

vue3面试题及答案（vue面试题必问题和答案汇总）

1、Vue3 出现解决了什么问题？它有哪些优势？

Vue2 代码模式下存在的几个问题：

- 随着功能增加，复杂的组件代码越来越多，变得难以维护。主要原因是 vue2 通过选项式API组织的代码，一个逻辑功能可能有多处代码，不易别人上手。
- 缺少一个比较干净的在多个组件之间可以提取和复用逻辑的机制。
- 类型推断不友好。
- 没有静态类型检查，只有代码运行之后才能发现错误等等。

Vue3 是在 Vue2 的基础上进行了一些优化，对 typeScript 有了更好的支持。Vue3 和 Vue2 是可以共存的，Vue3 能够向下兼容支持选项式 API，同时又新增了一些特性，大大提升了性能。

Vue3 的优势：

- 性能更好
- 体积更小
- 更好地 ts 支持
- 更好的代码组织
- 更好的逻辑抽离
- 更多新的功能



vue 3

2、Vue3 新特性有哪些？

vue3的新增特性有：

1、性能提升

- 响应式性能提升，由原来的 `Object.defineProperty` 改为基于ES6的 `Proxy`，使其速度更快，消除警告。
- 重写了 `Vdom`，突破了 `Vdom` 的性能瓶颈。
- 进行模板编译优化。
- 更加高效的组件初始化。

2、更好的支持 `typeScript`

有更好的类型推断，使得 `Vue3` 把 `typeScript` 支持得非常好。

3、新增 `Composition API`

`Composition API` 是 `vue3` 新增的功能，比 `mixin` 更强大。它可以把各个功能模块独立开来，提高代码逻辑的可复用性，同时代码压缩性更强。

4、新增组件

- `Fragment` 不再限制 `template` 只有一个根节点。
- `Teleport` 传送门，允许我们将控制的内容传送到任意的 `DOM` 中。
- `Suspense` 等待异步组件时渲染一些额外的内容，让应用有更好的用户体验。

5、Tree-shaking：支持摇树优化

摇树优化后会将不需要的模块修剪掉，真正需要的模块打到包内。优化后的项目体积只有原来的一半，加载速度更快。

6、Custom Renderer API：自定义渲染器

实现 `DOM` 的方式进行 `WebGL` 编程。

3、`vue2` 和 `vue3` 的响应式有什么区别？

`vue2` 的响应式原理：

- 对象：是通过 `Object.defineProperty` 对 对象的已有属性值的读取和修改进行劫持。

```
1 Object.defineProperty( 'data', 'count', { get(){ }, set(){ } }
2 })
```

存在的问题：对象直接添加新属性或删除已有属性的时候，界面不会自动更新。

- 数组：是通过重写数组更新数组一系列更新元素的方法，来实现元素修改的劫持。

```
1 // 伪代码
2 const originalProto = Array.prototype
3 const arrayProto = Object.create(originalProto)
4 ['push', 'pop', 'shift', 'unshift', 'splice', 'reverse', 'sort'].forEach(key=>{
5   arrayProto[key] = function() {
6     originalProto[key].apply(this, arguments)
7     notifyUpdate()
8   }
9 })
```

存在问题：直接通过下标替换元素或更新`length`的时候，界面不会自动更新。

针对 vue2 对象和数组的某些更新，界面不能自动更新的问题，vue2 通过 \$set 方法，实现视图的实时更新。

vue3的响应式原理：

vue3 响应式是使用 ES6 的 proxy 和 Reflect 相互配合实现数据响应式，解决了 vue2 中视图不能自动更新的问题。

proxy 是深度监听，所以可以监听对象和数组内的任意元素，从而可以实现视图实时更新。

4、vue3 组合式API生命周期钩子函数有变化吗？

setup 是围绕 beforeCreate 和 created 生命周期钩子运行的，所以不需要显示的定義它們。其他的钩子都可以编写到 setup 内。

值得注意的是组合式API中的钩子函数，通过在生命周期钩子前面加上 “on” 来访问组件的生命周期钩子，需要注册，并且只能在 setup 期间同步使用，因为它们依赖于内部的全局状态来定位当前组件实例。

```
1 import { onMounted } from "vue" export default {
2   setup() { // mounted
3     onMounted(() => { console.log('Component is mounted!')
4   })
5 }
6 }
```

