

# trello项目实战

## 课堂目标

- 掌握vue全家桶基本使用
- 掌握nodejs创建后端接口
- 掌握mysql的基本操作
- 了解cors解决跨域问题

## 知识要点

- vue项目创建
- vue基本使用、vuex状态管理、vue-router路由管理
- nodejs项目创建
- cors跨域解决
- mysql数据库操作
- koa2框架使用

## trello项目功能介绍

- 注册登录功能
- 列表添加功能
- 任务添加功能
- 任务详细内容功能

## 用到的技术点

- 一、前端vue全家桶
- 二、前后端交互axios
- 三、后端nodejs及koa框架
- 四、mysql数据库
- 五、跨域问题cors
- 六、jwt鉴权

## 环境搭建

- 一、安装nodejs环境
- 二、安装mysql数据库
  - mac下安装 地址：<https://dev.mysql.com/downloads/mysql/>下载
  - 设置密码

- `cd /usr/local/mysql/bin/`
- `./mysql -u root -p`
- `set password for 用户名@localhost = password('新密码');`
- 通过navicate连接
- windows下安装mysql
- 下载地址<https://dev.mysql.com/downloads/mysql/>

### 三、安装vue-cli;

```
npm install @vue/cli
```

```
yarn global add @vue/cli
```

检查版本是否正确 `vue --version`

- 延伸知识点
  - npm 修改源地址
    - `npm config set registry https://registry.npmjs.org/`
    - `npm config set registry https://registry.npm.taobao.org/`
  - 查看下载源地址
    - `npm config list`
  - yarn 安装
    - `npm install -g yarn`

## 创建项目

### 一、创建vue项目

- `vue create trello` 创建项目
- vue-router介绍

### 二、创建服务端项目

### 三、解决跨域问题实现网络请求

## 实现注册登录

- 一、创建数据库
- 二、实现注册功能
- 三、实现登录功能

## 常用sql语句

### 一、增

```
INSERT INTO 表名 (字段一, 字段二, 字段三) VALUES ("值一", "值二", "值三");
```

## 二、删

```
DELETE FROM 表名 WHERE 条件;
```

## 三、改

```
UPDATE 表名 SET 设置的内容 WHERE 条件语句;
```

## 四、查

```
SELECT 字段 FROM 表名 WHERE 条件语句;
```

## 五、条件语句;

1. AND 2. OR 3. ORDER BY (DESC/ASC) 4. LIMIT 5. LIKE 6. JOIN ON 7. AS

# 登录 jwt验证



一、jwt: json web token是为了在网络应用环境间传递声明而执行的一种基于JSON的开放标准

## 二、生成token

- jsonwebtoken 模块

```
const token = jwt.sign({
  name: reslut[0].username,
  _id: reslut[0].id
}, 'my_token', { expiresIn: '2h' });
```

## 三、缓存token: cookie 或者是 locaStroage

## 四、token的组成

- 头部的基本信息;

```
{
  "typ": "JWT",
  "alg": "HS256"
}
```

- payload :存放自定义信息 ; 预定义信息有如下几个:

iss: 该JWT的签发者  
sub: 该JWT所面向的用户  
aud: 接收该JWT的一方  
exp(expires): 什么时候过期, 这里是一个Unix时间戳  
iat(issued at): 在什么时候签发的

- signature 签名 哈希需要有secret;

## 五、前端的校验;

- axios的拦截器

```
Vue.prototype.$http.interceptors.request.use(function(config){
  let token = localStorage.getItem("trello_token");
  // console.log("???",token);
  if(token){
    config.headers.withCredentials = true;
    config.headers.common['Authorization'] = 'Bearer ' + token;
  }
  return config;
})
```

- 跨域相关设置

- 后端允许设置头部

```
ctx.set('Access-Control-Allow-Credentials', true); //1.服务端允许带凭证; 上面允许的域名不能是所有; 2.cookie设置也要通过ajax;
ctx.set("Access-Control-Allow-Headers", "content-type,authorization,withcredentials");
```

## 六、校验token

- koa-jwt模块

- koajwt({secret:'my\_token'})

## trello主页

- 设置主页路由
- MyHead组件 和 subTitle组件
- 引入rest.css
- 引入字体图标的处理

## 主页动态数据呈现

### 添加sql表

- list表的相关字段
  - id、listtitle、sortid、uid
- task任务表的设计
  - id、taskid、taskdescribe、sortid、lid

根据list及task表查询对应的数据

前端数据呈现

- axios发送数据
- vuex管理请求的数据