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 Ē 我的课程

Q

12: Vue 3 框架设

?

 $\odot$ 

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 > 12: Vue 3 框架设计概述

#### 全部开发者教程 Ξ

06: 为什么说框架的设计过程 其实是一个不断取舍的过程?

07: .vue 中的 html 是真实的 html 吗?

08: 什么是运行时?

09: 什么是编译时?

10: 运行时 + 编译时

11: 什么是副作用

12: Vue 3 框架设计概述

13: 扩展: 所谓良好的 `TypeScript` 支持,是如何提

供的?

14: 总结

#### 第三章: Vue 3源码结构 -搭建框架雏形

01: 前言

02: 探索源码设计: Vue3 源

# Sunday • 更新于 2022-10-19

◆ 上一节 11: 什么是副作用 13: 扩展: 所谓... 下一节 →

### 索引目录

## 12: Vue 3 框架设计概述

根据前面的学习我们已经知道了:

- 1. 什么是声明式
- 2. 什么是命令式
- 3. 什么是运行时
- 4. 什么是编译时
- 5. 什么是运行时+编译时
- 6. 同时也知道了框架的设计过程本身是一个不断取舍的过程

那么了解了这些内容之后,下来 vue3 的一个基本框架设计:

对于 vue3 而言,核心大致可以分为三大模块:

1. 响应性: reactivity 2. 运行时: runtime 3. 编译器: compiler

我们以以下基本结构来描述一下三者之间的基本关系:

```
<> 代码块
     <template>
 2
         <div>{{ proxyTarget.name }}</div>
3
     </template>
 5
     <script>
 6
     import { reactive } from 'vue'
     export default {
 8
        setup() {
 9
            const target = {
                name: '张三'
1.0
11
12
            const proxyTarget = reactive(target)
13
            return {
14
                proxyTarget
1.5
16
         }
17
     }
18
     </script>
19
```

#### 在以上代码中:

- 1. 首先, 我们通过 reactive 方法, 声明了一个响应式数据。
  - 1. 该方法是 reactivity 模块对外暴露的一个方法
  - 2. 可以接收一个复杂数据类型,作为 Proxy (现在很多同学可能还不了解什么是 proxy ,没有关 系后面我们会详细介绍它,现在只需要有个印象即可)的 被代理对象(target)
  - 3. 返回一个 Proxy 类型的 代理对象 (proxyTarget)







- 2. 然后,我们在 tempalte 标签中,写入了一个 div。我们知道这里所写入的 html 并不是真实的 html ,我们可以把它叫做 模板,该模板的内容会被 编译器 ( compiler ) 进行编译,从而生成一个 render 函数
- 3. 最后, vue 会利用 运行时 (runtime) 来执行 render 函数,从而渲染出真实 dom

以上就是 reactivity、runtime、compiler 三者之间的运行关系。

当然除了这三者之外, vue 还提供了很多其他的模块,比如: SSR ,我们这里只是 **概述了基本的运行逻辑。** 

✔ 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

. •

3