

[首页](#)[新闻](#)[博问](#)[专区](#)[闪存](#)[班级](#)[代码改变世界](#)[注册](#)[登录](#)

古兰精

[博客园](#)[首页](#)[新随笔](#)[联系](#)[订阅](#)[管理](#)

随笔 - 1123 文章 - 0 评论 - 161 阅读 - 481万

vue-router路由元信息详解

一、官方文档

路由元信息：定义路由的时候可以配置 `meta` 字段



```
const router = new VueRouter({
  routes: [
    {
      path: '/foo',
      component: Foo,
      children: [
        {
          path: 'bar',
          component: Bar,
          // a meta field
          meta: { requiresAuth: true }
        }
      ]
    }
  ]
})
```

公告

昵称：古兰精

园龄：4年4个月

粉丝：356

关注：1

[+加关注](#)

2021年9月						
日	一	二	三	四	五	六
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

搜索

 找找看

```
}  
]  
}))
```



那么如何访问这个 `meta` 字段呢？

首先，我们称呼 `routes` 配置中的每个路由对象为 **路由记录**。路由记录可以是嵌套的，因此，当一个路由匹配成功后，他可能匹配多个路由记录。

例如，根据上面的路由配置，`/foo/bar` 这个 URL 将会匹配父路由记录以及子路由记录。

一个路由匹配到的所有路由记录会暴露为 `$route` 对象（还有在导航守卫中的路由对象）的 `$route.matched` 数组。因此，我们需要遍历 `$route.matched` 来检查路由记录中的 `meta` 字段。

下面例子展示在全局导航守卫中检查元字段：



```
router.beforeEach((to, from, next) => {  
  if (to.matched.some(record => record.meta.requiresAuth)) {  
    // this route requires auth, check if logged in  
    // if not, redirect to login page.  
    if (!auth.loggedIn()) {  
      next({  
        path: '/login',  
        query: { redirect: to.fullPath }  
      })  
    }  
  } else {  
    next()  
  }  
} else {
```


 谷歌搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

随笔分类

Android/iOS移动端App应用(61)
CI/CD/Jenkins/Docker部署知识(36)
Electron桌面端App应用(2)
ES6(34)
Git/Filder抓包/Jmeter压测(13)
Hibernate学习(12)
javascript设计模式(12)
JavaScript学习心得(169)
JavaWeb项目笔记(22)
Java知识学习(87)
jQuery(15)
JSP和Servlet(9)
Linux/Nginx/Shell/CPU服务器知识等(56)
MySQL/PostgreSQL数据库知识(54)
NodeJS(48)
更多

```
next() // 确保一定要调用 next()  
}  
})  

```

解析:

1、meta 字段就是路由元信息字段, requiresAuth 是自己起的字段名称, 用来标记这个路由信息是否需要检测, true 表示要检测, false 表示不需要检测 (这个名称随便起, 比如我自己的就起的 requiresId, 建议起个有意义的名称)

2、if (to.matched.some(record => record.meta.requiresAuth)), 如果对这类写法不熟悉, 可以去看看es6的箭头函数, 这句话就是返回遍历的某个路由对象, 我们定义为record, 检测这个对象是否拥有meta这个对象, 如果有meta这个对象, 检测它的meta对象是不是有requiresAuth这个属性, 且为true, 如果满足上述条件, 就确定了是这个/foo/bar路由。

```
function(i){ return i + 1; } //ES5  
(i) => i + 1 //ES6
```

3、this route requires auth, check if logged in, 说明auth信息是必须的, 检验是否登录, 也就是在这个路由下, 如果检测到auth这个用户名, 那么进行下一步操作!

4、案例下面就有了判断, if (!auth.loggedIn()), 通过自己封装的检验方法auth.loggedIn(), 检验用户是否登录, 从而决定渲染下一步操作!

总结, vue-router路由元信息说白了就是通过meta对象中的一些属性来判断当前路由是否需要进一步处理, 如果需要处理, 按照自己想要的效果进行处理即可! 其实比较简单, 大家可以研究下。

5、我项目中实例



随笔档案

2021年9月(55)
2021年8月(55)
2021年7月(54)
2021年6月(80)
2021年5月(54)
2021年4月(15)
2021年3月(14)
2021年2月(18)
2021年1月(21)
2020年12月(27)
2020年11月(42)
2020年10月(4)
2020年9月(11)
2020年8月(33)
2020年7月(9)
更多

最新评论

1. Re:Echarts使用dataset数据集管理数据
看到楼主提示, 想了一下也可以简写成
this.series = Array.from({length:this.l
ineDime.length - 1}).fill({ type:'line'
});...

