

一. Vue 核心小知识点

1、vue 中 key 值的作用

key 的特殊属性主要用在 Vue 的虚拟 DOM 算法，在新旧 nodes 对比时辨识 VNodes。如果不使用 key，Vue 会使用一种最大限度减少动态元素并且尽可能的尝试修复/再利用相同类型元素的算法。使用 key，它会基于 key 的变化重新排列元素顺序，并且会移除 key 不存在的元素。

有相同父元素的子元素必须有独特的 key。重复的 key 会造成渲染错误。

最常见的用例是结合 v-for:

```
1 <ul>
2   <li v-for="item in
3     items" :key="item.id">...</li>
   </ul>
```

它也可以用于强制替换元素/组件而不是重复使用它。当你遇到如下场景时它可能会很有用:

完整地触发组件的生命周期钩子

触发过渡

```
1 <transition>
2   <span :key="text">{{ text }}</span>
3 </transition>
```

当 text 发生改变时， 会随时被更新，因此会触发过渡。

2、vue 中子组件调用父组件的方法

子组件调用父组件的方法可以使用 this.\$emit()

3、vue 等单页面应用及其优缺点

优点：

- 1、具有桌面应用的即时性、网站的可移植性和可访问性。
- 2、用户体验好、快，内容的改变不需要重新加载整个页面。
- 3、基于上面一点，SPA 相对对服务器压力小。
- 4、良好的前后端分离。SPA 和 RESTful 架构一起使用，后端不再负责模板渲染、输出页面工作，web 前端和各种移动终端地位对等，后端 API 通用化。

5、同一套后端程序代码，不用修改就可以用于 Web 界面、手机、平板等多种客户端；

缺点：

- 1、不利于 SEO。（如果你看中 SEO，那就不应该在页面上使用 JavaScript，你应该使用网站而不是 Web 应用）
 - 2、初次加载耗时相对增多。
 - 3、导航不可用，如果一定要导航需要自行实现前进、后退。
-

二. v-show 和 v-if 指令的共同点和不同点？

v-show 指令是通过修改元素的 displayCSS 属性让其显示或者隐藏

v-if 指令是直接销毁和重建 DOM 达到让元素显示和隐藏的效果（注意：v-if 可以实现组件的重新渲染）

三. 如何让 CSS 只在当前组件中起作用?

将当前组件的 `<style>` 修改为 `<style scoped>`

四. `<keep-alive></keep-alive>` 的作用是什么?

`<keep-alive></keep-alive>` 包裹动态组件时，会缓存不活动的组件实例，主要用于保留组件状态或避免重新渲染。

大白话：比如有一个列表和一个详情，那么用户就会经常执行打开详情=>返回列表=>打开详情…这样的话列表和详情都是一个频率很高的页面，那么就可以对列表组件使用

`<keep-alive></keep-alive>`进行缓存，
这样用户每次返回列表的时候，都能从缓存中快速渲染，而
不是重新渲染

五. Vue 中引入组件的步骤？

1. 采用 ES6 的 `import ... from ...` 语法

或

CommonJS 的 `require()` 方法引入组件

2. 对组件进行注册，代码如下

注册

```
Vue.component('my-component',  
{ template: '<div>A custom component!</div>' })
```

3.使用组件<my-component></my-component>

六. 指令 v-el 的作用是什么?

提供一个在页面上已存在的 DOM 元素作为 Vue 实例的挂载目标.可以是 CSS 选择器,也可以是一个 HTML Element 实例,

七. 在 Vue 中使用插件的步骤

1.

采用 ES6 的 import ... from ... 语法

2.

3.

使用全局方法 `Vue.use(plugin)` 使用插件, 可以传入一个选项对象

```
Vue.use(MyPlugin, { someOption: true })
```

4.

5.

八. 请列举出 3 个 Vue 中常用的生命周期钩子函数?

1.

created: 实例已经创建完成之后调用, 在这一步, 实例已经完成数据观测, 属性和方法的运算, `watch/event` 事件回调. 然而, 挂载阶段还没有开始, `$el` 属性目前还不可见

2.

