1、vue2 和vue3的核心特点 ?

vue2 基于webpac基础搭建。

vue3 中 万物皆组件，一个功能用一个组件。

共同点：都是基于组件化和双向数据绑定。

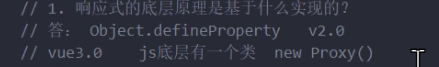
2、说说你对模块化的理解？

3、vue全家桶包含什么东西？

一般来说指的是脚手架vue-cli、路由vue-Router、状态管理模式vuex、Axios、elementUI等ui框架和打包工具webpack

**4、响应式的底层原理是基于什么实现？（vue2.0和 3.0区别）**

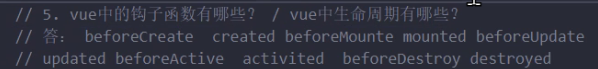
**v2.0基于 Object.**



**5、vue-cli中如何解决跨域？**

**vue.config.js中配置proxy跨域。**

**6、vue的钩子函数有那些？（生命周期）**



**7、**

截图.png

**8、created和 mounted区别？**

created是挂载this，

mounted是渲染了dom

**9、v-bind作用？**

给元素属性绑定动态的值，

v-bind的简写是 ： ，

v-on 简写 @

**10、元素的自定义属性实现原理？**

获取元素的属性，看看是否为v-开头，如果是，利用正则表达式取值并写出对应逻辑。

**11、mvvm中的vm是什么意思？**

vm指的是 数据变视图变，视图变数据变.

**12、v-bind和v-model的区别？**

2 作用于form表单，应用于视图改变数据

1 绑定动态的值，数据变这个值变。

**13、v-if和v-show的区别和应用场景？**

v-if是是否创建元素，

v-show是display：none，如果需要频繁切换用v-show。

**14、v-if和v-for的执行顺序？**

vue2：v-for高于v-if，所以不推荐这么使用。

vue3：v-if高于v-for，

（如果渲染一个list，你需要渲染一个list数据，将数据 处理后在渲染。）

还可以声明一个template模板，在模板上写v-if;