--扬帆启航yun

2. Re:浅析UI设计与UX设计、UI设计网
站推荐
what's up (/ω \)

--挂路灯

3. Re:浅析UI设计与UX设计、UI设计网
站推荐
whats up (' ▽ | ♥)

```

router.beforeEach((to, from, next) => {
  const token = store.getters.userInfo
  if(to.matched.some(record => record.meta.requireAuth)){
    next()//如果路由中有meta的requireAuth, 且为true, 就不进行登录验证, 否则进行登录验证
  }else{
    if(token){
      next()
    }else{
      if(to.path==="/login"){
        next()
      }else{
        next({path: '/login'})
      }
    }
  }
  return
})

```



//系统模块

```

export default [
  {
    path: '/login',
    name: 'login',
    component: () => import('@/views/system/login')
  },
  {
    path: '/register',

```

--挂路灯

4. Re:浏览器工作原理：浅析浏览器渲染进程 - HTML、CSS和JavaScript是如何变成页面的？（上）

非常nice 感谢

--阿甸的蒲

5. Re:浅析SpringSecurity未授权或权限不足的处理：AuthenticationEntryPoint认证入口点及其实现类简介及AccessDeniedHandler介绍

博主你好，请问，如果在 FilterSecurityInterceptor 过滤链之前，SecurityContext没有添加认证对象，那么 FilterSecurityInterceptor 过滤链...

--爱吃小鱼粮的松鼠

阅读排行榜

1. JS/JQuery获取当前元素的上一个/下一个兄弟级元素等元素的方法(344921)
2. vue里ref (\$refs)用法(169271)
3. Vue上传文件：ElementUI中的upload实现(162533)
4. 集合转数组的toArray()和toArray(T[] a)方法(146445)
5. java中List和Array相互转换(136911)

评论排行榜

1. 详解vue组件的keep-alive(10)
2. Echarts使用dataset数据集管理数据(7)

```
    name: 'register',
    component: () => import('@views/system/register'),
    meta: {requireAuth: true}
  },
  {
    path: '/perfectInfo/:identifier',
    name: 'perfectInfo',
    component: () => import('@views/system/perfectInfo'),
    meta: {requireAuth: true}
  },
  {
    path: '/resetPassword',
    name: 'resetPassword',
    component: () => import('@views/system/resetPassword'),
    meta: {requireAuth: true}
  }
]
```



项目中使用：**通过路由元信息meta实现不同路由展现不同页面（带不同的meta信息，展示不同页面布局）**



```
<template>
  <el-container :class="['app uf-col']">
    <template v-if="$route.meta.fullScreen">
      <router-view></router-view>
    </template>
    <template v-else-if="$route.meta.homePages">
```

3. echarts使用技巧（一）echarts的图表自适应resize问题、单选、缩放等(7)
4. nuxtjs如何在单独的js文件中引入store和router(6)
5. 使用Git将本地项目上传到Github操作详解(6)

推荐排行榜

1. 集合转数组的toArray()和toArray(T[] a)方法(14)
2. vue里ref (\$refs)用法(9)
3. JS/JQuery获取当前元素的上一个/下一个兄弟级元素等元素的方法(9)
4. Vue上传文件：ElementUI中的upload实现(8)
5. java中List和Array相互转换(8)

```
<Nav></Nav>

<router-view></router-view>
</template>
<template v-else>
  <WHeader></WHeader>
  <el-container>
    <WMenu></WMenu>
    <router-view></router-view>
  </el-container>
</template>
</el-container>
</template>
```



//系统模块

```
export default [
  {
    path: '/login',
    name: 'login',
    component: () => import('@/views/system/login'),
    meta: {fullScreen: true}
  },
  {
    path: '/register',
    name: 'register',
    component: () => import('@/views/system/register'),
    meta: {requireAuth: true, fullScreen: true}
  },
]
```

