



```

51         const newValue = effect.run()
52         if (deep || hasChanged(newValue, oldValue)) {
53             cb(newValue, oldValue)
54             oldValue = newValue
55         }
56     }
57 }
58
59 // 调度器
60 let scheduler = () => queuePreFlushCb(job)
61
62 const effect = new ReactiveEffect(getter, scheduler)
63
64 if (cb) {
65     if (immediate) {
66         job()
67     } else {
68         oldValue = effect.run()
69     }
70 } else {
71     effect.run()
72 }
73
74 return () => {
75     effect.stop()
76 }
77 }

```

2. 在 `packages/reactivity/src/reactive.ts` 为 `reactive` 类型的数据, 创建 **标记**:

```

<> 代码块
1  export const enum ReactiveFlags {
2      IS_REACTIVE = '__v_isReactive'
3  }
4
5  function createReactiveObject(
6      ...
7  ) {
8      ...
9      // 未被代理则生成 proxy 实例
10     const proxy = new Proxy(target, baseHandlers)
11     // 为 Reactive 增加标记
12     proxy[ReactiveFlags.IS_REACTIVE] = true
13     ...
14 }

```

```

<> 代码块
1  /**
2   * 判断一个数据是否为 Reactive
3   */
4  export function isReactive(value): boolean {
5      return !!value && value[ReactiveFlags.IS_REACTIVE]
6  }

```

3. 在 `packages/shared/src/index.ts` 中创建 `EMPTY_OBJ` :

```

<> 代码块
1  /**
2   * 只读的空对象
3   */
4  export const EMPTY_OBJ: { readonly [key: string]: any } = {}

```

4. 在 `packages/runtime-core/src/index.ts` 和 `packages/vue/src/index.ts` 中导出 `watch` 函数

5. 创建测试实例 `packages/vue/examples/reactivity/watch.html` :

<> 代码块

```
1  <script>
2    const { reactive, watch } = Vue
3
4    const obj = reactive({
5      name: '张三'
6    })
7
8
9    watch(obj, (value, oldValue) => {
10      console.log('watch 监听被触发');
11      console.log('value', value);
12    })
13
14    setTimeout(() => {
15      obj.name = '李四'
16    }, 2000);
17  </script>
```

此时运行项目，却发现，当前存在一个问题，那就是 **watch 监听不到 reactive 的变化**

那么这是因为什么呢？

大家可以先思考下这个问题，我们下一小节继续来说~~~

08: 框架实现：深入 scheduler 调度系统实... ◀ 上一节      下一节 ▶ 10: 问题分析：watch 下的依赖收集

✎ 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