程涛 taonetwork2018

# vue 移动端UI组件库

1:阶段课程介绍

vue 移动端UI组件库/项目 6day

html5新特性(视频/绘图) 7day

微信公众平台 小程序(项目)+公众号 6day

混编(phonegap) 1day

## 1.2: vue ui 组件

vue 当前很火前端框架vue 针对PC用户

pc 端与移动端区别

(1)屏幕宽度:992px >

(2)操作方式:鼠标 事件 手指:触碰操作

-饿了么:基于VUE框架开发移动端组件库 Mint-UI

-MUI :开源团队 MUI

http://dev.dcloud.net.cn/mui/

-滴滴:

## 1.3: 组件mint-ui安装与使用

方式一:学习环境下载安装

下载mint-ui js/css/font

方式二:生产环境下载安装(脚手架)

下载 npm i mint-ui #电脑联网

配置 复杂配置

## 1.4: 组件mint-ui组件分类

-css Components 组件

-js Components 组件

-Form Components 组件

## 1.5: 组件mint-ui组件 Toast 提示框

-下载 mint-ui js mint css

-下载 vue.js

#html加载文件顺序先加载vue.js 再加mint.js

this.$toast("提示消息");

this.$toast({})

|  |  |
| --- | --- |
| message | 提示性文字 |
| postion | top;middle;bottom |
| duration | 持续时间(毫秒) -1 不消失 |
| className | 添加样式名称 |
| close() | (关闭) 删除提示消息对象 |

## 1.6: 组件mint-ui组件 lazyLoad

<ul>

<li>

<img v-lazy="x.png" />

<li>

...

</ul>

#懒加载图片显示样式

img[lazy=loading]{

width:150px;

height:40px;

margin:auto;

}

## 1.7: 组件mint-ui组件 轮播图

<mt-swipe :auto="4000" :show-indicators="true">

<mt-swipe-item>

<img src="x.jpg" />

</mt-swipe-item>

</mt-swipe>

#注意:mint-ui 组件轮播图默认高度0

# 2:vue cli 学子商城(vue ui版本)

## 2.1:完成学子商城

(后端程序):node.js 服务器 mysql数据库

服务器程序

node.js 非常高效 跑车

apache 稳定 家用轿车

nginx 高效 高档家轿车

mysql 关系型开源免费数据库 (互联网产品)

redis 非关系型数据库 (并发量高)

(前端程序):vue-cli 脚手架

## 2.2:vue cli 脚手架

项目为什么用脚手架

(1)前端项目太复杂

(2)脚手架管理维护项目第三方库

(3)热部署

-src: 你创建程序目录

main.js 入口文件

作用:负责加载此次项目使用模块

负责配置模块

负责创建Vue 实例对象

router.js 路由模块

作用:引入不同组件并且配置组件

访问路径

App.vue 根组件

components 项目中开发组件

lib/mui/css mui库使用样式

/fonts 字体图标

/js mui库js

-node\_modules 第三方模块

axios vue 发送ajax请求第三方模块

mint-ui mint-ui 组件库

#安装第三方模板npm i 库名称;

-package.json 项目描述文件和配置文件

配置启动脚手架命令 server

配置启动脚手架端口号 8080

-启动项目 npm run serve

#常规开发组件简单工作流程

(1)需要哪些第三方组件在 main.js 加载配置

(2)在components目录下创建你的组件

(3)在router.js 配置访问路径

## 2.3:vue cli 脚手架-创建测试组件

-main.js 加载需要第三方模块

-在components 创建自定义组件 Exam01.vue

<template></template> 显示数据与格式

<script></script> 完成业务功能

<style></style> 外观样式

-在router.js 配置如果访问组件(访问路径)

引入组件

import 组件对象名称 from "./components/Exam01.vue"

配置组件访问路径

{path:"/访问路径",component:组件对象名称}

#注意:访问路径加 /

-访问组件

http://127.0.0.1:8080/#组件访问路径 回车

练习:创建组件 List.vue

-显示data中定义用户列表信息

-编号;名称;地址 三个对象保存数组

常见错误:

(1)创建组件模板

- Failed to compile 编译出错 <template>标签出错

<div>...<div>

-创建组件没有显示 组件访问路径缺少 /

-脚手架启动失败

npm ERR! code ENOENT

npm ERR! errno -4058

npm ERR! syscall open

原因:启动脚手架位置不正确

原因:脚手架内容损坏

vue\_app\_00/npm run serve

(1)package.json

- router.js 配置路径出错

\* ./components/List01.vue in ./src/router.js

## 2.4:Home



## 2.5:Home-创建空组件

(1)创建空组件

src/components/tabbar/Home.vue

(2)配置访问路径 /Home

(3)测试 http://127.0.0.1:8080/#/Home

-顶部导航栏--mint-ui组件

## 2.6:Home-顶部导航条

mint-ui库创建顶部导航条

(1)下载mint-ui库到当前脚手架

npm i mint-ui

#如果下载成功 node\_modules/mint-ui

(2)将mint-ui Header 组件引入脚手架 main.js

import {Header} from "mint-ui"

