

全部开发者教程

三

05: 框架实现：构建 h 函数，
处理 ELEMENT +
ARRAY_CHILDREN 场景

06: 源码阅读：h 函数，组件
的本质与对应的 VNode

07: 框架实现：处理组件的
VNode

08: 源码阅读：h 函数，跟踪
Text、Comment、
Fragment 场景

09: 框架实现：实现剩余场景
Text09: 框架实现：实现剩余
场景 Text、Comment、
Fragment

10: 源码阅读：对 class 和
style 的增强处理

11: 框架实现：完成虚拟节点
下的 class 和 style 的增强

12: 总结

第十章: runtime 运行时 -
构建 render 渲染器



Sunday • 更新于 2022-10-19

◀ 上一节 10: 源码阅读：... 12: 总结 下一节 ▶

11: 框架实现：完成虚拟节点下的 class 和 style 的增强

那么明确好了增强的原理之后，那么下面我们就可以进行对应的实现了。

1. 创建 packages/shared/src/normalizeProp.ts：

<> 代码块

```
1  import { isArray, isObject, isString } from '.'
2
3  /**
4   * 规范化 class 类，处理 class 的增强
5   */
6  export function normalizeClass(value: unknown): string {
7    let res = ''
8    // 判断是否为 string，如果是 string 就不需要专门处理
9    if (isString(value)) {
10      res = value
11    }
12    // 额外的数组增强。官方案例：https://cn.vuejs.org/guide/essentials/class-and-style.
13    else if (isArray(value)) {
14      // 循环得到数组中的每个元素，通过 normalizeClass 方法进行迭代处理
15      for (let i = 0; i < value.length; i++) {
16        const normalized = normalizeClass(value[i])
17        if (normalized) {
18          res += normalized + ' '
19        }
20      }
21    }
22    // 额外的对象增强。官方案例：https://cn.vuejs.org/guide/essentials/class-and-style.
23    else if (isObject(value)) {
24      // for in 获取到所有的 key，这里的 key (name) 即为 类名。value 为 boolean 值
25      for (const name in value as object) {
26        // 把 value 当做 boolean 来看，拼接 name
27        if ((value as object)[name]) {
28          res += name + ' '
29        }
30      }
31    }
32    // 去左右空格
33    return res.trim()
34  }
```

2. 在 packages/runtime-core/src/vnode.ts 的 createVNode 增加判定：

<> 代码块

```
1  if (props) {
2    // 处理 class
3    let { class: klass, style } = props
4    if (klass && !isString(klass)) {
5      props.class = normalizeClass(klass)
6    }
7  }
```

索引目录

11: 框架实现：完



可以创建 `packages/vue/examples/runtime/h-element-class.html` 测试用例：

<> 代码块

```
1  <script>
2    const { h, render } = Vue
3
4    const vnode = h('div', {
5      class: {
6        'red': true
7      }
8    }, '增强的 class')
9
10   console.log(vnode);
11 </script>
```

打印可以获取到正确的 `vnode`。

10: 源码阅读: 对 class 和 style 的增强处理 ◀ 上一节 下一节 ▶ 12: 总结

 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



 意见反馈

 收藏教程

 标记书签