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

17: 框架字现: 新

?

 $\odot$ 

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 > 17: 框架实现: 新建 compat 模块, 把 render...

## 全部开发者教程 Ξ

12: 框架实现: 外理根节点的 转化,生成 JavaScript AST

13: 扩展知识: render 函数的 生成方案

14: 源码阅读: 编译器第三 步: 生成 render 函数

15: 框架实现: 构建 CodegenContext 上下文对象

16: 框架实现: 解析 JavaScript AST,拼接 render 函数

17: 框架实现: 新建 compat 模块,把 render 转化为

18: 总结

## 第十五章: compiler 编译 器 - 深入编辑器处理逻辑

01: 前言

02: 响应性数据的编辑器处 理:响应性数据的处理逻辑

Sunday • 更新于 2022-10-19

↓ 上一节 16: 框架实现: ...18: 总结 下一节 ▶

索引目录

## 17: 框架实现: 新建 compat 模块, 把 render 转化为 function

此时,我们的 render 函数构建,已经可以完成了。但是我们当前的 render 本质上还是一个字符串, 所以我们需要通过 new Function 来把它变为函数。

那么这样的一个 new Function 的过程, 我们其实可以在 vue 中完成。

- 1. 创建 packages/vue-compat/src/index.ts 模块
- 2. 新增 compileToFunction 方法:

```
<>代码块
 1
      import { compile } from '@vue/compiler-dom'
      function compileToFunction(template, options?) {
         const { code } = compile(template, options)
 5
 6
         const render = new Function(code)()
         return render
     }
10
11
      export { compileToFunction as compile }
```

3. 在 packages/vue/src/index.ts 中修改 compile 的导出:

```
<> 代码块
     // export { compile } from '@vue/compiler-dom'
     export { compile } from '@vue/vue-compat'
```

4. 修改测试实例:

```
<> 代码块
       const { compile, h, render } = Vue
       // 创建 template
       const template = `<div> hello world </div>`
       // 生成 render 函数
 6
       const renderFn = compile(template)
 9
       // 创建组件
10
      const component = {
11
        render: renderFn
12
13
14
       // 通过 h 函数, 生成 vnode
       const vnode = h(component)
       // 通过 render 函数渲染组件
17
       render(vnode, document.querySelector('#app'))
18
19
     </script>
```

✔ 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

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

?

.

0