

1、【单选题】

| 得分：2分

下列关于for of说法正确的是: ()

- ☐ A.for((value,i) of 数组), 变量value可获得每个元素的值, 变量i可获得每个元素的下标位置
- ☐ B.for(var i of 数组), 变量i可获得每个元素的下标位置
- ☐ C.for of 的作用和for in完全一样
- ☒ D.for(var value of 数组), 变量value可获得每个元素的值

【正确答案】 D

【答题时间】 2021-11-25 08:14:50

【答案解析】 for of只能获得每个元素的值, 无法获得每个元素的位置所以A和B都是错的。只有C正确for of专门遍历数字下表的数组和类数组对象, for in专门遍历自定义下标的关联数组和对象。所以for of 和for in完全不同。所以D错误。【审核人】魏丽君, 【录题人】魏丽君

2、【单选题】

| 得分：2分

下列关于this说法错误的是 ()

- ☐ A.Student.prototype.intr=function(){ ... this.sname ...} 中this->将来调用这个函数的某个子对象
- ☐ B.obj.fun() fun()中的this->.前的obj对象
- ☐ C.fun() 或 (function(){ ... })() 中的this->window
- ☒ D.new Fun() Fun()中的this->window

【正确答案】 D

【答题时间】 2021-11-25 08:14:50

【答案解析】 总结: new做了几件事目前为止this的用法: 4种: 1. obj.fun() fun()中的this->.前的obj对象——所以A正确, 不选2. new Fun() Fun()中的this->new创建的新对象——所以B错误, 选3. fun() 或 (function(){ ... })() 中的this->window——所以C正确, 不选4. Student.prototype.intr=function(){ ... this.sname ...} 中this->将来调用这个函数的某个子对象, 比如lilei或hmm所以D正确, 不选【审核人】魏丽君, 【录题人】魏丽君

3、【单选题】

| 得分：2分

下列关于VUE中绑定class描述错误的是: ()

- ☐ A.<元素 class=" class名n" :class="{ class名1, class名2, class名3 }">
data:{
 Class名1: true,
 Class名2: false,
 Class名3: true
}
- ☐ B.<元素 :class="{ class名1, class名2, class名3 }">
data:{
 Class名1: true,
 Class名2: false,
 Class名3: true
}

- ☒ C. <元素 class="{ class名1, class名2, class名3 }">
 data:{
 变量:{
 Class名1: true,
 Class名2: false,
 Class名3: true
 }
 }
- ☐ D. <元素 :class="变量">
 data:{
 变量:{
 Class名1: true,
 Class名2: false,
 Class名3: true
 }
 }

【正确答案】C

【答题时间】2021-11-25 08:14:50

【答案解析】用对象语法来绑定class。如何：2种：1). <元素 :class="变量">data:{ 变量:{ class名1: true, class名2: false, class名3: true }}所以，A对2). 简写：用匿名对象语法来绑定class <元素 :class="{ class名1, class名2, class名3 }">data:{ class名1: true, class名2: false, class名3: true}所以，C对3). 固定不变的class和动态变化的class，可以并存固定不变的class名们，放在不带:的class中:动态变化的class名们，放在带:的class中<元素 class="固定不变的class名们" :class="{ class名1, class名2, class名3 }">data:{ class名1: true, class名2: false, class名3: true}所以，D对4). B错在张冠李戴。把A和C两种情况混搭了。【审核人】魏丽君，【录题人】魏丽君

4、【单选题】

| 得分：0分

下列关于vuex中各个组成部分描述不正确的是（）

- ☐ A.向服务器端发送异步ajax请求的函数，应该放在actions中
- ☒ B.修改state中变量的函数，应该定义在mutations中
- ☐ C.多个组件需要共用的变量，应该保存在state对象中
- ☐ D.向服务器端发送异步ajax请求的函数，应该放在mutations中

【正确答案】D

【答题时间】2021-11-25 08:14:50

【答案解析】多个组件需要共用的变量，应该保存在state对象中——所以A正确，不选修改state中变量的函数，应该定义在mutations中——所以B正确，不选向服务器端发送异步ajax请求的函数，应该放在actions中——所以C错误，应该选；D正确，不选。【审核人】魏丽君，【录题人】魏丽君