//静态主页模块

```
export default [  
  {  
    path: '/rescue',  
    name: 'rescue',  
    component: () => import('@/views/pages/rescue'),  
    meta: {homePages: true}  
  }  
]
```



然后我们再看下路由导航守卫，加上homePages：如果路由元信息里面requireAuth为true，或者homePages为true，都不拦截



```
router.beforeEach((to, from, next) => {  
  const token = store.getters.userInfo  
  if(to.matched.some(record => record.meta.requireAuth || record.meta.homePages)){  
    next()  
  }else{  
    if(token){  
      next()  
    }else{  
      if(to.path==="/login"){  
        next()  
      }else{  
        next({path: '/login'})  
      }  
    }  
  }  
})
```

```
}  
return  
})
```



二、Array some() 方法

some() 方法用于检测数组中的元素是否满足指定条件（函数提供）。

some() 方法会依次执行数组的每个元素：

- **如果有一个元素满足条件，则表达式返回true，剩余的元素不会再执行检测。**
- 如果没有满足条件的元素，则返回false。

注意： 1、some() 不会对空数组进行检测。 2、some() 不会改变原始数组。

1、语法：array.some(function(currentValue,index,arr),thisValue)

2、参数说明：

function(currentValue, index,arr)：必须。函数，数组中的**每个元素都会执行这个函数**

函数参数：

currentValue：必选。当前元素的值

index：可选。当前元素的索引值

arr：可选。当前元素属于的数组对象

thisValue：可选。对象作为该执行回调时使用，传递给函数，**用作 "this" 的值**。如果省略了 thisValue ， "this" 的值为 "undefined"

3、返回值：布尔值。如果数组中有元素满足条件返回 true，否则返回 false。

4、检测数组 ages 中是否有元素大于输入框输入的值：



```
<p>最小年龄: <input type="number" id="ageToCheck" value="18"></p>
<button onclick="myFunction()">点我</button>
<p>判断结果: <span id="demo"></span></p>
<script>
var ages = [4, 12, 16, 20];
function checkAdult(age) {
    return age >= document.getElementById("ageToCheck").value;
}
function myFunction() {
    document.getElementById("demo").innerHTML = ages.some(checkAdult);
}
</script>
```



分类: Vue框架生态链

好文要顶

关注我

收藏该文



古兰精

关注 - 1

粉丝 - 356

+加关注

3

0

« 上一篇: [修改Nginx与Apache配置参数解决http状态码: 413上传文件大小限制问题](#)

» 下一篇: [解决vuex在页面刷新后数据丢失的问题](#)

posted @ 2018-05-22 22:10 古兰精 阅读(18907) 评论(0) 编辑 收藏 举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

登录后才能查看或发表评论, 立即 [登录](#) 或者 [逛逛](#) 博客园首页

【推荐】百度智能云超值优惠: 新用户首购云服务器1核1G低至69元/年

【推荐】跨平台组态\工控\仿真\CAD 50万行C++源码全开放免费下载!

【推荐】阿里云云大使特惠: 新用户购ECS服务器1核2G最低价87元/年

【推荐】和开发者在一起: 华为开发者社区, 入驻博客园科技品牌专区

【推广】园子与爱卡汽车爱宝险合作, 随手就可以买一份的百万医疗保险



编辑推荐:

- .Net Core with 微服务: 分布式事务 - 可靠消息最终一致性
- 通过 Wireshark 抓包分析谈谈 DNS 域名解析的那些事儿
- 概述 .NET 6 ThreadPool 实现
- 现代操作系统管理内存, 到底是分段还是分页?
- 源码解析 Grpc 拦截器 (C# 版本)

最新新闻:

- 「元宇宙」迟早要被炒糊 (2021-09-23 21:13)
- iPhone 13预售官网被抢崩, 苹果真的降价了吗? (2021-09-23 21:00)
- 从卖咖啡到做教育, 十岁的知乎“折腾”在赚钱路上 (2021-09-23 20:45)

- Facebook宣布元宇宙相关主管晋升CTO，推动AR、VR硬件研发与升级 (2021-09-23 20:30)
- 花呗接入央行征信，会影响申请银行房贷吗？ (2021-09-23 20:18)
- » 更多新闻...

Copyright © 2021 古兰精
Powered by .NET 6 on Kubernetes