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

Q 🃜 💄 我的课程



Q

从所有教程的词条中查询…

首页 > 慕课教程 > Vue3源码分析与构建方案 > 10: 框架实现: 构建 trigger 触发依赖

全部开发者教程 :Ξ

07: 框架实现: 构建 effect 函数, 生成 ReactiveEffect 实例

08: 框架实现: track && trigger

09:框架实现:构建 track 依赖收集函数

10:框架实现:构建 trigger 触发依赖

11: 总结: 单一依赖的 reactive

12: 功能升级: 响应数据对应 多个 effect

13: 框架实现: 构建 Dep 模块, 处理一对多的依赖关系

14: reactive 函数的局限性

15: 总结

第六章:响应系统 - ref的响应性

01: 前言

Sunday・更新于 2022-10-19

↓ 上一节 09: 框架实现: ...11: 总结: 单一... 下一节 ▶

索引目录

10: 框架字现: 核

?

··

10: 框架实现: 构建 trigger 触发依赖

那么在上一小节中,我们已经成功保存依赖到 WeakMap 中了,那么接下来我们就可以在 setter 的时候触发保存的依赖,以此来达到 响应性 数据的效果了。

在 packages/reactivity/src/effect.ts 中

```
()代码块
1 /**
     * 触发依赖的方法
2
     * @param target WeakMap 的 key
3
     st @param key 代理对象的 key, 当依赖被触发时, 需要根据该 key 获取
4
5
6
    export function trigger(
        target: object,
8
        key?: unknown
9
    ) {
        // 依据 target 获取存储的 map 实例
10
        const depsMap = targetMap.get(target)
11
        // 如果 map 不存在,则直接 return
12
        if (!depsMap) {
13
14
15
        // 依据 key,从 depsMap 中取出 value,该 value 是一个 ReactiveEffect 类型的数据
16
        const effect = depsMap.get(key) as ReactiveEffect
17
        // 如果 effect 不存在,则直接 return
18
19
        if (!effect) {
            return
21
        // 执行 effect 中保存的 fn 函数
22
        effect.fn()
2.3
24
```

此时,我们就可以在触发 setter 时,执行保存的 fn 函数了。

那么接下来我们实现对应的测试实例,在 packages/vue/examples/reactivity/reactive.html 中:

```
()代码块
    <script>
      const { reactive, effect } = Vue
 4
       const obj = reactive({
 5
        name: '张三'
 6
       })
 8
       // 调用 effect 方法
       effect(() => {
10
        document.querySelector('#app').innerText = obj.name
11
12
13
     setTimeout(() => {
        obj.name = '李四'
14
      }, 2000);
15
     </script>
```

♪ 意见反馈

♡ 收藏教程

口 标记书签

那么,至此我们就已经完成了一个简单的 响应式依赖数据处理

✔ 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 A

⊡

?

 \odot

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

♪ 意见反馈

♡ 收藏教程

口 标记书签