(3)将mint-ui Header 将Header 组件Vue实例对象main.js

Vue.component(组件标签名称,组件对象);

示例:

Vue.component(Header.name,Header);

(4)在指定组件Home.vue 使用Header

<mt-header title="学子商城"></mt-header>

## 2.7:Home-轮播

(1)下载不用

(2)引入组件

import {Header,Swipe,SwipeItem} from "mint-ui"

(3)注册组件

Vue.component(Swipe.name,Swipe);

Vue.component(SwipeItem.name,SwipeItem);

(4)在Home组件使用轮播

<mt-swipe>

<mt-swipe-item>

<img src="" />

</mt-swipe-item>

</mt-swipe>

常见错误

1:图片不能显示

-图片路径不正确 图片都需要保存服务器端

-脚手架运行 public/index.html app.js[组件..]

-轮播样式添加错误

## 2.8:第三方组件库 mui

mui 库示例mui-master\examples\hello-mui\examples

九宫格 grid-default.html

作业1：完成Home 九宫格

今天学习的内容

1:复习昨天的内容

远程同学微信没电稍等一会再问问题

2:作业 九宫格

九宫格:在mint-ui组件库基于vue框架

mui不是基于vue框架 只是css/js文件

(1)官方网站下载安装包

(2)copy css js fonts[字体图标] src/lib/mui/...

(3)使用css文件 main.js 直接 import 引入任何组件

即可以使用

(4)src/img/...

# 3:今天的目标

## 3.1:Home----九宫格图片大小-(重点)

前端设计(PS) 项目所有外观结果ps效果图

-物理像素:设计效果图片中像素 120px

-逻辑像素:程序中使用像素 60px

#智能手机:iphone4 技术参数标准 ..

#大公司:前端设计ps图片

**#iphone6 逻辑像素物理像素 2x dpr**

#iphone6 plus 3x

常见错误:

(1)启动脚手架 8081

## 3.2:Home----tabbar-(底部导航条)

(1)mui 组件库中有 tabbar.html

## 3.3:Home----图片轮播

(1)图片保存服务器端 文件夹 public/img/...

图片保存路径放在 数据库中

http://127.0.0.1:3000/img/banner1.png

为什么?图片多;安全

(2)创建服务器

//服务器项目目录

vue\_server\_00

public/img/轮播图图片

pool.js 连接池

db.sql 数据库创建表脚手架

app.js 主程序

(3)通过脚手架使用ajax获取服务器端图片列表

axios 第三方组件与vue 关系

-下载 npm i axios

-main.js 引入axios 库

import axios from "axios"

-main.js 配置axios 库跨域访问保存session值

axios.defaults.withCredentials=true

-main.js 将axios 库配置Vue

Vue.prototype.axios = axios

-Home.vue

this.axios.get(url).then(result=>{

})

## 3.4:跨域访问

三种不同跨域访问方式

-访问域名不同 a.com/1.js -> b.com/2.js

-访问端口不同 a.com:8080/1.js ->a.com:3000/2.js

-访问协议不http://127.0.0.1/1.js https://127.0.0.1/2.js

## 解决跨域访问

-jsop 填充式json

-在服务器端程序配置允许那个域名下程序访问

nodejs 下载模块 cors

npm i cors

-在主程序 app.js

**const cors = require("cors"); #引入模块**

**app.use(cors({ #允许那个域名程序跨域访问**

**origin:["http://127.0.0.1:8080","http://localhost:8080"]**

**credentials:true**

**}))**

app.get("/list"...)

#注意:配置代码在所有请求之前添加

#常见错误:轮播图片错误

# 4:新闻列表

(1)分析新闻列表保存数据库 表 名 列 类型

vue\_server\_00/db.sql

(2)app.js 分页请求

(3)脚手架创建组件

- /home/NewsList.vue /NewsList.vue

-发送ajax请求获取新闻列表第一页数据

-使mui组件创建新闻列表

常见错误:

- net::ERR\_CONNECTION\_RESET

原因:服务器 node.js 出错并且停止工作

查看node.js 控制台出错消息

Table 'xz.xz\_news' doesn't exist 新闻表不存在

作业1：创建过滤器过滤器日期格式

2019年1月29日 17:43:00

作业2: 完成新闻列表

-添加button按钮 加载更多...

