

堆栈溢出需要来自其他域的外部 JavaScript，该域被阻止或无法加载。使用其他来源重试。

Vue-i18n - 无法读取未定义的属性"配置"

问 3 年, 5 个月前 9 个月前活动 查看 12k 次



首先，我告诉你什么工作（在应用程序.js）

11



```
import router from './routes.js';
import VueI18n from 'vue-i18n';

const messages = {
  en: {
    message: {
      hello: 'hello world'
    }
  }
}

// Create VueI18n instance with options
const i18n = new VueI18n({
  locale: 'en', // set locale
  messages, // set locale messages
});

const app = new Vue({
  el: '#app',
  router,
  i18n
});
```

但是，如果我想在朗分开的代码.js

```
import VueI18n from 'vue-i18n';

const messages = {
  en: {
    message: {
      hello: 'hello world'
    }
  }
}
```

堆栈溢出需要来自其他域的外部 JavaScript，该域被阻止或无法加载。使用其他来源重试。

```
}  
  
export default new VueI18n({  
  locale: 'en', // set locale  
  messages, // set locale messages  
});
```

这样我就可以在应用程序中写.js

```
import router from './routes.js';  
import i18n from './lang.js';  
  
const app = new Vue({  
  el: '#app',  
  router,  
  i18n  
});
```

但不知何故，即使路线.js也完全相同，这也不起作用。

我的引导.js看起来像这样，如果重要的是知道。

```
import Vue from 'vue';  
  
window.Vue = Vue;  
  
import VueRouter from 'vue-router';  
import VueI18n from 'vue-i18n';  
  
Vue.use(VueRouter);  
Vue.use(VueI18n);
```

我很抱歉的长代码，但不知何故，错误在于从'./lang.js'导入i18n：我得到的消息：未加密的类型：无法读取未定义的属性"配置"

vue.js 维埃-伊18n

堆栈溢出需要来自其他域的外部 JavaScript，该域被阻止或无法加载。使用其他来源重试。



4799 关注者

3,293

5

27

54

3 个答案

积极

古老

票



在创建应用实例的主文件中，将i18n添加为选项，而不是这样： `Vue.use(VueI18n)`

20



```
new Vue({  
  el: '#app',  
  i18n, // < --- HERE  
  store,  
  router,  
  template: '<App/>',  
  components: { App }  
})
```

把它放在之后 `el`；

这应该是你的语言：

```
import Vue from 'vue'  
import VueI18n from 'vue-i18n'  
import en from './en'  
import fr from './fr'  
import ro from './ro'
```

```
Vue.use(VueI18n)
```

```
export default new VueI18n({  
  locale: 'en',  
  fallbackLocale: 'en',  
  messages: {  
    en, fr, ro  
  }  
})
```

堆栈溢出需要来自其他域的外部 JavaScript，该域被阻止或无法加载。使用其他来源重试。



丑. 荣博小

8,445 4 33 52

只需添加详细信息，您**必须**在使用Vue.use (VueI18n) **后**构建新的VueI18n

10



```
Vue.use(VueI18n);
// must be called after vue.use
const i18n = new VueI18n({
  locale: "en",
  fallbackLocale: "en",
  messages: {
    en
  }
});
new Vue({
  el: "#app",
  i18n,
  render: (h) => h(App),
});
```

共享 改进此答案 跟随

回答1月22'20在15:15



西里亚克

101 1 4

代码分离的想法可以这样实现

0



```
/* i18n.ts */

import VueI18n from "vue-i18n";

import en from "../locales/en.json";

// Export as an arrow function
export default () =>
  new VueI18n({
    locale: process.env.VUE_APP_I18N_LOCALE || "en",
    fallbackLocale: process.env.VUE_APP_I18N_FALLBACK_LOCALE || "en",
```

堆栈溢出需要来自其他域的外部 JavaScript，该域被阻止或无法加载。使用其他来源重试。

```
/* plugins.ts */

import Vue from "vue";

// Plugins
import VueI18n from "vue-i18n";

import i18n from "./i18n";

Vue.use(VueI18n);

export default {
  i18n: i18n()
};
```

最后， main.ts

```
/* main.ts */

import Vue from "vue";

import plugins from "./core/plugins";

import App from "./App.vue";

new Vue({
  ...plugins,
  render: (h) => h(App)
}).$mount("#app");
```

诀窍是在导出 i18n 设置时使用箭头功能。值得一提的是，此问题不会发生在其他一些 Vue 插件中。

共享 改进此答案 跟随

编辑10月17'20在12:18



大卫·巴克

3,462 29 24 31

回答10月17'20在11:52



克尔·雅科夫列夫

1 2

