引入Vuex

# 说明

## nodejs: v17.9.0

## npm: 8.5.5

## VUE: ^3.2.13

## Vue CLI: @vue/cli 5.0.4

## Vue Router：^4.0.15

## axios:^0.27.2

## Vuex: ^4.0.2

# 参考资料

## Vuex源码

<https://github.com/vuejs/vuex>

## Vuex官网

<https://vuex.vuejs.org/>

pinia

## https://www.zhoujie16.cn/article/10430

# 引入Vuex

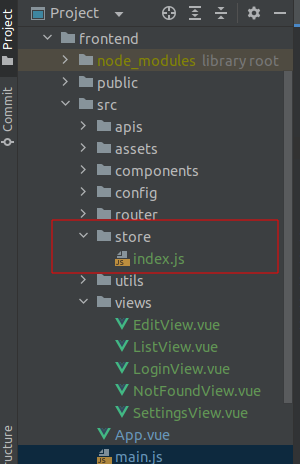
## 安装

npm install vuex@next --save

参考：<https://vuex.vuejs.org/installation.html#npm>

## 创建index.js文件

虽然官方文档给的参考示例是在初始化app那里引入Vuex的(参考：https://vuex.vuejs.org/guide/#the-simplest-store)，但是当项目变大之后，应该使将Vuex相关的内容剥离出来。所以我们在src目录下新建store目录，然后在store目录下创建index.js文件用来撰写与Vuex相关的内容。



## 将store实例作为plugin安装到vue实例上

// main.js

import { createApp } from "vue";

import "normalize.css";

import "@/assets/css/base.css";

import App from "@/App.vue";

import router from "@/router";

import store from "@/store"

// app.use(）只能传递一个参数，所以需要多写几次app.use()

const app = createApp(App);

app.use(router)

app.use(store)

app.mount("#app");

## state

作用：定义数据。

1. 获取存储到vuex里面的值xxx

在组件中访问：store.state.xxx

在JS中访问：this.$store.state.xxx

参考：https://vuex.vuejs.org/guide/#the-simplest-store

## mutations

作用：修改数据(state)。

## actions

因为将异步调用和mutations结合使用让程序很难推断（理解），所以有了action，mutations用于同步操作，actions用于异步操作。

# 项目结构图

//todo