1:复习昨天的重点

2:复习node.js 需要注意知识点--(重点)

2.1:参数传递获取造型

客户端脚手架(发) (参数传递) node.js(收)

-发ajax this.axios.get("http://127.0.0.1:3000/list?id=9&age=19")

-收 node.js req(request 请求对象)

var id = parseInt(req.query.id);

var age = parseInt(req.query.age);

错误:node.js NaN 参数错误

2.2:完成一个功能使用哪条SQL node.js

(1)登录:

-参数:用户名密码 uname upwd

-sql:SELECT uname,upwd FROM xz\_login

WHERE uname = ? AND upwd = md5(?)

用户名或密码错 result.length == 0

(2)商品列表(分页):

-参数:pno 页码 pageSize 页大小

-sql SELECT id,img\_url,price,title FROM xz\_product

LIMIT ?,?

? (pno-1)\*pageSize

? pageSize

-sql SELECT count(id) FROM xz\_product;

Math.ceil(20/pageSize); 向上取整

(3)购物车:

(0)列表 SELECT id,name,price,title,count FROM xz\_cart;

(1)添加

参数:pid 商品编号 price 商品价格 count 数量

-查询当前商品是否在购物车中

SELECT id FROM xz\_cart WHERE pid = ?

-如果商品己经存在 更新数量

UPDATE xz\_cart SET count = ? WHERE id = ?

-如果商品不存在 添加

INSERT INTO xz\_cart VALUES(null,pid,price,count,now())

(2)计算合计 脚手架计算属性

(3)修改(购物车中商品数量)

参数:id购物车项目id count 数量

sql:UPDATE xz\_cart SET count = ? WHERE id = ?

(4)删除[多选]

参数:uid 谁的购物车 id=1,2,3

字符串拼接 1,2,3

sql:DELETE FROM xz\_cart WHERE id IN (1,2,3)

(5)创建订单:

-参数:uid 购物车用户id

-查询购物车中数据 SELECT id,uid,price FROM xz\_cart;

-添加订单表 INSERT INTO xz\_order VALUES(?,?....)

2.3:node.js 非阻塞[高效 无序] 如果有功能需要严格顺序

sql1

pool.query(sql1,(err,result)=>{

//sql1 一定执行结束 结果

sql2

pool.query(sql2,)

});

## 4.1:修改脚手启动时显示Home 组件

默认脚手路径 http://127.0.0.1:8080/#/

{path:'/',component:HelloContainer},

修改:当脚手架访问 / 让浏览器自动将地址 /修改/Home

功能: 跳转

{path:'/',redirect:'/Home'}

{path:'/Home',component:Home},

## 4.2:修改App.vue 组件

4.1:创建按钮 "加载更多..." mint-ui

-import {Button...} from "mint-ui"

-Vue.component(Button.name,Button);

-<mt-button type="primary" size="large">加载更多...</mt-button>

4.2:创建过滤器 格式化日期对象 创建全局过滤器

-创建过滤器main.js

Vue.filter(过滤器名称,function(val){

-依据参数创建日期对象

-年 月 日 小时 分 钟 秒

-返回字符串 Y-M-D h:mi:s

});

-调用过滤器组件模板

{{item.ctime | 过滤器名称}}

4.3: 点击按钮 "加载更多..." 查看下一页

-为加载更多按钮绑定点击事件

-创建函数加载更多 {下一页}

-遇到问题:显示下一页内容上页内容覆盖

希望保存上一页内容

this.list = [];

this.list = result.data.data;

var rows = this.list.concat(result.data.data);

this.list = rows;

# 5:学子商城--商品列表

## 5.1:分析查询表

-查询当前页内容

xz\_laptop 商品表

xz\_laptop\_pic 商品图片表

商品编号 商品标题 商品价格 商品图片

lid title price md

(1)查询表保存在 FROM后

(2)为每张表创建别名

(3)查找两张表哪列有关联的 = <= >=

SELECT l.lid,l.title,l.price,p.md

FROM xz\_laptop l,xz\_laptop\_pic p

WHERE l.lid = p. laptop\_id

GROUP BY l.lid

LIMIT ?,?;

app.js

## 5.2:脚手架

-创建空组件 src/components/goods/GoodList.vue

-配置访问路径 /GoodsList

-测试

-设置布局

调错/添加按钮/添加点击事件

常见错误

(1) net::ERR\_CONNECTION\_RESET node.js程序出错停止工作

解决:查看node.js 控制台出错信息

(2)error in your SQL syntax

原因:sql语法写错了

作业1：商品详情组件

今天学习的内容

1:复习昨天的重点

# 6:今天的目标

## 6.1:脚手架--组件之间跳转与参数

组件之间跳转主要(组件1->跳->组件2)

**(1)标签方式跳转(优先选)**

<router-link to="组件路径">

.....