**15、为什么在v-for中要加一个key？**

在底层中，vue通过这个key来找到你的元素，并且来改变这个值的。

底层中，vue列表数据被修改时，是根据这个元素中的key来判断是否需要修改哪一个元素。

**16、组件封装的维度？**

一种是业务组件（只有当前项目中可用），一种是非业务组件（公共组件，这种组件是可以复用到其他项目中）。

17、公司中你用的icon图是什么格式？

png。大小：设计稿的两倍，移动端一般用的是二倍图

18、常用的图片格式 区别是什么？

jpg、jpge、svg、png、gif

前两个是服务端返回的图片类型，因为这两种占用带宽小，且会自动压缩，所以这两种图会比较模糊，因为每转发一次就压缩一次。

png 是ui给的图片样式，在项目中几乎都是png，原因是png不会被压缩，不模糊。

19、图标的命名规范。

icon\_ name.png

20、如何实现父子传参？

在父组件中调用子组件，可以通过在父组件的标签上设置一些自定义的属性，子组件通过props接收。

21、链式调用底层实现原理？

链式调用的实现方案是在每一个方法体中返回一个方法。

22、说一下mvvm的理解？

m：model 模型层，服务端写的代码的层

v：view 视图层 <template></template>

vm：数据双向绑定，export default{ }，vm层

23、说一下对computed和watch的区别？

1 是计算属性，底层用的是Object.defineProperty来实现数据支持，所以不支持异步，必须有当前值，只能设置一个值

2 是监听，是监听当前属性是否发生变化，如果变化就执行逻辑，watch支持异步，可以设置多个值。

24、vue中data的相关理解？

1、开发中，尽可能少定义data中的变量

2、data对象可以返回一个json，也可以返回一个方法，但这个方法中必须返回一个json

3、如果data是一个json，则有可能对其他组件中的data变量造成冲突

4、写项目时，尽可能用方法代替json

25、data声明数据的维度？

越少越好， 因为data对象是有数据双向绑定的，声明越多，性能越差。

26、在前端中，你知道哪些项目优化？

1、cdn

2、在vue实例中尽可能少声明data对象

1. 箭头函数和普通函数的区别?
   1. this 指向不一样。
   2. 箭头函数的this 是指当前函数的父级作用域的this
   3. 普通函数指的是谁调用,this 就是谁
2. 说一下在有子组件的前提下，钩子函数的运行顺序。
   1. 先执行父级的beforeCreate created beforemount
   2. 再执行子组件的beforeCreate created beforemount mounted
   3. 再执行父级的mounted;
3. Xss攻击的理解
   1. Xss 是利用 html的动态生成注入 恶意js 代码。
4. V-html 的安全问题
   1. 非常不安全，几乎不会用，不然会存在xss攻击。
5. 知识点: 大胡子语法{{}}支持表达式（有确切数值）;但是一般情况,表达式应该用computed来替代。
6. vue中的指令 v- v-if v-show v-else-if v-else v-for ；
7. Vue常用的修饰符？(答错了，也完全不影响，除非面试官有毒)
   1. Prevent stop once enter code
8. 用某个 this 值的时候 需要现在data中声明如果不声明会有问题。
9. 你知道的浏览器的内核?
   1. Trident Gecko(fire-fox) Webkit(chrome)
10. 说一下Object.defineProperty的理解
    1. Vue2中用来实现数据双向绑定,
11. 知识点:v-model 的相关只是,v-model适用于form表单中的input(文本框) select(下拉框) textarea(多行文本) radio(单选) checkbox(多选)
    1. V-model 绑定的值,如果该form标签有value那么绑定的就是value值 //<input type=”text” value=’username"> username => value
    2. <select v-model=”user”></select>他的值是option下面的value值,
12. 组件中的命名:
    1. 在声明时应该在-
    2. 在组件中props接受时，驼峰命名法.
13. 全局组件如何注册和的定义?
    1. 第一个参数代表组件名，第二个参数代表组件的内容
    2. App.component(‘compent-name’,componetent-content)
    3. 知识点:组件命名要用横线分割法
14. 动态组件 （非常非常有用）
    1. 主要使用is 把哪个组件弄过来<component is></component>
15. 对于全局组件的理解
    1. 横线分割法永远的神 App.component(‘my-component’,content);
16. 全局组件的应用场景
    1. 因为全局组件中调用组件，所以可以来写无限极联动的效果；
    2. 如果你的需求是数据有多少层级未定义，就可以用全局组件来实现。有点像样式上的递归。
17. 如果没有特殊必要，没有必要使用全局组件。全局组件没有局部组件性能好,如果每一个页面都需要用到的组件，那么应该封装成全局组件。
18. 局部组件的使用发发
    1. Impoet footerItem from [‘@/components/footerItem.vue’](mailto:‘@/components/footerItem.vue’)
    2. Var vm = Vue.create({components:{footerItem,’footer-item’:footerItem}});
19. Es6 又叫ECMAscript2015
20. 2015年以后出的都叫es6
21. 知识点:props 传参时支持json 和数组两种传参方式
    1. App,createApp({props:[‘title’,’icon’]});不校验穿参数类型
    2. App.createApp({props:{title:String//typeString, Boolean,Array,Object,Number,Function,Promise，方式二校验参数类型
    3. }})
22. 在任何场景下,props传入的书信那个，都建议声明类型;
23. 知识点:在子组件中props接收过来的值不能直接修改，如果想要修改，可以把这个赋值给另一个变量
24. Vue 中的单向数据流?
    1. 在vue 中 组件就是使用的单向数据流。
25. 双向数据流？
    1. v-model 双向数据绑定。
26. 知识点，自定义属性命名规范？
    1. 自定义属性 data-
    2. <child-component :user-info=”userinfo”/>不需要变
    3. <child-component :data-user-info=”userinfo”/ 如果需要改变重新复制
    4. Vue.createApp({props:[‘dataUserInfo’],data(){return:{userInfo:{}}}});
27. 子组件的默认复制
    1. Vue.createApp({
       1. Props:{
          1. Username:String,
          2. Data:[String,Number],
          3. }
    2. })
28. 如果组件中的参数是attr 的话他们会默认挂在到该组件的根元素上。组件中的默认事件也会挂在根元素上。
29. 单向数据流?
    1. 在vue中组件都是单向数据流，因为他为了可维护性。
    2. 如果使用双向数据流,会导致代码可维护性降低；
30. Vue2和vue3的区别
    1. Vue2中根元素只能由一个，vue3中可以有多个根元素。
       1. <template><div></div><div></div></template>vue2报错vue3没事。
31. 子组件如何监听父组件时间?Or 说一下如何实现子传父？
    1. 父组件调用子组件，再子组件声明自定义事件，
    2. 子组件通过emits:[‘customEventName’]接受事件名
    3. 通过this.$emit(‘customEventName’) 调用
    4. customName：事件名称
32. 组件中如何实现双向数据流?工作中应该尽可能避免这样的使用。
    1. 父组件调用子组件,在子组件声明v-model属性，这个属性就可以实现双向数据流。
    2. 子组件通过input框的input事件来吧value的值改变了。
    3. 因为value值的改变所以父组件的v-model绑定的值也放生改变这样就起到了双向数据流。
33. 插槽的用法?
    1. 如果有多个插槽必须声明在template上 <template v-slot:default=”scope”></template>
    2. 如果有一个默认插槽可以声明在组件上<zujian v-slot:default=”scope”></zujian>
34. 在一个组件中，如果有且只有一个默认插槽可以缩写成为v-slot;
    1. Eg:<child-component v-slot:default=”scope”></child-component>(标准写法)
    2. Eg:<child-component v-slot=”scope”></child-component>(简写写法)
35. V-slot 可以简写成#
    1. V-slot:header == #header
    2. 在uniapp中大部分使用#slotName,
    3. uniapp是有vue的语法打包成小程序的一个技术栈,
36. 爷孙传参相关知识。
    1. 层级过多