

vue源码

Vue3 MVVM - 3. Ref

Ref的作用和Reactive 基本一样，却可以让普通值成为响应式。其内部使用了reactive。Ref会返回一个可变的响应式对象，其维护着内部的值。这么说比较抽象，我们上代码吧。

```
1 function ref(val) {
2   if (typeof val === 'object' && val.__v_isRef) return
3   let _value = typeof val === 'object' ? reactive(val) : val
4   const r = {
5     __v_isRef: true,
6     get value() {
7       collectDependencies(r, 'value'),
8       return _value
9     },
10    set value(newValue) {
11      if (newValue !== _value) {
12        val = newValue
13        _value = typeof val === 'object' ? reactive(val) : val
14
15        trigger(r, 'value')
16      }
17    }
18  }
19
20  return r
21 }
```

这里最主要的部分就是这个 r 对象，这个对象就是上面代码里面val变量的包装。所以这就是为什么普通值也可以是响应式因为r对象内部的get 和 set 里面收集依赖的target 其实就是r.value 自己。那么这样等于和传入的值已经没有关联了，所有的r.value更新会让所有收集的依赖都运行。实现的思想就是使用一个额外的对象作为包装。