# angular配置echarts

npm install echarts -S

npm install ngx-echarts -S

npm install @types/echarts -D

## app.module.ts

在app.module.ts中导入NgxEchartsModule模块

// echarts

import { NgxEchartsModule } from 'ngx-echarts';

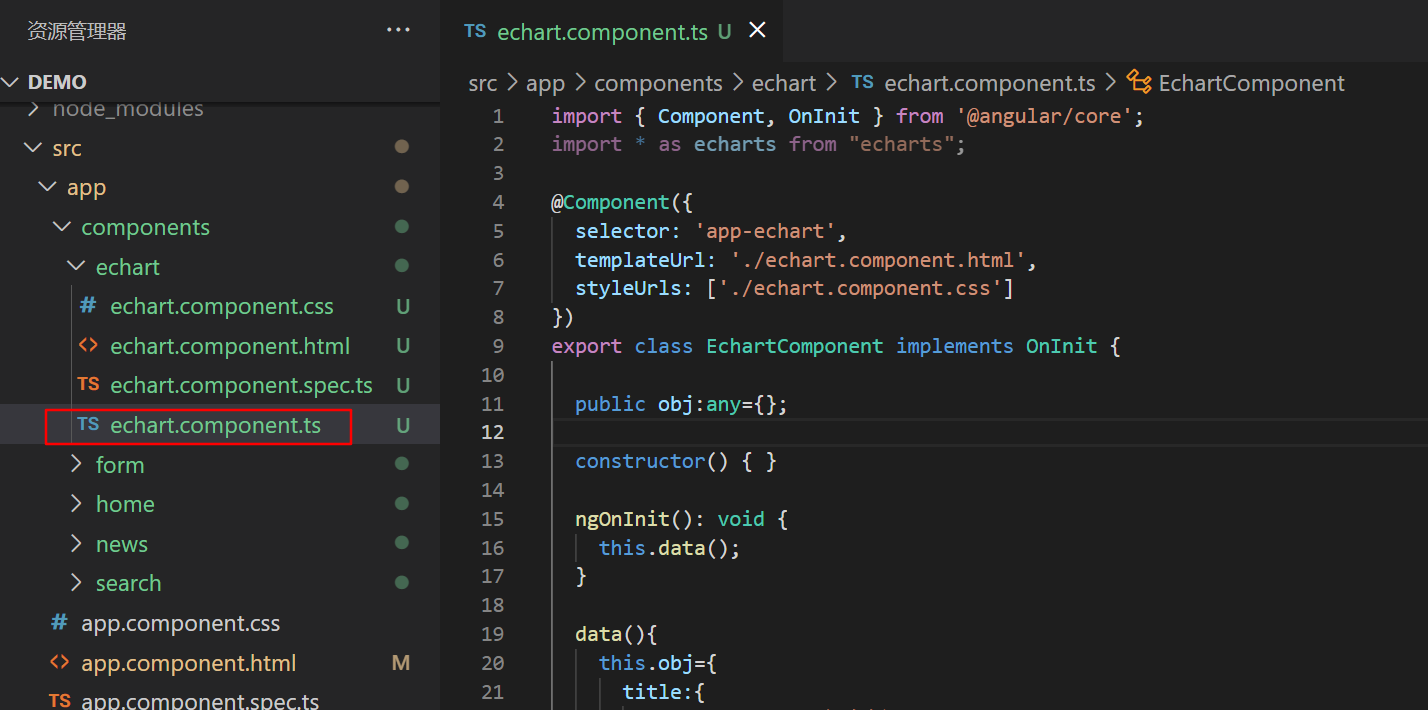
imports: [

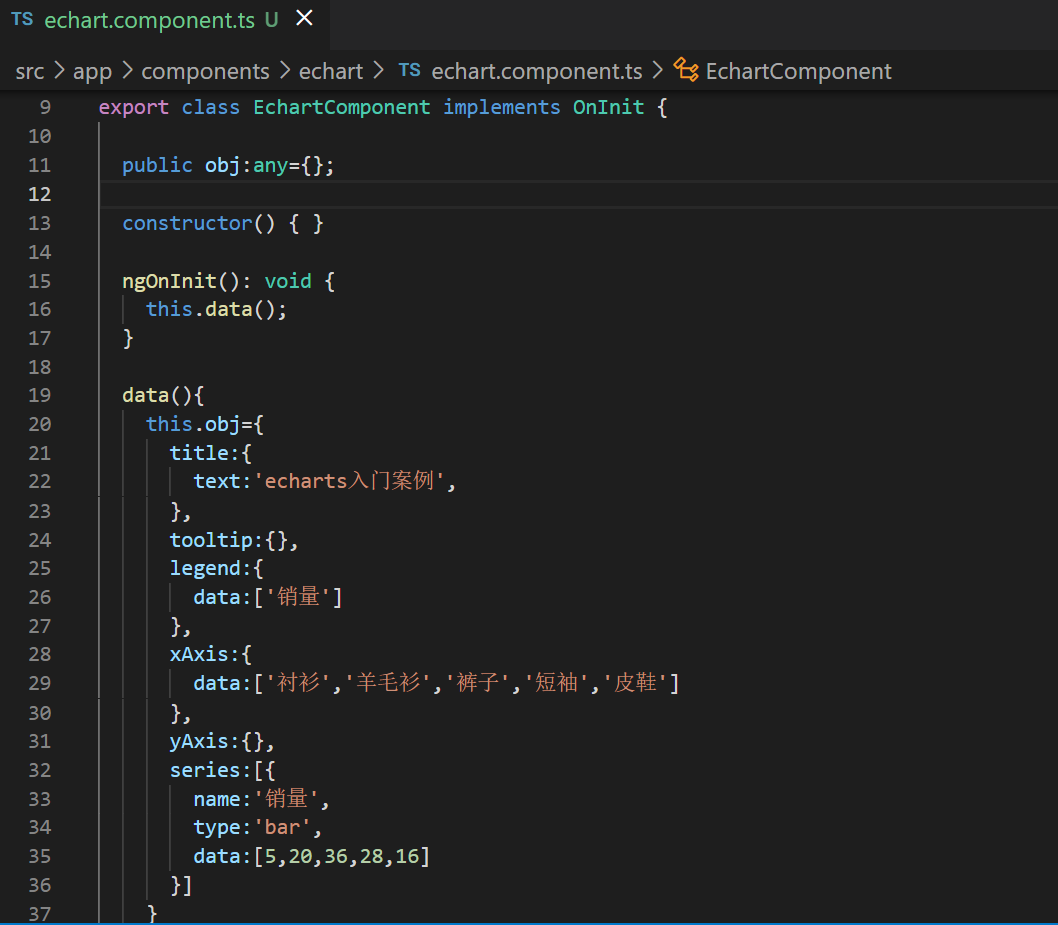
    NgxEchartsModule

  ],

## xxx.component.ts

在对应的component.ts中，添加处理逻辑。如echart.component.ts：

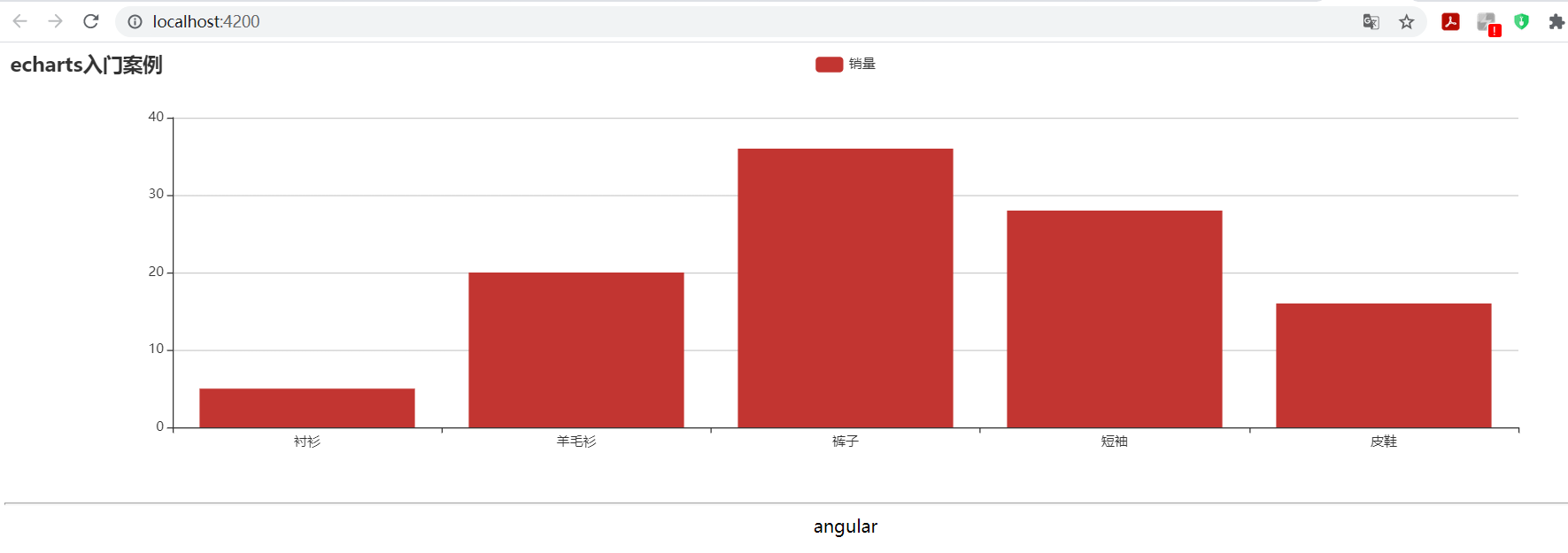




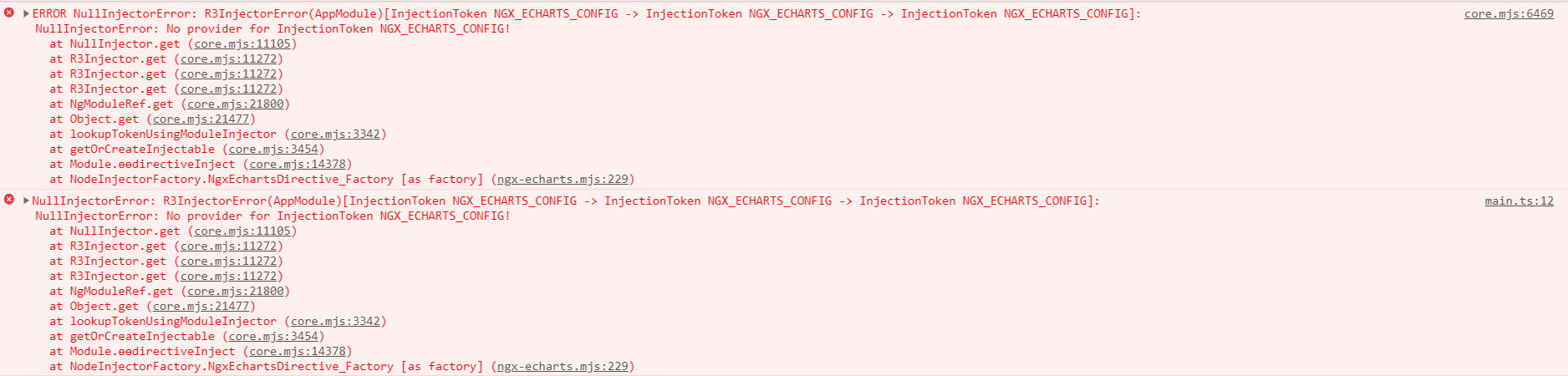
