

全部开发者教程

04：框架实现：基于 renderer 完成 ELEMENT 节点挂载

05：框架实现：合并渲染架构，得到可用的 render 函数

06：源码阅读：渲染更新，ELEMENT 节点的更新操作

07：框架实现：渲染更新，ELEMENT 节点的更新实现

08：源码阅读：新旧节点不同元素时，ELEMENT 节点的更新操作

09：框架实现：处理新旧节点不同元素时，

10：框架实现：删除元素，ELEMENT 节点的卸载操作

11：源码阅读：class 属性和其他属性的区分挂载

12：深入属性挂载：HTML Attributes 和 DOM



Sunday • 更新于 2022-10-19

◀ 上一节 09：框架实现：... 11：源码阅读：... 下一节 ▶

10：框架实现：删除元素，ELEMENT 节点的卸载操作

那么此时我们已经有了 `unmount` 函数，我们知道触发 `unmount` 函数，即可卸载元素。

那么接下来我们就可以基于这样的函数来去实现 **卸载** 操作了。

创建如下测试实例 `packages/vue/examples/imooc/runtime/render-element-remove.html`：

<> 代码块

```
1  <script>
2    const { h, render } = Vue
3
4    const vnode = h('div', {
5      class: 'test'
6    }, 'hello render')
7    // 挂载
8    render(vnode, document.querySelector('#app'))
9
10   // 延迟两秒，执行卸载操作
11   setTimeout(() => {
12     render(null, document.querySelector('#app'))
13   }, 2000);
14 </script>
```

当我们触发 `render(null, document.querySelector('#app'))` 时，`vue` 会删除之前渲染的 `vnode`。即为：**卸载操作**。

这里查看对应源码逻辑，也非常简单，查看 `packages/runtime-core/src/renderer.ts` 中 `render` 函数：

<> 代码块

```
1  const render: RootRenderFunction = (vnode, container, isSVG) => {
2    // 不存在新的 vnode 时
3    if (vnode == null) {
4      // 但是存在旧的 vnode
5      if (container._vnode) {
6        // 则直接执行卸载操作
7        unmount(container._vnode, null, null, true)
8      }
9    } else {
10     ...
11   }
12   ...
13 }
```

这块代码比较简单，我们直接实现即可：

在 `packages/runtime-core/src/renderer.ts` 中为 `render` 函数补充卸载逻辑：

<> 代码块

```
1  /**
2   * 渲染函数
3   */
4  const render = (vnode, container) => {
```

索引目录

10：框架实现：册



```
7         if (container._vnode) {
8             unmount(container._vnode)
9         }
10    } else {
11        // 打补丁（包括了挂载和更新）
12        patch(container._vnode || null, vnode, container)
13    }
14    container._vnode = vnode
15 }
```

创建对应测试实例，卸载成功。

09: 框架实现：处理新旧节点不同元素时， [← 上一节](#) [下一节 ▶](#) 11: 源码阅读：class 属性和其他属性的区分...

[✎ 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](#)