

7.3

1.进行了vue的安装及环境配置，了解其开发工具

参考教程

[\(16条消息\) Vue安装及环境配置、开发工具vue 开发工具袁小萌同学的博客-CSDN博客](#)

一、node.js安装和配置

二、安装vue及脚手架

三、安装vue-cli 3.x

在安装vue时遇到如下的问题：

npm ERR! code EPERM

npm ERR! syscall mkdir

npm ERR! path D:\Node\node_cache_cacache

npm ERR! errno -4048

原因及解决方法：

修改了npm的全局安装路径

增加node安装目录下的node_global和node_cache文件夹的权限

完成情况：

可以成功运行mytest3

其他功能都在学习中

2.学习组长下发的视频（第一集到第五集，vue简介模块）

HTML（熟悉） CSS（熟悉） JavaScript（用过不熟悉） Ajax（用过不熟悉）

开发工具：vs code（插件live server，代码保存网页自动刷新）

vue基础 vue本地应用 网络应用 综合应用

vue简介：vue是一个开源的JavaScript框架，用于构建用户界面。它专注于实现视图层的响应式和组件化，使开发者能够更高效地构建交互式的单页面应用（SPA）和动态网页。

1.js框架 2.简化dom操作 3.响应式数据驱动

vue基础-el挂载点

vue基础-data数据对象

学习情况汇总：

小组成员成功运行第一个vue程序

运行了群里同学发的例子

7.4（第七集到第十七集）

Vue指令（TIPS:因为md会自动把html语句转为内容 所以这里的尖括号<都省略了）

1)v-text：数据绑定

通过id绑定让div中的数据显示为Vue这个数据结构中的特定值

在vue中这样定义：

```
var app = new Vue({  
  el:"#app", //这里和div的id相对应  
  data:{  
    message:"测试测试",  
  }  
})
```

在div中有两种写法：

1. `h2 v-text="message"` 这样整个h2都是message中的内容 也可以用`v-text="message+'1' "` 这里1会作为代码的字符串拼接放到message的内容之后
2. 在标签中可以添加`{{message}}+'数据库'` 这样用于部分的替换

2) v-html：解释为html的文本绑定

与v-text区别：这个标签中的内容被添加为html语句，而与普通的text文本不同

在刚才的例子中，如果将message的内容替换为一个html语句，就会被当成html来渲染

例如把message转为：

```
a href='www.baidu.com' >百度链接 </a
```

就会在html页面中渲染一个连接 名字为百度链接

3) v-on : 事件绑定

在标签中加上v-on:"事件名"="方法名" 或者 @"事件名"=""

然后在vue数据结构中的methods中添加对应的函数即可，例如：

```
var app = new Vue({
```

```
  el:"#app", //这里和div的id相对应
```

```
  methods:{
```

```
    doIT:function(){
```

```
      alert('弹窗出现')
```

```
    }
```

```
  }
```

```
})
```

然后在对应的标签中添加（可以是button，也可以是h2等等）

@click="doIT" 就可以实现一个弹窗 // 这个是针对于单击 如果要双击可以用@dblclick="" 也可以去找其他的事件名

在vue中method从data中取数据：

this.变量名

函数中的具体和js一样

4) 例子：计数器

利用@click和v-text来实现即可

5) v-show : 显示绑定，根据表达式的真假切换元素的显示状态

```
img src='addr' v-show="isShow">
```

这个isShow可以写定义在vue中的变量，也可以是与vue有关变量的布尔表达式（最终会被解析为bool值）

本质上是用vue来调整div中的display属性值

6) v-if: 用法和v-show一样，但本质不同。v-if通过直接移动dom内容（注释）让其不显示，而不是向v-show一样操纵display属性来实现。

移动dom的开销会更大，所以频繁使用的属性会用v-show。但还要具体问题具体分析

7) v-bind: 专门操纵元素的某个属性（比如写在src title class中的属性）

写法: `v-bind:src="addr"`（引号内容可以设置为表达式，例如字符拼接）

类名是否生效：（类名为active或者空）

1. `img v-bind:class="isActive?'active':''"` (类似于C++的三元表达)
2. `img v-bind:class="{active:isActive}"` (对象方式)

简写:

: src（冒号后直接跟属性名）

8) 案例：图片切换

在v-bind中可以写数组取值`pictures[index]`一类的表达式

9) v-for: 根据数据（数组、字符串、迭代器等）生成列表结构（常用ul和li，当然也可以用在h等等标签中）

根据数据的长度，把模板中的所有东西都拷贝一份并显示

```
<div id="app">
  <ul>
    <li v-for="(item,index) in arr" :title="item">
      {{ index }}{{ item }}
    </li>
    <li v-for="(item,index) in objArr">
      {{ item.name }}
    </li>
  </ul>
</div>
```

```
var app = new Vue({
  el: "#app",
  data: {
    arr: [1, 2, 3, 4, 5],
    objArr: [
      { name: "jack" },
      { name: "rose" }
    ]
  }
})
```

基本写法 (item,index) in 数据名

这个写法类似python的enumerate函数，这里的item是根据数据类型来的，使用方式要遵循item本身的数据类型；index固定为对应的索引

渲染的结果会随着数据内容的变化而变化

10)v-on的补充

1.传递自定义参数：（传递函数形参）

2.除了click和dbclick之外，还有其他的事件，例如@keyup.enter 代表与按下回车绑定

11) v-model:获取和设置表单元素的值（或者叫 双向数据绑定）

```
<div id="app">
  <input type="text" v-model="message" />
</div>
```

```
var app = new Vue({
  el: "#app",
  data: {
    message: "黑马程序员"
  }
})
```

message的值会与vue的值相互连通 两边都会影响另一边的值

7.5

7.6

7.7