## xxx.component.html

<div echarts [options]="obj"></div>

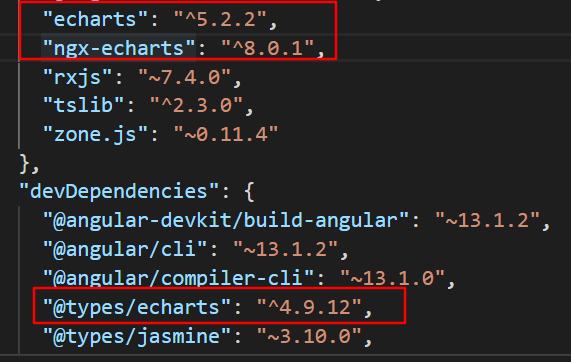
执行ng serve --open运行：



# 解决echarts版本不兼容

刚开始配置时，安装的echarts与ngx-echarts的版本不兼容，报错

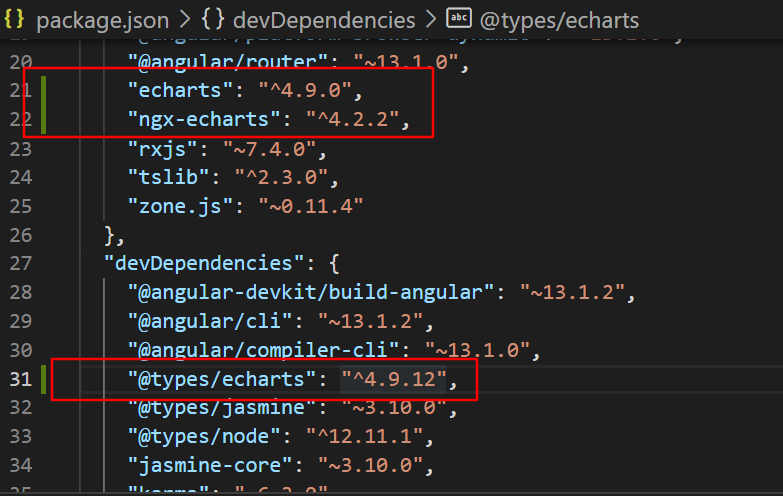
不兼容的版本如下所示：



重新安装合适的依赖

npm install echarts --save

npm install ngx-echarts --save



更换版本后，编译成功。

使用下面的

npm install echarts -S

npm install ngx-echarts -S

npm install @types/echarts -D安装依赖编译失败可能与使用的angular版本有关，我使用的angular版本如下

Angular CLI: 13.1.2

Node: 16.13.1

Package Manager: npm 8.1.2

OS: win32 x64

Angular:

...

Package Version

------------------------------------------------------

@angular-devkit/architect 0.1301.2 (cli-only)

@angular-devkit/core 13.1.2 (cli-only)

@angular-devkit/schematics 13.1.2 (cli-only)

@schematics/angular 13.1.2 (cli-only)

因此推荐使用

npm install echarts --save

npm install ngx-echarts --save

安装合适的依赖版本。

# 搜索缓存实现

实现搜索栏中输入的内容，显示在搜索框下面（同样的内容不会重复出现）

## ts

import { Component, OnInit } from '@angular/core';

@Component({

  selector: 'app-search',

  templateUrl: './search.component.html',

  styleUrls: ['./search.component.css']

})

export class SearchComponent implements OnInit {

  public keyword:string='';

  public historyList:any[]=[];

  constructor() { }

  ngOnInit(): void {

  }

  doSearch(){

    if (this.historyList.indexOf(this.keyword) == -1){

      this.historyList.push(this.keyword);

    }

    this.keyword='';

    //console.log(this.keyword);

  }

}

## html

<div class="search">

    <input type="text" [(ngModel)]="keyword"/> <button (click)="doSearch()">搜索</button>

