#### Vue3常见面试题

注：所有的回答保持条理清晰，避免不做分门别类地夸夸其谈。因为总结与思考也是实战经验和熟悉程度的体现。以下内容可做参考。

从 vue2 到 vue3 有哪些升级更新，谈谈你的理解？

1. 底层数据劫持的变化Object.defineProperty 到 Proxy，解决了对象动态属性的代理问题（不必通过Vue.$set来hack），包括数组一些原生方法不能触发set而需要框架重写这类方法的问题。（兼容性可以顺带一提）
2. 组件形态的变化，由配置式向组合式演进。如vue2组件的一个生命周期只有一次配置，组合API则不限制多次使用。（其他细节可以自由发挥）
3. 组合API的优势。提供强大的逻辑/代码复用能力（虽然mixin也能解决，但是mixin存在数据源不明确，同名属性会覆盖的问题，越庞大越难维护，如果想回答这个点，建议先了解一下mixin），同一业务的整合能力，避免涉及到this的指向问题
4. 其他一些更新细节。如createApp，teleport、v-if & v-for 、知道什么就列举什么，没有哪个面试官会拒绝一个知识面丰富的候选人。
5. vue3中还能用vue2的语法吗？

可以的，但还要注意不兼容的更新（https://v3.cn.vuejs.org/guide/migration/introduction.html#非兼容的变更）。

1. vue2中想使用vue3的语法，你会怎么做？

务必实践一下，不要背记答案！只要实践过，就自己组织语言自信地回答。

import VueCompositionApi from '@vue/composition-api'

Vue.use(VueCompositionApi)

**vue3 源码了解过吗？如果了解过，谈谈响应式或diff算法的实现？**

1. 双指针变异算法（virtual DOM的diff）：熟悉课程源码
2. 响应库reactivity从状态的声明，到依赖的track，到副作用的trigger讲一遍。
3. 干说不如现场用笔写出reactivity的伪码：将课上实现的核心部分了熟于心，边讲变写。

**react与vue有什么不同？讲讲你的理解。**

开放题，说细了会讲出一本书，抓住重点来说即可：

相同点

1. 均有类似的生命周期概念（哪些相似自己说），有配套的路由、状态管理库、自上而下的状态渲染机制（虽然vue有双向绑定的说法，但这是局部的语法糖的实现，本身依然是单向数据流）、虚拟dom的对比。都是前端比较优秀的MV\*框架。（其他自己发挥，如一改jquery命令式的编程方式，UI = F(state，action)，把界面的维护解耦为state + action的维护）
2. 状态管理库的设计和使用也极为相似：

redux: action => reducer => store => connect(mapStateToProps, mapDispatchToProps)

vuex: actions => mutations => store => mapState/mapMutations + getters

不同点

1. react推崇的是状态immutable，vue（同mobx）恰恰是mutable。

即前者要求数据不允许被改变，每次更新前都是返回一份新的状态。而后者始终在操作一个状态。

因此，react要通过递归节点树找到需要更新的节点，而vue通过代理源对象，能够在对象属性变化时，直接细粒度地去更新跟踪了这个属性的节点。

又因此，vue是天然对新手友好的框架，性能优化在框架层面就做得十分优秀。而react的优化对开发者的技术要求更高，需要掌握一系列的优化手段（memo、useCallback、useMemo、useRef...）。

第一点也决定了react hooks闭包内外的状态往往随着更新变得不一致，新手易导致各种bug！vue则没有这种困扰。