5、【单选题】

| 得分：2分

下列关于TypeScript中class写法描述正确的是（）

- ☒ A.class 类型名{
 属性1: 数据类型=初始值
 属性2: 数据类型=初始值
 ...
 //构造函数必须定义数据类型
 constructor(形参1:数据类型, 形参2:数据类型,...){
 this.属性1=形参1;
 this.属性2=形参2;
 ...
 }
 /*方法定义*/
- ☐ B.class 类型名{
 属性1
 属性2
 ...
 //构造函数必须定义数据类型

- ```

 constructor(形参1, 形参2,...){
 this.属性1=形参1;
 this.属性2=形参2;
 ...
 }
 /*方法定义 */

```
- ☐ C.class 类型名{  
属性1  
属性2  
...  
//构造函数必须定义数据类型  
function(形参1, 形参2,...){  
this.属性1=形参1;  
this.属性2=形参2;  
...  
}  
/\*方法定义 \*/
- ☐ D.class 类型名{  
//构造函数必须定义数据类型  
constructor(形参1:数据类型, 形参2:数据类型,...){  
this.属性1=形参1;  
this.属性2=形参2;  
...  
}  
/\*方法定义 \*/

【正确答案】 A

【答题时间】 2021-11-25 08:14:50

【答案解析】 class 类型名{ //必须提前声明属性, 才能使用  
类型 constructor(形参1:数据类型, 形参2:数据类型,...){  
——所以, 只有B正确【审核人】魏丽君, 【录题人】魏丽君

属性1: 数据类型=初始值      属性2: 数据类型=初始值      ...      //构造函数必须定义数据类型  
this.属性1=形参1;      this.属性2=形参2;      ...      }      /\*方法定义...\*/      }

## 6、【单选题】

| 得分： 2分

下列关于Vue中nextTick()函数描述错误的时 ()

- ☐ A.今后只要在mounted()之外的地方编写原生DOM操作, 又不想被vue挂载过程覆盖, 都可用nextTick()
- ☐ B.nextTick定义方式:  
this.nextTick()=>{ ...原生DOM操作...})
- ☐ C.nextTick()中包含的代码, 注定只能在当前组件的生命周期结束后, 才自动执行
- ☒ D.nextTick()中包含的代码, 注定只能在当前组件的生命周期开始前提前触发

【正确答案】 D

【答题时间】 2021-11-25 08:14:50

【答案解析】 nextTick()中包含的代码, 注定只能在当前组件的生命周期结束后, 才自动执行——所以A正确, 不选, B错误, 应该选今后只要在mounted()之外的地方编写原生DOM操作, 又不想被vue挂载过程覆盖, 都可用nextTick()——所以, C正确, 不选。nextTick定义方式: this.nextTick()=>{ ...原生DOM操作...})。——所以, D正确, 不选【审核人】魏丽君, 【录题人】魏丽君

## 7、【单选题】

| 得分： 0分

下列关于VUE3中watch说法不正确的是 ()

- ☐ A.要想用watch必须先在顶部用import从vue中解构出watch函数。
- ☐ B.在setup()函数外部平级, 添加以下代码:  
watch(变量名,(newVal, oldVal)=>{  
//当变量发生变化时, 要执行的操作。  
})

- ☒ C.在setup()函数内部, 添加以下代码:  
watch(变量名,(newVal, oldVal)=>{  
//当变量发生变化时, 要执行的操作。  
})
- ☐ D.只要希望变量值发生变化时, 自动执行一项任务时, 都用监视函数

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】今后, 只要希望变量值发生变化时, 自动执行一项任务时, 都用监视函数——D正确, 不选import { defineComponent, ..., watch } from "vue" ——A正确export default defineComponent({ setup(){ var 变量=ref(值); watch(要监控的变量,(newVal, oldVal)=>{ ——B错误, 该选; C正确, 不该选 console.log('变量变了, 新值为\${newVal}, 旧值为\${oldVal}') //变量变化时要执行的操作 }) return { 变量 } }) 【审核人】魏丽君, 【录题人】魏丽君

