axios

2019年11月25日

13:47

1. axios的两种请求方式

get:

vue文件中

      let param = {

        id: id

      };

      getTicket(param).then(res => {

        this.img = res.data;

      });

api.js中

export function getHandleData(params) {

  return axios.get(

    URL + "/getMessage/getHandleData?date=" + new Date().getTime(),

    { params: params }

  );

}

post:

vue文件中

import { getMenuData } from "../../../plugins/api";

      var params = this.$qs.stringify(params);

       getMenuData(params).then(res => {

        this.list = res.data;

      });

api.js中

// 查询接口数据(刷新)

export function getInterfaceDBData(params) {

  return axios.post(

    URL + "/getMessage/getInterfaceDBData?date=" + new Date().getTime(),

    params

  );

}

1. 如何使用axios上传图片

        <Upload

          ref="upload"

          :show-upload-list="false"

          :format="['jpg', 'jpeg', 'png']"

          :max-size="208"

          :before-upload="handleBeforeUpload"

          :on-format-error="handleFormatError"

          :on-exceeded-size="handleMaxSize"

          type="drag"

//action表示上传地址,此处为无效地址,之后通过axios上传

          action="//jsonplaceholder.typicode.com/posts/"

          style="display: inline-block;width:58px;"

        >

          <div style="width: 58px;height:58px;line-height: 58px;">上传图片</div>

        </Upload>

        <Button type="primary" @click="submit(formData)">提交图片</Button>

//上传前,获取图片的base64加密内容

    handleBeforeUpload: function(file) {

      let reader = new FileReader();

      reader.readAsDataURL(file);

      const \_this = this;

      reader.onloadend = function() {

        file.url = reader.result;

        \_this.formData = file;

      };

    },

   //提交车票

    submit: function(file) {

      if(file == ""){

        return;

      }//未选择图片

      if (this.id != "") {

        this.delete(this.id);

      }//上传前,根据id删除图片

      let param = this.$qs.stringify({

        uploadImg: file.url

      });

      addTicket(param).then(res => {

        this.$message(res.data.message);

        this.getTicket(res.data.name);

        this.id = res.data.name;

        window.parent.postMessage({

        cmd: "del"

      })

      });

    },

    //获取车票

    getTicket: function(id) {

      let param = {

        id: id

      };

      getTicket(param).then(res => {

        this.img = res.data;

      });

    },

  //删除车票

    delete: function() {

      let params = {

        id: this.id

      };

      delTicket(params).then(res => {

        this.img = "";

        this.id = "";

      });

    }

其他办法:

1. **阿里云**
2. **传统文件服务器上传**

    upload: function() {

      var f = this.$ref.photo;

      let param = new FormData();

      param.append("img", f.files[0]);

//通过append向form对象添加数据

      let config = {

        headers: { "Content-Type": "multipart/form-data" }

      }; //添加请求头

      axios

        .post(

          this.$store.state.https +

            this.$store.state.geturl.upload +

            "?token=" +

            this.$store.state.tokenstatic,

          param,

          config

        )

        .then(response => {

          this.form.url =

            this.$store.state.geturl.photourl + response.data.data.url;

          console.log(this.form.url);

          if (this.form.url) {

            alert("图片上传成功");

          }

        });

    }

**3.axios实现跨域,vue上实现的两种方式:**

1)config文件中设置代理

proxy: {

"/api": {

target: "<http://127.0.0.1:8018>", //后台服务地址

ws: true,

changeOrigin: true,

pathRewrite: {

"^/api": ""

}

}

}

设置请求头

let config = {

async: false,

// baseURL: process.env.VUE.APP.API.URL,

timeout: 600 \* 1000, //Timeour

headers: {

"Content-Type": "application/x-www-form-urlencoded"

// 'Content-Type': 'application/json'

},

withCredentials: false //Check cross-site Access-Control

};

const \_axios = axios.create(config);

//导出新建立的axios接口

export default \_axios;

调用\_axios接口,实现前后台交互,其中,URL = "api",代理将代理访问'http://127.0.0.1:8080

(config中设置的端口)'

export function getOneLevel(params) {

return axios.post(

URL + "/businessProcess/getOneLevel?date=" + new Date().getTime(),

params

);

}

注意 : 控制台获取到的访问是http:localhost:8020/api/….,这并不是代理不成功.

2)需要后台进行跨域设置,设置请求头过程类似,使用\_axios接口时,可以写为

export function getOneLevel(params) {

return axios.post(

"<http://127.0.0.1:8080/businessProcess/getOneLevel?date=>" + new Date().getTime(),

params

);

}

或者在设置请求头后,添加

\_axios.defaults.baseURL = '<http://localhost:8080/>'

使用接口时,可以写为

export function getOneLevel(params) {

return axios.post(

"/businessProcess/getOneLevel?date=" + new Date().getTime(),

params

);

}

**后台解决blocked by CORS:**

response.setHeader("Access-Control-Allow-Origin", "\*");

或者更新springboot 的相关jar包

在类上面添加@CrossOrigin

**qs库对数组的format有以下几种形式：**

qs.stringify({ids: [1, 2, 3]}, { indices: false })

// 形式： ids=1&ids=2&ids=3

qs.stringify({ids: [1, 2, 3]}, {arrayFormat: 'indices'})

// 形式： ids[0]=1&aids1]=2&ids[2]=3

qs.stringify({ids: [1, 2, 3]}, {arrayFormat: 'brackets'})

// 形式：ids[]=1&ids[]=2&ids[]=3

qs.stringify({ids: [1, 2, 3]}, {arrayFormat: 'repeat'})

// 形式： ids=1&ids=2&ids=3

GET和POST请求参数区别:

POST:

export function uploadOrder(params) {

return axios.post('http://127.0.0.1:8080' + '/demo/uploadOrder?date=' + new Date().getTime(),

params)

}

GET:

export function getGoodsList(id) {

return axios.get('http://127.0.0.1:8080' + '/demo/getGoodsList?date=' + new Date().getTime(), {

params: id

})

}

注意: POST的params外没有大括号,例如 return axios.post(url,params) ,否则会导致请求参数类似于

{"params":"string=true&list=12321%7C%E5%90%B9%E9%A3%8E%E6%9C%BA%7C99%7C1&

list=12321%7C%E5%90%B9%E9%A3%8E%E6%9C%BA%7C99%7C1"}

GET请求可以直接使用{params},不需要{params: params}

猜想: POST的参数是通过将实参转换为字符串,进而拼接到请求参数上的,或者直接将字符串实参拼接

到参数上;而GET的参数是通过传递对象,获取params对象的value值,将其转换为字符串后拼接到请求

参数中

问题 :

1. 控制台报500 :检查node服务器是否已经启动,在package.json中添加start和server