3.

mounted: el 被新创建的 vm.\$el 替换，并挂载到实例上去之后调用该钩子。如果 root 实例挂载了一个文档内元素，当 mounted 被调用时 vm.\$el 也在文档内。

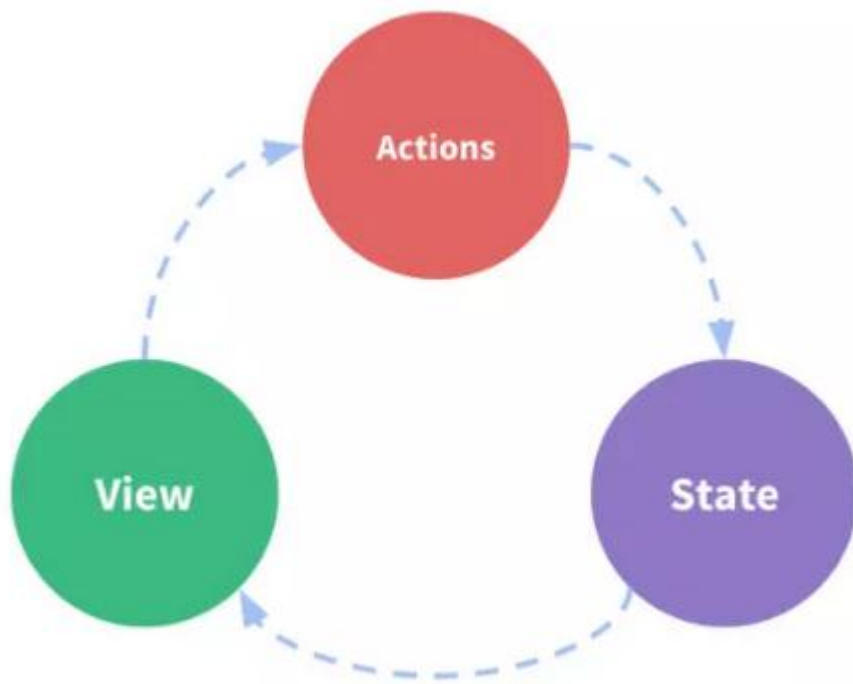
4.

5.

activated::keep-alive 组件激活时调用

6.

九. 请简述下 Vuex 的原理和使用方法



数据单向流动

一个应用可以看作是由上面三部分组成: View, Actions, State, 数据的流动也是从 View => Actions => State => View 以此达到数据的单向流动.

但是项目较大的, 组件嵌套过多的时候, 多组件共享同一个 State 会在数据传递时出现很多问题. Vuex 就是为了解决这些问题而产生的.

Vuex 可以被看作项目中所有组件的数据中心,我们将所有组件中共享的 State 抽离出来,任何组件都可以访问和操作我们的数据中心

Vuex 的组成: 一个实例化的 Vuex.Store 由 state, mutations 和 actions 三个属性组成:

- state 中保存着共有数据
- 改变 state 中的数据有且只有通过 mutations 中的方法,且 mutations 中的方法必须是同步的
- 如果要写异步的方法,需要些在 actions 中,并通过 commit 到 mutations 中进行 state 中数据的更改.

十. vue watch 的高级用法--监听对象的属性变化

1. 监听对象需要深度监听,如下代码可以监听整个 msg 对象的变化

1	watch: {		
2	msg: {		
3	handler(newValue, oldValue) {		
4	console.log(newValue)		
5	},		
6	deep: true		
7	}		
8	}		

2. 监听对象里面某个属性的变化，通过 computed 做中间层实现

1	computed: {	
2	channel() {	
3	return	
4	this.msg.channel	
5	}	
6	},	
7	watch:{	
8	channel(newValue,	
9	oldValue) {	
10		

11

```
console.log('new: %s,  
old: %s', newval,  
oldVal)
```

```
//这里面可  
以执行一旦监听的值  
发生变化你想做的操  
作
```

```
}
```

```
}
```