## 8、【单选题】

得分: 2分

下列关于Vue2脚手架结构描述正确的是 ()

- ☒ A.全局组件及其他不足以成为页面的子组件, 应该放在src/components文件夹下
- ☐ B.页面组件应该放在src/components文件夹下
- ☐ C.路由器对象保存在src/router.js文件中
- ☐ D.配置彻底懒加载和http-proxy代理方式跨域, 应该在src/vue.config.js中

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】Vue2脚手架文件夹结构:页面组件应该放在src/views文件夹下——所以B错误路由字典和路由器对象应该放在src/router/index.js文件中——所以C错误不足以构成一个完整页面的子组件或全局组件应该放在src/components文件夹下——所以A正确http-proxy代理方式跨域和彻底懒加载, 应该在脚手架项目根目录, src文件夹外部平级的vue.config.js文件中配置——所以D错误 【审核人】魏丽君, 【录题人】魏丽君

## 9、【单选题】

得分: 2分

关于修改元素内容描述错误的是: ()

- ☒ A.获取或修改原始HTML内容: 元素.html
- ☐ B.获取或修改表单元素的值: 表单元素.value
- ☐ C.获取或修改纯文本内容: 元素.textContent
- ☐ D.获取或修改原始HTML内容: 元素.innerHTML

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】总结修改内容3种: a. 获取或修改原始HTML内容: 元素.innerHTML ——所以A错误, 选A。C正确, 不选b. 获取或修改纯文本内容: 元素.textContent ——所以B正确, 不选c. 获取或修改表单元素的值: 表单元素.value ——所以D正确, 不选 【审核人】魏丽君, 【录题人】魏丽君

## 10、【单选题】

得分: 0分

设置对象eric中的eid属性只读, 下列写法正确的是 ()

- ☐ A.Object.defineProperty(eric, "eid", false)
- ☒ B.Object.defineProperty(eric, eid, { writable:false })

- ☐ C.Object.defineProperty(eric, "eid", { writable:false })
- ☐ D.Object.defineProperties(eric, "eid", { writable:false })

【正确答案】C

【答题时间】2021-11-25 08:14:50

【答案解析】总结：1). 只修改一个属性的开关: i. Object.defineProperty(对象, "属性名", {开关: true或false,... : ... })——所以C正确, 其余都不对。2). 同时修改一个对象的多个属性的开关:i. Object.defineProperties(对象,{属性名1:{开关: true或false,... : ...},属性名2:{... : ...} })A错在, 没写开关名B错在, 属性名eid必须加""D错在, 使用defineProperties()时, 属性名都要写在{}里边。【审核人】魏丽君, 【录题人】魏丽君

## 11、【单选题】

得分：2分

下列不属于跨域请求的是: ()

- ☐ A.http://www.a.com网站下的网页向http://www.b.com网站发送请求
- ☒ B.http://www.a.com/users/login.html网页向http://www.a.com/index地址发送请求
- ☐ C.https://www.a.com网站下的网页向http://www.b.com网站发送请求
- ☐ D.http://oa.a.com网站下的网页向http://hr.a.com网站发送请求

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】跨域包括: (1). 域名不同: http://www.a.com下的网页 向 http://www.b.com下的服务端发送请求——所以A属于(2). 子级域名不同: http://oa.tedu.com下的网页 向 http://hr.tedu.com下的服务端发送请求——所以D属于(3). 端口号不同: http://localhost:5500下的网页 向 http://localhost:5050下的服务器端发送请求(4). 协议不同: http://12306.cn 下的网页 向 https://12306.cn下的服务端发送请求——所以C属于(5). 即使是本机自己访问自己, 域名下的网页向本机IP下的网页发送请求也算跨域:http://localhost:5050 下的网页 向 http://127.0.0.1:5050 下的服务端发送请求B不属于跨域。B的协议、主机名、端口号完全一样, 只是相对路径不同, 所以不算跨域, 只算站内访问另一个页面而已。【审核人】魏丽君, 【录题人】魏丽君

## 12、【单选题】

得分：2分

关于插槽描述正确的是 ()

