

从本章开始我们将要在 `VNode` 的基础上构建 `renderer` 渲染器。

根据上一章中的描述我们知道，在 `packages/runtime-core/src/renderer.ts` 中存放渲染器相关的内容。

Vue 提供了一个 `baseCreateRenderer` 的函数，该函数会返回一个对象，我们把返回的这个对象叫做 `renderer` 渲染器。

对于该对象而言，提供了三个方法：

1. `render` : 渲染函数
2. `hydrate` : 服务端渲染相关
3. `createApp` : 初始化方法

查看 `baseCreateRenderer` 的代码，我们也可以发现整个 `baseCreateRenderer` 中包含了 2000 行代码，可见整个渲染器是非常复杂的。

所以说，对于我们的实现而言，还是和之前一样，我们将谨遵 **没有使用就当做不存在** 和 **最少代码的实现逻辑** 这两个核心思想，来构建整个 `render` 的过程。

在实现 `render` 的过程中，我们也会和学习 `h` 函数时一样，利用 `h` 函数时的测试实例，配合上 `render` 函数来分类型进行依次处理。

那么明确好了这些内容之后，我们下面就来进入渲染器的世界吧。