十一. vue prop 不同数据类型设置默认值

<https://www.cnblogs.com/yalong/p/9771607.html>

十二. vue 实现组件 props 双向绑定

<https://www.cnblogs.com/yalong/p/9759264.html>

1、active-class 是哪个组件的属性？嵌套路由怎么定义？

答：vue-router 模块的 router-link 组件。

嵌套路由请参

考：https://blog.csdn.net/github_26672553/article/details/54861174

2、怎么定义 vue-router 的动态路由？怎么获取传过来的动态参数？

答：在 router 目录下的 index.js 文件中，对 path 属性加上 /:id。使用 router 对象的 params.id 例如：`this.$route.params.id;`

3、vue-router 有哪几种导航钩子？

答：三种，一种是全局导航钩子：

`router.beforeEach(to,from,next)`，作用：跳转前进行判断拦截。第二种：组件内的钩子；第三种：单独路由独享组件
具体参

考：<https://www.jianshu.com/p/aae084e97452>

4、scss 是什么？安装使用的步骤是？有哪几大特性？

答：预处理 css，把 css 当成函数编写，定义变量,嵌套。先装 `css-loader`、`node-loader`、`sass-loader` 等加载器模块，在 `webpack-base.config.js` 配置文件中加多一个拓

展:extension, 再加多一个模块: module 里面 test、loader

4.1、scss 是什么? 在 vue.cli 中的安装使用步骤是? 有哪几大特性?

答: CSS 的预编译。

使用步骤:

第一步: 用 npm 下三个 loader (sass-loader、css-loader、node-sass)

第二步: 在 build 目录找到 webpack.base.config.js, 在那个 extends 属性中加一个拓展.scss

第三步: 还是在同一个文件, 配置一个 module 属性

第四步: 然后在组件的 style 标签加上 lang 属性, 例如:

lang=" scss"

有哪几大特性:

1、可以用变量, 例如 (\$变量名称=值);

2、可以用混合器，例如：

定义了字体的混合器

```
@mixin font-dpr($font-size){  
  $font:$font-size/2;  
  font-size: $font;  
  [data-dpr="2"] & { font-size: $font+2px;}  
  [data-dpr="3"] & { font-size: $font+4px;}  
}
```

使用方法如下

```
.div{  
  
  @include font-dpr(24px);  
  
}
```

3、可以嵌套

5、mint-ui 是什么？怎么使用？说出至少三个组件使用方法？

答：基于 vue 的前端组件库。npm 安装，然后 import 样式和 js，vue.use (mintUi) 全局引入。在单个组件局部引入：
import {Toast} from 'mint-ui'。组件一：Toast('登录成功'); 组件二：mint-header; 组件三：mint-swiper

6、v-model 是什么？怎么使用？ vue 中标签怎么绑定事件？

答：可以实现双向绑定，指令(v-class、v-for、v-if、v-show、v-on)。vue 的 model 层的数据属性。绑定事件：<input @click=doLog() />

7、axios 是什么？怎么使用？描述使用它实现登录功能的流程？

答：请求后台资源的模块。npm install axios -S 装好，然后发送的是跨域，需在配置文件中 config/index.js 进行设置。后台如果是 Tp5 则定义一个资源路由。js 中使用 import 进来，然后.get 或.post。返回在.then 函数中如果成功，失败则是在.catch 函数中

8、axios+tp5 进阶中，调用 axios.post('api/user')是进行的什么操作？ axios.put('api/user/8')呢？

答：跨域，添加用户操作，更新操作。

9、什么是 RESTful API？ 怎么使用？

答：是一个 api 的标准，无状态请求。请求的路由地址是固定的，如果是 tp5 则先路由配置中把资源路由配置好。标准有： .post .put .delete

10、vuex 是什么？ 怎么使用？ 哪种功能场景使用它？

答：vue 框架中状态管理。在 main.js 引入 store，注入。新建了一个目录 store，…… export 。场景有：单页应用中，组件之间的状态。音乐播放、登录状态、加入购物车

