

从所有教程的词条中查询...



首页 > 慕课教程 > ElementPlus 的组件库自主开发 > 将组件 Render 到 DOM 节点上

全部开发者教程



张轩 • 更新于 2023-07-24

◀ 上一节 需求分析 总结 下一节 ▶

将组件 Render 到 DOM 节点上

使用 createApp 的弊端

- 这个方法太重了，它其实返回的是一个应用的实例，而我们这里需要轻量级的解决方案。

隆重推出 render 函数

<> 代码块

```
1 // 一个 vue 内部神奇的函数，文档中都没有特别的记录
2 // 它负责将一个 vnode 渲染到 dom 节点上
3 // 它是一个很轻量级的解决方案
4 import { render } from 'vue'
5
6 render(vNode, DOM节点)
```

通过 VNode 拿到组件实例

```
const component = vNode.component
// 它是一个特殊类型，称之为 ComponentInternalInstance
```

📝 意见反馈

♥️ 收藏教程

🔖 标记书签

索引目录



[将组件 Render 到 DOM 节点上](#)



```
5 // 在组件内部也可以使用一个特殊的函数获取这个实例
6 const instance = getCurrentInstance()
  console.log('inner instance', instance)
```

shallowReactive

文档地址: <https://cn.vuejs.org/api/reactivity-advanced.html#shallowreactive>

- 和 `reactive()` 不同, 这里没有深层级的转换: 一个浅层响应式对象里只有根级别的属性是响应式的。属性的值会被原样存储和暴露。
- 假如是数组的话, 创建一个浅层响应式的空数组, 这意味着数组的元素不会被递归地转换成响应式对象。当我们对数组进行一些增删改操作时, Vue 会自动检测到这些变化, 并更新对应的视图。

剩余开发计划

- 添加手动删除
- 添加 `zIndex`
- 添加键盘关闭 (按 `esc` 可以关闭 `message`)
- `hover` 到 `Message` 上面的时候不会自动关闭
- 添加动画以及样式

需求分析 ◀ 上一节 下一节 ▶ 总结

 我要提出意见反馈





 意见反馈

 收藏教程

 标记书签