1. 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

1. Vue-router有哪几种导航钩子？

答：三种

全局导航钩子：router.berforeEach(to,from,next) 作用：跳转进行判断拦截

组件内的钩子，单独路由独享组件。

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

1. scss是什么？在vue.cli中的安装使用的步骤是？有哪几大特性？

答：

预处理css，把css当前函数编写，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：extenstion，在加多一个模块；module里面test、loader。也可以直接说是css的预编译

①用npm 下三个loader(sass-loader、css-loader、node-sass)

②在build目录找到webpack.base.config.js，在那个extends属性中添加一个拓展 .scss

③还是在同一个文件，配置一个module属性

④在组件的style标签加上lang属性：ex； lang=’scss’

特性：

①可以用变量， $变量名称=值

②可以用混合器 例如()

③可以嵌套

1. 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

1. 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，通过指令( v-class、v-for、v-if、v-show、v-on )

Vue的model层的data属性，绑定事件：<input @click=函数名() />

1. 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，需再配置文件中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。

1. 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。

1. 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

1. 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。新建一个目录store，

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。

1. mvvm框架是什么？它和其他框架(jquery)的区别是什么？哪些场景适合？

答：

一个model+view+viewModel的框架，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。

1. 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，另外一个是函数，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

1. 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 循环遍历数据 v-bind:class 绑定一个属性 v-model 实现双向数据绑定

1. Vue-router是什么？它有哪些组件？

答：

Vue-router用来写路由一个插件，插件：router-link、router-view。

1. 导航钩子有哪些？它们有哪些参数？

答：

全局钩子和组件内独享的钩子；beforeRouterEnter、afterEnter、beforeRouterUpdate、beforeRouterLeave

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) 最常用的就这几种。

16、vue的双向数据绑定原理是什么？

答：

Vue.js是采用数据结构劫持结合发布者-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。

具体步骤：

①需要observe的数据对象进行递归遍历，包括子对象的属性，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，那么就监听到了数据变化

②compile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。

③Watcher订阅者是Observe和Compile之间通信的桥梁，主要做的事情是：

在自身实例化时往属性订阅器(dep)里面添加自己

自身必须有一个update()方法

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。

④MVVM作为数据绑定的入口，整合Observe、Compile和Watcher三者，通过observe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observe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。

1. 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beforeCreate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。

1. 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，在使用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。

1. 你是怎么认识vuex的？

答：

Vuex可以理解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。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。

1. 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。

1. 请说出vue.cli项目中src目录每个文件夹和文件的用法？

答：

assets文件夹是方静态资源；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。

1. Vue.cli中怎样使用自定义的组件？有遇到过哪些问题吗？

答：

①在components目录新建你的组件文件( 文件名.vue ) script一定要export default{}

②在需要用的页面( 组件 )中导入 import 文件名 from ‘路径’；

③注入到vue的子组件的components属性上面，components:{文件名}

④在template视图view中使用。

问题有；命名是使用驼峰命名法，在使用时要遵循烤串式命名法使用。

1. 聊聊你对vue.js的template编译的理解？

答：

简而言之，就是先转化ast树，在得到的rnder函数返回VNode( vue的虚拟DOM节点 )

通过compile编译器把template编译成AST语法树( abstract syntax tree ) compile是createCompile的返回值，caeateCompile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 )

1. vuex是什么？为什么使用vuex?

答：

Vuex类似Redux的状态管理器，用来管理vue的所有组件状态。

当你打算开发大型单页应用( spa )，会出现多个视图组件，依赖同一个状态，来自不同视图的行为需要变更同一个状态。

1. Vue与angular的区别？

答：

Angular所有的数据和方法都是挂载在$scope上，而vue的数据和方法都是挂载在vue上，只是数据挂载在vue的data，方法挂载在vue的methods上，vue的代码风格更加优雅，json格式书写代码，vue.js有更好的性能，并且非常容易优化。Angular当watcher越来越多时会变得越来越慢，因为作用域内的每一次变化，所有watcher都要重新计算。

Vue、react覆盖中小型，大型项目。Angular一般用于大型项目( 因为比较厚重 )

1. Vue为什么不直接操作DOM？

答：

因为操作DOM对象后，会触发一些浏览器行为，比如布局( layout )和绘制( paint )。Paint是一个耗时的过程，然而layout是一个更耗时的过程，我们无法确定layout一定是自上而下或是自下而上进行的，甚至一次layout会牵涉到整个文档布局的重新计算。浏览器的layout是lazy的，也就是说：在js脚本执行时，是不会去更新DOM的任何对DOM的修改都会被暂存在一个队列中，在当前js的执行上下文完成执行后，会根据这个队列中的修改，进行一次layout。

1. 你怎么理解vue是一个渐进式的框架？

答：

我觉得渐进式就是不必一开始就用vue所有的全家桶，可以根据场景。按需使用想要的插件。也可以说使用vue不需要太多的需求。

1. Vue声明组件的state是用data方法，那为什么data是通过一个function来返回一个对象，而不是直接写一个对象呢？

答：

从语法上说，如果不用function返回就会出现语法错误导致编译不通过，从原理上的话，大概就是组件可以被多次创建，如果不使用function就会使所有调用该组件的页面共用同一个数据域，这样就失去了组件的概念了。

1. 说下vue组件之间的通信？

答：

父传子：this.$refs.xxx 子传父：this.$parent.xxx

还可以通过$emit方法发出一个消息，然后$on接受这个消息

非父子组件间通信：vue有提供vuex，以状态共享方式来实现通信，对于这一点，应该注意考虑平衡，从整体设计角度去考量，确保引入它的必要。

1. Vue中mixin与extend区别？

答：

mixin是全局混合对象,会影响到所有之后创建的vue实例,而vue.extend是对单个实例进行扩展。

同名钩子函数( bind、inserted、uddate、componentUpdate、unbind )将混合为一个数组，因此都将被调用，混合对象的钩子将在组件自身钩子之前调用。

methods、components、directives将被混为同一个对象，两个对象的键名( 方法名、属性名 )冲突时，取组件(而非mixin)对象的键值对。

1. 为什么要选vue？与其他框架对比的优势和劣势？

答：

①方便阅读的中文文档

②容易上手

③速度快体积小。

④基于组件化的开发方式

⑤提高了代码的可读性和可维护性。

优势：

Vuejs吸收了react和angular的优点，强调了react组价化的概念，吸收angular灵活的指令和页面操作的方法。

1. Vue路由，hash history是怎么监听的？

答：

hash: 即地址栏URL中的#符号，hash虽然出现URL中，但不会被包含在HTTP请求中，对后端完全没有影响，因此改变hash不会重新加载页面。

History：利用了HTML5 History Interfince中新增的pushState()和replaceState()方法。

1. Vue与react的区别？

答：

都是组件方式开发，组件实例之间可以嵌套，提供了钩子函数，都不内置列数ajax，route等功能到核心包，而是以插件的方式加载，在组件开发中都支持mixins的特性。