11、mvvm 框架是什么？ 它和其它框架（jquery）的区别是什么？ 哪些场景适合？

答：一个 model+view+viewModel 框架，数据模型 model，viewModel 连接两个

区别：vue 数据驱动，通过数据来显示视图层而不是节点操作。

场景：数据操作比较多的场景，更加便捷

12、自定义指令（v-check、v-focus）的方法有哪些？它有哪些钩子函数？还有哪些钩子函数参数？

答：全局定义指令：在 vue 对象的 directive 方法里面有两个参数，一个是指令名称，另外一个函数。组件内定义指令：directives

钩子函数：bind（绑定事件触发）、inserted（节点插入的时候触发）、update（组件内相关更新）

钩子函数参数：el、binding

13、说出至少 4 种 vue 当中的指令和它的用法？

答: v-if: 判断是否隐藏; v-for: 数据循环出来; v-bind:class: 绑定一个属性; v-model: 实现双向绑定

14、vue-router 是什么? 它有哪些组件?

答: vue 用来写路由一个插件。router-link、router-view

15、导航钩子有哪些? 它们有哪些参数?

答: 导航钩子有: a/全局钩子和组件内独享的钩子。

b/beforeRouteEnter、afterEnter、beforeRouterUpdate、beforeRouteLeave

参数: 有 to (去的那个路由)、from (离开的路由)、next (一定要用这个函数才能去到下一个路由, 如果不用就拦截) 最常用就这几种

16、Vue 的双向数据绑定原理是什么?

答: vue.js 是采用数据劫持结合发布者-订阅者模式的方式, 通过 Object.defineProperty() 来劫持各个属性的 setter,

getter，在数据变动时发布消息给订阅者，触发相应的监听回调。]

详细请参

考：<http://www.cnblogs.com/libin-1/p/6893712.html>

具体步骤：

第一步：需要 observe 的数据对象进行递归遍历，包括子属性对象的属性，都加上 setter 和 getter

这样的话，给这个对象的某个值赋值，就会触发 setter，那么就能监听到了数据变化

第二步：compile 解析模板指令，将模板中的变量替换成数据，然后初始化渲染页面视图，并将每个指令对应的节点绑定更新函数，添加监听数据的订阅者，一旦数据有变动，收到通知，更新视图

第三步：Watcher 订阅者是 Observer 和 Compile 之间通信的桥梁，主要做的事情是：

- 1、在自身实例化时往属性订阅器(dep)里面添加自己
- 2、自身必须有一个 update()方法

3、待属性变动 `dep.notice()` 通知时，能调用自身的 `update()` 方法，并触发 `Compile` 中绑定的回调，则功成身退。

第四步：MVVM 作为数据绑定的入口，整合 `Observer`、`Compile` 和 `Watcher` 三者，通过 `Observer` 来监听自己的 `model` 数据变化，通过 `Compile` 来解析编译模板指令，最终利用 `Watcher` 搭起 `Observer` 和 `Compile` 之间的通信桥梁，达到数据变化 -> 视图更新；视图交互变化(input) -> 数据 `model` 变更的双向绑定效果。

ps: 16 题答案同样适合”vue data 是怎么实现的？”此面试题。

17、请详细说下你对 vue 生命周期的理解？

答：总共分为 8 个阶段创建前/后，载入前/后，更新前/后，销毁前/后。

创建前/后：在 `beforeCreated` 阶段，vue 实例的挂载元素 `$el` 和数据对象 `data` 都为 `undefined`，还未初始化。在 `created` 阶段，vue 实例的数据对象 `data` 有了，`$el` 还没有。

载入前/后: 在 beforeMount 阶段, vue 实例的 \$el 和 data 都初始化了,但还是挂载之前为虚拟的 dom 节点,data.message 还未替换。在 mounted 阶段,vue 实例挂载完成,data.message 成功渲染。

更新前/后: 当 data 变化时,会触发 beforeUpdate 和 updated 方法。

销毁前/后: 在执行 destroy 方法后,对 data 的改变不会再触发周期函数,说明此时 vue 实例已经解除了事件监听以及和 dom 的绑定,但是 dom 结构依然存在

18、请说下封装 vue 组件的过程?

