慕课网首页 免费课 体系课 慕课教程 专栏 手记 企业服务 实战课

我的课程



Q

03: 如何让你的糕

?

··

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 > 03: 如何让你的程序变得更加 "聪明" ?

全部开发者教程 Ξ

09: 引入代码格式化工具: prettier 让你的代码结构更加 规范

10: 模块打包器: rollup

11: 初见框架雏形: 配置路径 映射

12: 总结

第四章:响应系统-响应系 统的核心设计原则

01: 前言

02: JS 的程序件

03: 如何让你的程序变得更加 "聪明"?

04: vue 2 的响应性核心 API: Object.defineProperty

05: Object.defineProperty 在设计层的缺陷

06: vue3的响应性核心 API:

Sunday • 更新于 2022-10-19

◆ 上一节 02: JS 的程序性 04: vue 2 的响... 下一节 ▶

索引目录

03: 如何让你的程序变得更加"聪明"?

你为了让你的程序变得更加"聪明", 所以你开始想: "如果数据变化了, 重新执行运算就好了"。

那么怎么去做呢?你进行了一个这样的初步设想:

- 1. 创建一个函数 effect, 在其内部封装 计算总价格的表达式
- 2. 在第一次打印总价格之前, 执行 effect 方法
- 3. 在第二次打印总价格之前, 执行 effect 方法

那么这样我们是不是就可以在第二次打印时,得到我们想要的 50 了呢?

所以据此, 你得到了如下的代码:

```
<>代码块
    <script>
     // 定义一个商品对象,包含价格和数量
     let product = {
       price: 10,
5
       quantity: 2
6
      }
      // 总价格
8
      let total = 0;
      // 计算总价格的匿名函数
    + let effect = () => {
    + total = product.price * product.quantity;
11
12
      // 第一次打印
1.3
    + effect();
14
15
     console.log(`总价格: ${total}`); // 总价格: 20
    // 修改了商品的数量
16
17
    product.quantity = 5;
     // 第二次打印
18
19
    + effect();
      console.log(`总价格: ${total}`); // 总价格: 50
2.0
21
    </script>
```

在这样的代码中,我们成功的让第二次打印得到了我们期望的结果:数据变化了,运算也重新执行了。

但是大家也可以发现,在我们当前的代码中存在一个明显的问题,那就是: **必须主动在数量发生变化之 后,重新主动执行 effect **才可以得到我们想要的结果。那么这样未免太麻烦了。有什么好的办法吗?

肯定是有的,我们继续来往下看~~~~

▶ 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

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

⊡

?

.

0