<router-link>

示例:Home.vue -> NewsList.vue

<router-link to="/NewsList">

</router-link>

**(2)编程序方式跳转**

this.$router.push("组件路径");

示例:

<img src="1.jpg" @click="jump"/>

jump(){

this.$router.push("/NewsList");

}

示例:

Exam01.vue -> NewsList.vue

<h1 @click="jump">跳转NewsList.vue 组件</h1>

jump(){

this.$router.push("/NewsList")

}

**(3)组件之间跳转需要传递参数**

Exam01.vue ->NewsList.vue

?nid=6

-传递参数 Exam01.vue

this.$router.push("/NewsList?nid=6")

-接收参数 NewsList.vue

var id = this.$route.query.nid;

#通常情况接收参数 created()

#$router 属性负责组件之间跳转

#$route 属性负责接收参数

## 6.2:新闻列表/新闻详细

-创建空组件 src/components/home/NewsInfo.vue

-添加组件访问路径 /NewsInfo

-检查 NewsList.vue 新闻列表

标签跳转

<router-link :to="'/NewsInfo?nid='+item.id">

</router-link>

编程方式 ?nid=变量

-创建ajax 请求获取新闻详细信息

-node.js /findNewsInfo

参数:id

验证参数是否合法: /^\d{1,}$/

sql: SELECT id,title,content,ctime,img\_url FROM xz\_news WHERE id=?

json:{code:1,data:[]}

-保存当前 data:{info:{}}

## 6.3:脚手架--新闻详细

-添加 mui库card.html 卡片布局

-添加 新闻 标题 内容 时间

-保存参数 id

-发送ajax请求获取数据并显示

# 7:今天的目标

## 7.1:评论列表--发表评论

用户点击新闻列表某一条新闻，看到新闻详细发表评论

-用户输入评论内容

-发表评论 [将用户评论内容保存数据库 xz\_comment]

## 7.2:评论列表--发表评论-开发评论

-查询数据库表 xz\_comment[id,content,ctime,nid]

-node.js 创建程序接收评论内容添加

(1)用户请求方式 POST app.post("/addcomment")

(2)请求参数 content,nid

2.1:post获取参数 下载安装第三方模块 body-parser

2.2:post获取参数 配置模块

2.3:post 获取参数 req.body.nid req.body.content

(3)SQL INSERT INTO xz\_comment VALUES(null,?,now(),?)

(4)JSON {code:1,msg:"发表成功"}

|  |  |
| --- | --- |
| 用户请求方式 GET | 用户请求方式 POST |
| 发送请求参数：最1KB | 发送请求参数：不限制 |
| 作用:获取服务器数据 | 作用:向服务器发送数据 |
| 场景:分页;详细; | 场景:注册;添加;上传图片 |
| 安全:较差 | 安全:较差 HTTPS安全级别高 |

-脚手架 comment.vue 发表评论

-引入第三方模板 qs

1.1:下载npm i qs 模块

1.2:在main.js 引入qs模块 import qs from 'qs';

1.3:在main.js Vue对象属性 Vue.prototype.qs = qs;

1.4 comment.vue

var postData = this.qs.stringify({

nid:9,

content:"......"

})

this.axios.post("127..", postData).then(result=>{

})

-在comment.vue 添加mint-ui Toast

由Toast只有在comment.vue 等少数组件中使用

(1)在当前组件引入 import {Toast} from "mint-ui"

(2)立即调用 Toast("评论内容不能为空");

-常见错误

(1)Duplicate keys detected: '2'

<div v-for="item in list" :key="item.id"></div>

原因:如果item.id有重复值 2 2

this.list=[] this.pno++

post this.pno = 1; getMore() [{id:1,img.},{id:2}]

## 7.3:商品详细

1-创建空组件 src/components/goods/GoodInfo.vue

2-添加访问路径 /GoodInfo

3-添加 card.html 卡片布局

4-创建 src/components/sub/swiperBox.vue 子组件

5\*-swiperBox.vue 当父组件调用子组件时传递数组

GoodInfo.vue 商品名称; 价格;数量 1

【添加购物车】

\*-获取当前商品编号

GoodList.vue -> GoodInfo.vue 参数pid

(1)为商品列表中每一张图片绑定点击事件

跳转GoodInfo.vue 组件

参数 ?pid=2

#通用参数从模板传递事件处理函数

<img @click="jumpInfo" :data-lid="item.lid"/>

jumpInfo(e){

var pid = e.target.dataset.lid;

this.$router.push("/GoodInfo?pid="+pid)

}

-发送ajax 获取商品名称 价格

node.js

参数:pid

sql: SELECT lname,price FROM xz\_laptop WHERE lid = ?

json {code:1,data:[]}

GoodInfo.vue

-data:{info:{}}

-methods:{findGood} result.data.data[0]

-created()