下图是选项式API 和 组合式API 生命周期钩子对比：

选项式 API	Hook inside setup
beforeCreate	Not needed*
created	Not needed*
beforeMount	onBeforeMount
mounted	onMounted
beforeUpdate	onBeforeUpdate
updated	onUpdated
beforeUnmount	onBeforeUnmount
unmounted	onUnmounted
errorCaptured	onErrorCaptured
renderTracked	onRenderTracked
renderTriggered	onRenderTriggered
activated	onActivated
deactivated	onDeactivated

5、Composition API 与 Options API 有什么区别？

Options API 是啥？

vue2 中我们把一个 vue 文件中 data、methods、props、mounted、computed 等定义属性和方法，共同处理页面逻辑，这种方式叫做 Options API。

这种方式开发的复杂组件，同一个功能的往往需要在不同 vue 配置项中定义属性和方法，代码比较分散。如果功能比较复杂，维护代码的时候往往会很难分清每个方法对应的功能，增加了代码维护成本。所以 vue3 舍弃了 Options API，换用 Composition API。

Composition API 是啥？

Composition API 是 vue3 新增的，所以 vue2 没有。在 Composition API 中，根据代码逻辑功能来组织的，一个功能所定义的所有 API 都会放到一起，这样即使功能复杂，代码量增大，都可以一下子定位到某个功能的所有代码，代码维护方便。它的最大特点就是：高内聚，低耦合。

vue3 仍然支持 options API，但我们更推荐使用 Composition API。优劣比较：

- 更好的可编程性。
- 更优的代码组织。
- 更好的逻辑抽象能力。
- 对 tree-shaking 友好，代码也更容易压缩。
- 没有 this，没烦恼。



6、watch 和 watchEffect 的区别？

watch 和 watchEffect 都是监听器，watchEffect 是一个副作用函数。它们之间的区别有：

1. watch 需要传入监听的数据源，而 watchEffect 可以自动手机数据源作为依赖。
2. watch 可以访问到改变之前和之后的值，watchEffect 只能获取改变后的值。
3. watch 运行的时候不会立即执行，值改变后才会执行，而 watchEffect 运行后可立即执行。这一点可以通过 watch 的配置项 immediate 改变。

7、vue2 如何升级到 vue3 ?

如果是把之前的 vue2 项目升级到 vue3，先卸载旧版本的 vue-cli，安装最新版本。安装完成之后，检查 vue 的版本。然后需要注意，把项目中 vue3 发生改变或被废弃的特性需要进行修改。如：

- \$children 被 vue3 移除，使用 \$children 的需要替换为 \$ref。
- filters 被移除，更改为 computed。
- \$destroy 被移除，需要删除掉。
- 插槽的新变化。
- Vuex 使用发生改变。
- vue-router 使用发生改变等等。

可以自行在官网查看升级指南。地址如图



8、v-if 和 v-for 的优先级哪个高?

在 vue2 中 v-for 的优先级更高，但是在 vue3 中优先级改变了。v-if 的优先级更高。

在 vue2 中 v-for 和 v-if 同时出现时，可放在一个标签内，如下写法：

```
1 <div v-for="item in ss" v-if="item.show" :key="item.id">
2   {{ item.show }}</div>data() {
3   return{
4     ss:[
5       { id: 1, show: true, name: '1' },
6       { id: 2, show: false, name: '2' },
7       { id: 3, show: true, name: '3' },
8     ]
9   }
```

在 vue3 中这样写会报错，就是因为 v-if 的优先级更高，所以 item.show 是未定义报错了。

9、script setup 是干啥的？

script setup 是 vue3 的语法糖，简化了组合式 API 的写法，并且运行性能更好。使用 script setup 语法糖的特点：

- 属性和方法无需返回，可以直接使用。
- 引入组件的时候，会自动注册，无需通过 components 手动注册。
- 使用 defineProps 接收父组件传递的值。
- useAttrs 获取属性，useSlots 获取插槽，defineEmits 获取自定义事件。
- 默认不会对外暴露任何属性，如果有需要可使用 defineExpose 。

10、Vue3 性能提升主要体现在哪几方面？

- 1、响应式性能提升
- 2、编译优化
- 3、源码体积的优化