular";
3
4 @Component({
5   selector: "app-detail",
6   templateUrl: "./detail.page.html",
7   styleUrls: ["./detail.page.scss"],
8 })
9 export class DetailPage implements OnInit {
10   // AlertController: 弹出框服务
11   constructor(public router: NavController, public alertC: AlertController) {}
12
13   goback() {
14     this.router.back();
15   }
16
17   ngOnInit() {}
18
19   onClick() {
20     // alertC 与 axios 都是 Promise 制作的，
21     // create(): 负责生成弹出框，生成后会通过 then 执行下一步。 res 就是生成的弹出框
22     // res.present(): 是 弹出提示框 的意思
23     this.alertC
24       .create({
25         header: "标题",
26         subHeader: "子标题",
27         message: "具体内容",
```

```

28     buttons: [
29       "确定",
30       "取消",
31       // 通过{} 对象方式, 为按钮添加回调
32       { text: "哈哈", handler: () => console.log("哈哈 按钮点击") },
33     ],
34   })
35   .then((res) => res.present());
36 }
37
38 // 绑定搜索框, 实时获取搜索框的内容
39 onSearchChange(event) {
40   // console.log(event);
41   let value = event.detail.value;
42   console.log(value);
43   // 搭配 搜索接口, 就可以 实时显示 搜索的数据
44 }
45 }
46

```

```

1  <ion-header>
2    <ion-toolbar>
3      <!-- 刷新页面: 路由记录会消失, 导致返回按钮 没有记录, 就不显示返回按钮 -->
4      <!-- 必须点击跳转才能出 返回按钮 -->
5      <!-- 默认: 返回按钮 -->
6      <!-- i-back-button -->
7      <!-- <ion-buttons slot="start">
8        <ion-back-button></ion-back-button>
9      </ion-buttons> -->
10
11     <!-- 自定义按钮 -->
12     <ion-button fill="clear" (click)="goback()" slot="start"> 返回 </ion-button>
13
14     <ion-title>detail</ion-title>
15   </ion-toolbar>
16 </ion-header>
17
18 <ion-content>
19   <ion-button (click)="onClick()" expand="block" fill="clear" shape="round">
20     Click me
21   </ion-button>
22
23   <!-- 搜索框 -->
24   <!-- i-searchbar -->
25   <ion-searchbar
26     placeholder="商品名"
27     inputmode="decimal"
28     type="decimal"
29     (ionChange)="onSearchChange($event)"
30     [debounce]="250"
31     showCancelButton="always"
32   ></ion-searchbar>
33 </ion-content>
34

```

```

1 import { Component } from "@angular/core";
2 import { ActivatedRoute } from "@angular/router";
3
4 @Component({
5   selector: "app-tab2",
6   templateUrl: "tab2.page.html",
7   styleUrls: ["tab2.page.scss"],
8 })
9 export class Tab2Page {
10   /**
11    * vue 接收路由参数 有 两种方式:
12    * * this.$route.params.xxx
13    * * 属性传参: 路由配置中要添加 props:true 的配置项, 就可以通过 props: ['xxx'] 接
    收参数
14    */
15
16    // angular 通过 服务 接收参数
17    constructor(public route: ActivatedRoute) {}
18
19    ngOnInit(): void {
20      console.log(this.route);
21    }
22  }
23

```

Angular收尾项目

目的: 锻炼学员的 **动手+动脑** 能力

资源: [18_Angualr/IonicProjectFiles.zip](#)

放假时间要提前下载 React 的资源

大概 2G.

下周学 RN. 开发真正的手机端项目. 可以编译成 Apk 安装在手机上, 也可以发布到商店上线.

有安卓手机同学: 可以真机调试. 有iPhone的同学 可以换个安卓手机.

链接: https://pan.baidu.com/s/16jrKVqIZ_10H47t1rh4qBw

提取码: 1qc6

复制这段内容后打开百度网盘手机App, 操作更方便哦 -- 来自百度网盘超级会员v4的分享