答: 首先, 组件可以提升整个项目的开发效率。能够把页面抽象成多个相对独立的模块, 解决了我们传统项目开发: 效率低、难维护、复用性等问题。

然后, 使用 Vue.extend 方法创建一个组件, 然后使用 Vue.component 方法注册组件。子组件需要数据, 可以在 props 中接受定义。而子组件修改好数据后, 想把数据传递

给父组件。可以采用 emit 方法。

19、你是怎么认识 vuex 的？

答：vuex 可以理解作为一种开发模式或框架。比如 PHP 有 thinkphp，java 有 spring 等。

通过状态（数据源）集中管理驱动组件的变化（好比 spring 的 IOC 容器对 bean 进行集中管理）。

应用级的状态集中放在 store 中； 改变状态的方式是提交 mutations，这是个同步的事物； 异步逻辑应该封装在 action 中。

20、vue-loader 是什么？使用它的用途有哪些？

答：解析.vue 文件的一个加载器，跟 template/js/style 转换成 js 模块。

用途：js 可以写 es6、style 样式可以 scss 或 less、template 可以加 jade 等

21、请说出 vue.cli 项目中 src 目录每个文件夹和文件的用法？

答：assets 文件夹是放静态资源；components 是放组件；router 是定义路由相关的配置；view 视图；app.vue 是一个应用主组件；main.js 是入口文件

22、vue.cli 中怎样使用自定义的组件？有遇到过哪些问题吗？

答：第一步：在 components 目录新建你的组件文件（smithButton.vue），script 一定要 export default {

第二步：在需要用的页面（组件）中导入：import smithButton from ‘../components/smithButton.vue’

第三步：注入到 vue 的子组件的 components 属性上面，components:{smithButton}

第四步：在 template 视图 view 中使用，
<smith-button> </smith-button>

问题有：smithButton 命名，使用的时候则 smith-button。

23、聊聊你对 Vue.js 的 template 编译的理解？

答：简而言之，就是先转化成 AST 树，再得到的 render 函数返回 VNode（Vue 的虚拟 DOM 节点）

详情步骤：

首先，通过 compile 编译器把 template 编译成 AST 语法树（abstract syntax tree 即 源代码的抽象语法结构的树状表现形式），compile 是 createCompiler 的返回值，createCompiler 是用以创建编译器的。另外 compile 还负责合并 option。

然后，AST 会经过 generate（将 AST 语法树转化成 render function 字符串的过程）得到 render 函数，render 的返回值是 VNode，VNode 是 Vue 的虚拟 DOM 节点，里面有（标签名、子节点、文本等等）

挑战一下：

- 1、vue 响应式原理？
- 2、vue-router 实现原理？
- 3、为什么要造 vue？与其它框架对比的优势和劣势？
- 4、vue 如何实现父子组件通信，以及非父子组件通信？

5、vuejs 与 angularjs 以及 react 的区别？

6、vuex 是用来做什么的？

7、vue 源码结构

8、webpack 打包原理

9、gulp 构建原理

不定项选择题：

• 下列关于 vue 的组件说法不正确的是 ()

1. 不一定要写 style
2. template 视图里可以写多个 div 容器
3. 父组件给子组件传值需定义 props 属性
4. 子组件与父组件通信需定义 \$emit 属性

• 下面关于 vue 的声明周期说法不正确的是 ()

1. 总共分为 8 个阶段:创建前/后、载入前/后、更新前/后、销毁前/后、
2. updated 和 beforeUpdate 分别是更新完成和更新前

3. 创建后 this 才可以获取属性、mounted 时 \$el 节点才被渲染

4. created 创建后 \$el 就不是 undefined 了

• 下面关于 js 框架说法正确的是 ()

1. Vue 是一个 MVVM 框架

2. Vue 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件

3. Vue 中可以使用 v-for 指令来循环对象

4. 在 input 输入框中我们可以使用 v-model 指令来实现双向数据绑定