

# JAVA学科ZZJ241111班级JavaWebV12阶段1月第4周周考考题-研发3

1.题目:什么叫做回调地狱? 如何解决回调地狱? (总分10)

在使用JavaScript时, 为了实现某些逻辑经常会写出层层嵌套的回调函数, 如果嵌套过多, 会极大影响代码可读性和逻辑, 这种情况也被成为回调地狱。解决回调地狱有很多方法, 比如: Promise 对象、Generator 函数、async 函数

2.题目:说说你对ES6里的let和const关键字的理解? (总分6)

(1)ES6中的let关键字, 与var类似, 用于声明一个变量.(2分)

(2)ES6中的const关键字定义一个常量.(2分)

特点: (2分)

const 定义的变量与 let 变量类似, 但不能重新赋值

3.题目:说说你对ES6里箭头函数的理解 (总分10)

ES6中允许使用箭头=>来定义箭头函数, 语法简洁。 (2分)

特点: (每项2分)

(1) 箭头函数继承而来的this指向永远不变

(2) call()/apply()/bind()无法改变箭头函数中this的指向

(3) 箭头函数不能作为构造函数使用

(4) 箭头函数没有自己的arguments

(5) 箭头函数没有原型prototype

4.题目:计算属性computed和普通方法的区别的区别? (总分10)

计算属性是基于它的响应式依赖进行缓存的, 只在响应式依赖发生改变的时候计算属性才会重新调用函数求值, 而普通函数只要引起页面的变换, 所有函数都会被触发, 效率很低

5.题目:说说你对computed和watch的理解? (总分10)

概念:

(1) computed: 计算属性, 默认有缓存, 如果被依赖的属性没有发生变化, 就不会重新计算。

(2) watch: 侦听器, 当数据变化的时候, 执行一个函数。

区别:

computed:

(1) 是用来计算出一个值, 这个值我们调用的时候不需要用括号

(2) 如果被依赖的属性没有发生变化, 就不会重新计算 (缓存) 。

watch:

(1) 当数据变化的时候, 执行一个函数

两个选项

(1) immediate: 表示在第一次渲染的时候, 是否调用函数

(2) deep: 如果监听一个对象, 是否要监听对象里面属性的变换

6.题目:请说出vue.cli项目中src目录中assets, components, router, view文件夹以及, main.js文件的作用? (总分10)

(1) assets文件夹是放静态资源; (2分)

(2) components是放组件; (2分)

(3) router是定义路由相关的配置; (2分)

(4) view视图; app.vue是一个应用主组件; (2分)

(5) main.js是入口文件(2分)

7.题目:Vue插槽的作用?插槽的类型? (总分10)

插槽的作用:在对应的组件内部利用slot插槽, 可以接收组件外部传入的内容插槽的使用让组件的布局更加灵活

类型:

(1) 具名插槽

(2) 作用域插槽

8.题目:vuex中有几个核心属性, 分别是什么? 作用分别是什么? (总分10)

VueX 是一个专门为 Vue.js 应用设计的状态管理架构, 统一管理和维护各个vue组件的可变化状态(你可以理解成 vue 组件里的某些 data )。 (5分)

Vue有五个核心概念, 分别是: (每一个1分)

state => 基本数据

getters => 从基本数据派生的数据

mutations => 提交更改数据的方法, 同步!

actions => 像一个装饰器, 包裹mutations, 使之可以异步。

modules => 模块化Vuex

9.题目:vue3中ref函数和reactive函数的异同点? (总分10)

相同点: 都是用来定义响应式数据的方法 (4分)

不同点: (每条2分)

ref定义一个响应式的数据, reactive定义一个对象类型的响应式数据 (基本类型不要用它, 要用ref函数)

ref用来定义基本数据类型, reactive定义对象类型数据

ref对应的响应式数据需要通过.value进行数据读写。reactive不需要。

10.题目:说说你对ES6里promise的理解? (总分14)

概念:

Promise 是异步编程的一种解决方案,所谓Promise, 简单说就是一个容器, 里面保存着某个未来才会结束的事件 (通常是一个异步操作) 的结果。从语法上说, Promise 是一个对象, 从它可以获取异步操作的消息。Promise 提供统一的 API, 各种异步操作都可以用同样的方法进行处理。 (5分)

优点: (3分)

避免了回调地狱 (函数作为参数层层嵌套, 难以阅读的现象);

缺点: (6分, 每项2分)

(1) 取消Promise, 一旦新建它就会立即执行, 无法中途取消。

(2) 如果不设置回调函数, Promise内部抛出的错误, 不会反应到外部。

(3) 当处于pending状态时, 无法得知目前进展到哪一个阶段 (刚刚开始还是即将完成)