    <hr />

    <span>搜索历史</span>

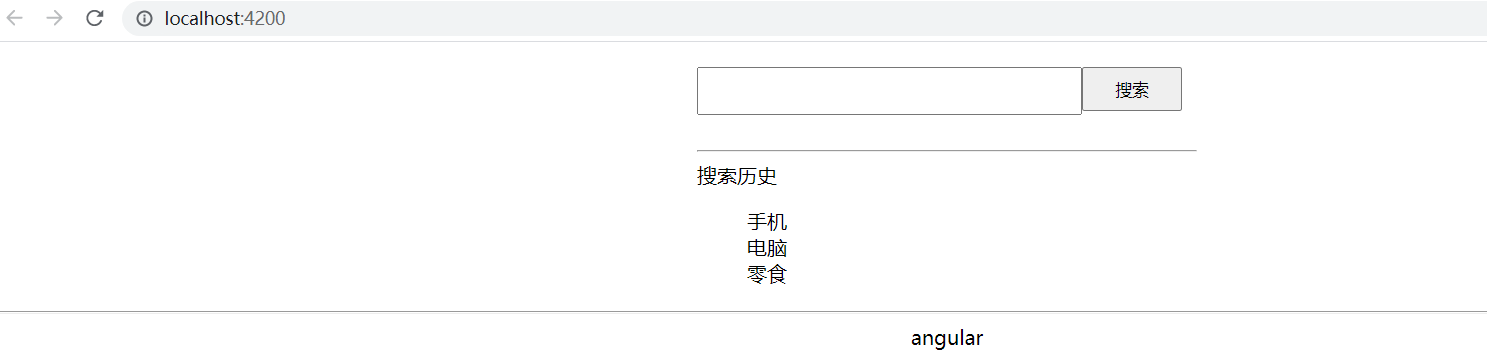
    <ul>

        <li \*ngFor="let item of historyList">{{item}}</li>

    </ul>

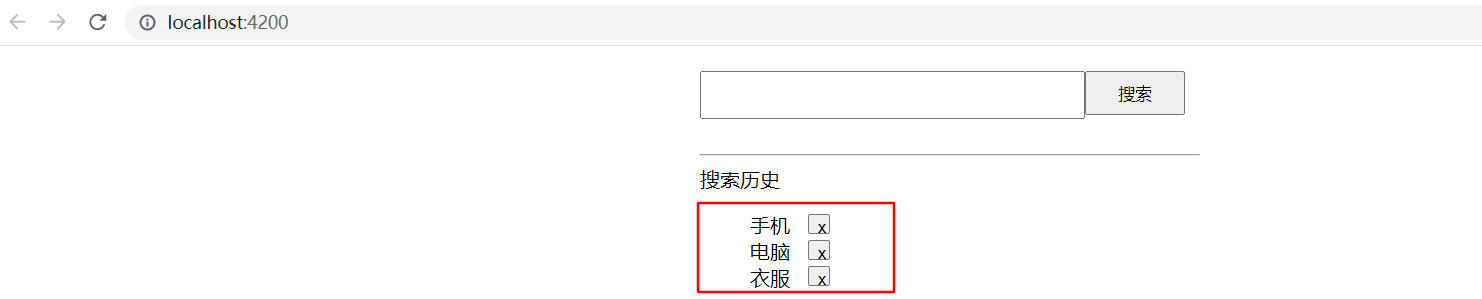
</div>

效果如下所示



删除历史记录

需求：每条记录后面有删除按钮，点击可以删除一条记录。



ts

  deleteHistory(key:any){

    // alert(key);

    this.historyList.splice(key,1);

  }

html

<ul>

        <li \*ngFor="let item of historyList;let key=index">

            {{item}} &nbsp;&nbsp;<button class="del-btn" (click)="deleteHistory(key)">x</button>

        </li>

    </ul>

# git代码仓库配置

<https://github.com/Angeliau/angular-demo.git>

