慕课网首页 免费课 实战课 体系课 慕课教程 专栏 手记 企业服务

Sunday • 更新于 2022-10-19

我的课程



Q

06: 框架字现: re

从所有教程的词条中查询…

全部开发者教程 Ξ

01: 前言

02: 源码阅读: ref 复杂数据 类型的响应性

03: 框架实现: ref 函数 - 构 建复杂数据类型的响应性

04: 总结: ref 复杂数据类型 的响应性

05: 源码阅读: ref 简单数据 类型的响应性

06: 框架实现: ref 函数 - 构 建简单数据类型的响应性

07: 总结: ref 简单数据类型 响应性

08: 总结

第七章:响应系统 computed && watch

01: 开篇

02: 源码阅读: computed 的 响应

03: 框架实现: 构建

```
◆ 上一节 05: 源码阅读: ...07: 总结: ref ...下一节 →
06: 框架实现: ref 函数 - 构建简单数据类型的响应性
```

首页 > 慕课教程 > Vue3源码分析与构建方案 > 06: 框架实现: ref 函数 - 构建简单数据类型的...

1. 在 packages/reactivity/src/ref.ts 中, 完善 set value 函数:

```
<> 代码块
     class RefImpl<T> {
1
2
        private _value: T
        private _rawValue: T
3
 5
 6
         constructor(value: T, public readonly __v_isShallow: boolean) {
8
9
            // 原始数据
            this._rawValue = value
10
11
12
13
14
         set value(newVal) {
15
16
17
             * newVal 为新数据
18
             * this._rawValue 为旧数据(原始数据)
             * 对比两个数据是否发生了变化
19
             */
20
            if (hasChanged(newVal, this._rawValue)) {
21
               // 更新原始数据
22
23
                this._rawValue = newVal
24
                // 更新 .value 的值
25
                this._value = toReactive(newVal)
26
                // 触发依赖
                triggerRefValue(this)
2.7
28
29
         }
30
     }
31
32
     * 为 ref 的 value 进行触发依赖工作
33
34
     export function triggerRefValue(ref) {
35
36
       if (ref.dep) {
37
            triggerEffects(ref.dep)
38
39
```

2. 在 packages/shared/src/index.ts 中, 新增 hasChanged 方法

```
<>代码块
1
     * 对比两个数据是否发生了改变
2
3
     export const hasChanged = (value: any, oldValue: any): boolean =>
         !Object.is(value, oldValue)
```

▶ 意见反馈

♡ 收藏教程

□ 标记书签

索引目录

?

 \odot

我们可以创建对应测试实例: packages/vue/examples/reactivity/ref-shallow.html <> 代码块 1 <script> const { ref, effect } = Vue 3 const obj = ref('张三') // 调用 effect 方法 effect(() => { document.querySelector('#app').innerText = obj.value 8 9 10 setTimeout(() => { 12 obj.value = '李四' 13 }, **2000)**; 14 </script> 测试成功,表示代码完成。 ✓ 我要提出意见反馈

企业服务 网站地图 网站首页 关于我们 联系我们 讲师招募 帮助中心 意见反馈 代码托管

★ % 6 **4**

Copyright © 2022 imooc.com All Rights Reserved | 京ICP备 12003892号-11 京公网安备11010802030151号





·