# March\_Half

2020-09-25

下载mysql、navicat、idea

Mysql：net start mysql（stop）

使用idea创建maven项目March\_Half

Spring：

<https://spring.io/guides/gs/spring-boot/>

<https://www.docs4dev.com/docs/zh/spring-boot/1.5.9.RELEASE/reference/getting-started-first-application.html#getting-started-first-application-code>

引入包

Mybatis：

<https://mybatis.org/mybatis-3/zh/index.html>

2020-09-26

找寻b站视频

创建march\_half\_wx

2020-09-27

调整微信小程序绑定页面

2020-9-28

调整为小程序绑定页面css

下载vue node npm

项目移入github

2020-09-29

创建march\_half\_vue

构建vue项目

2020-09-30

构建vue3.0项目（https://www.cnblogs.com/yangsg/p/13732415.html）

Vue3.0 ：

// 在vue2中 data 在vue3中使用 reactive代替

import { reactive, computed } from 'vue'

import { useRouter } from 'vue-router'

// setup相当于vue2.0的 beforeCreate和 created，是vue3新增的一个属性，所有的操作都在此属性中完成

// 通过reactive 可以初始化一个可响应的数据，与Vue2.0中的Vue.observer很相似

// 使用计算属性生成

// 在Vue3.0中，所有的数据和方法都通过在setup 中 return 出去

//provide inject 父子

* ~~beforeCreate~~ -> use setup()
* ~~created~~ -> use setup()
* beforeMount -> onBeforeMount
* mounted -> onMounted
* beforeUpdate -> onBeforeUpdate
* updated -> onUpdated
* beforeUnmount -> onBeforeUnmount
* unmounted -> onUnmounted
* errorCaptured -> onErrorCaptured
* renderTracked -> onRenderTracked
* renderTriggered -> onRenderTriggered

小小的todolist vue3 demo

继续mybatis

<https://blog.csdn.net/iku5200/article/details/82856621>



创建interface跟对应的映射xml：

右击类名copy referen --> xml文件namespace

方法名getUserByLoginName copy 🡪 xml文件id

初始化 开始连接微信小程序

成功实现登录

2020-10-01

重新创建idea项目

希望能微信小程序绑定系统账号

<https://my.oschina.net/u/3337958/blog/1618224>

<https://my.oschina.net/u/3337958/blog/1618405>

授权登录 获取每个微信的独特openid

2020-10-03

创建vue2.0项目制作后端  
 尝试在2.0导入npm install @vue/composition-api --save

2020-10-04

绑定信息页面 想实现增删改查

个人页面的信息修改

整体的修改了关于用户的数据表

根据这个实现了增加跟全体遍历

2020-10-05

实现登陆账号根据openid增删改查

开始准备实现home首页的内容

并开始后台页面

Api（axios）、拦截器（request.js）---封装axios

<http://www.axios-js.com/zh-cn/docs/#%E6%8B%A6%E6%88%AA%E5%99%A8>

2020-10-06

Vue3实现axios跨域

但是没有服务器还是尝试自己本地的数据库

Vue与spring之间的跨域一直不成功

2020-10-07

经过检查发现是vue的数值传递出现了问题，在get的情况下不应该使用date而是params。

了解了一下axios的get跟post 发现可以实现get在url里传递数值，还没有实现post的通过request body传递参数。

截至现在，基本实现微信小程序、vue、spring之间的增删改查，还有复杂的数据操作还没有开始。

接下来就是开始设计程序要实现哪些功能，做一个整体的规划。

希望能在时间跟精力至于可以实现更多方面。

2020-10-08

2020-10-09

改变了自己vue后台的背景图片

思考自己这个小程序大概需要实现那几个功能，开始设计自己的个人页面还有首页，初步考虑想用多彩的卡片式

（微信小程序实现卡片层叠滑动效果）

<https://www.jb51.net/article/163592.htm>

（css实现梯形（各种形状）的网页布局——transform的妙用）

<https://www.cnblogs.com/pengfei-nie/p/6919687.html>

删去最开始给个人页面设计的半圆形，改成梯形

改变自己的# 颜色配色，打算使用白色为主色调，采用多彩的卡片

2020-10-10

开始翻看论文，寻找文献参考

2020-10-11

敲完任务书的概况

实现个人页面的改版和绑定信息的页面，还有新建首页

使用echart制作图表

Rgb-16转换<https://www.sioe.cn/yingyong/yanse-rgb-16/>

在线取色器<http://www.jiniannet.com/Page/allcolor>

2020-10-12

继续后台的首页搭建





开启后台页面的首页，敲定颜色

构思结构，也大概确定下数据库的建表

2020-10-14

重建后台首页，并创建好相对应的导航栏

研究了sessionstorage和localStorage：

localstorage是本地存贮 ，清除本地存贮才可以。sessionstorage 是关闭浏览器/页面 数据就会清除。

（cookie 一般设定是20min 但是你也可以设置 记住： 它只在设定时间前有效）

localStorage是一种比cookie大的存贮空间。 大约5m

四种方法 ：1 localStorage.setItem(); 传

2. localStorage.getItem(); 接

3 localStorage.removeItem() ; 删

4 localStorage.setItem(); 改

sessionStorage

sessionStorage.key(int index) //返回当前 sessionStorage 对象的第index序号的key名称。若没有返回null。

sessionStorage.getItem(string key) //返回键名(key)对应的值(value)。若没有返回null。

sessionStorage.setItem(string key, string value) //该方法接受一个键名(key)和值(value)作为参数，将键值对添加到存储中；如果键名存在，则更新其对应的值。

sessionStorage.removeItem(string key) //将指定的键名(key)从 sessionStorage 对象中移除。

sessionStorage.clear() //清除 sessionStorage 对象所有的项。

Json：// 存储值：将对象转换为Json字符串

sessionStorage.setItem('user', JSON.stringify(userEntity));

// 取值时：把获取到的Json字符串转换回对象

var userJsonStr = sessionStorage.getItem('user');

userEntity = JSON.parse(userJsonStr);

2020-10-15

微调整后台页面的main框架

创建数据库types表，并在spring中创建好相对应的xml、dao、type、typeControll文件，实现类别的增删改查。而在后台页面中，加载出来所有类别。

（在此之前，基本完成两个平台的登录注册问题）

2020-10-16

傍晚解决昨天遗留下来的打开dialog会先灰屏的问题，:append-to-body="true"

2020-10-17

2020-10-18

开始在后台页面加上type的增删改查功能，并同样的样式完成了管理员的增删改查

2020-10-19

回忆之前做的nodejs项目

2020-10-20

边回忆边继续开始后台页面的图片上传功能

2020-10-21

执行任务

2020-10-22

继续努力实现图片上传

Tinymce组件  
 <https://www.cnblogs.com/qinshengfei/p/12307423.html>

<https://blog.csdn.net/weixin_40337982/article/details/84031778>

<https://www.cnblogs.com/aknife/p/11840347.html>

<https://blog.csdn.net/qq_42852004/article/details/90055999>

<https://www.cnblogs.com/wisewrong/p/8985471.html>

<https://blog.csdn.net/weixin_43809369/article/details/90212044>

2020-10-23

<https://liubing.me/vue-tinymce-5.html>

根据教程展示出了tinymce富文本，使用2的语法写了一个简单组件，可以随时调用。

图片上传是个极大的难点，特别是如何前端获取图片之后，如何传给后端。后端又如何将图片返回到前端预览。

<https://www.52pojie.cn/thread-822874-1-1.html>

<https://blog.csdn.net/qq_41618510/article/details/95199806>

<https://blog.csdn.net/zhizungege/article/details/106839214>