- ☒ A.外壳组件中应该包含多个组件相同部分的HTML+CSS+JS, 同时, 可能变化的位置, 可以用<slot>占位
- ☐ B.外壳组件中应该包含多个组件相同部分的HTML+CSS+JS, 同时, 可能变化的位置, 可以用<template>占位
- ☐ C.外壳组件中应该包含多个组件相同部分的HTML+CSS+JS, 同时, 可能变化的位置, 可以用<router-view>占位
- ☐ D.外壳组件中应该包含多个组件相同部分的HTML+CSS+JS, 同时, 可能变化的位置, 可以用<v-slot>占位

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】外壳组件中应该包含多个组件相同部分的HTML+CSS+JS, 同时, 可能变化的位置, 可以用占位。——只有D正确【审核人】魏丽君, 【录题人】魏丽君

## 13、【单选题】

得分：2分

不用查找就可直接获得的元素不包括: ()

- ☐ A.document.documentElement
- ☐ B.document.head

☐ C.document.body☒ D.document.html**【正确答案】D****【答题时间】**2021-11-25 08:14:50**【答案解析】**不需要查找就可直接获得的元素: 4种a. documentb. document.documentElement ——所以A正确, 不选c. document.head ——D正确, 不选, B错误, 选Bd. document.body ——C正确, 不选【审核人】魏丽君, 【录题人】魏丽君

## 14、【单选题】

| 得分: 2分

下列关于arguments对象的描述正确的是: ()

☒ A.每个函数中自带的, 无需自己创建, 就可直接使用☐ B.纯正的数组对象☐ C.下标为自定义名称的字符串下标☐ D.可使用数组类型的API**【正确答案】A****【答题时间】**2021-11-25 08:14:50**【答案解析】**什么是arguments: 1). 每个函数中自带的, 无需自己创建, 就可直接使用——A正确2). 自动接收所有传入函数内实参值的3). 类数组对象: ——错误 i. 什么是: 长得像数组的对象 ii. vs 数组: 相同: 1. 数字下标i, ——D错误 2. length, 3. 可用for循环遍历arguments: { 0: 值1, 1: 值2,..., length: 元素个数 } 不同: 类数组对象不是数组家孩子! 无权使用数组家任何函数! ——C错误【审核人】魏丽君, 【录题人】魏丽君

## 15、【单选题】

| 得分: 2分

下列关于VUE2中组件生命周期描述正确的是 ()

☒ A.组件生命周期包括四个阶段, 依次是: 创建、挂载、更新、销毁☐ B.组件生命周期包括四个阶段, 依次是: 捕获、触发、冒泡、阻止☐ C.组件生命周期包括四个阶段, 依次是: 创建、更新、挂载、销毁☐ D.组件生命周期包括四个阶段, 依次是: 挂载、创建、更新、阻止**【正确答案】A****【答题时间】**2021-11-25 08:14:50**【答案解析】**组件生命周期包括四个阶段, 依次是: 创建、挂载、更新、销毁。所以, 只有B正确【审核人】魏丽君, 【录题人】魏丽君

## 16、【单选题】

| 得分: 2分

下列对jquery新增选择器描述错误的是 ()

☒ A.:first可能选择多个元素☐ B.:nth-child(i), i从1开始☐ C.:first只能选择一个元素☐ D.:first-child可能选择多个元素

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】1). :nth-child(i)是css选择器，i从1开始！——所以A正确:first-child匹配作为其所在父元素下的第一个直接子元素的元素。如果有多个父元素，可找所有父元素下第一个孩子，就有可能返回多个元素——所以B正确基本过滤选择器: i :first 选择所有符合条件的元素中第一个元素 因为一个集合，只有一个头，所以整个集合中的第一个元素也只有一个。所以，C错误，D正确【审核人】魏丽君，【录题人】魏丽君

## 17、【单选题】

| 得分：2分

下列关于Vue2中监视函数定义正确的是（）

☐ A.export default {  
  data(){  
    return { 变量: 值 }  
  },  
  watch:{  
    变量():=>{  
      要执行的操作  
    }  
  }  
}

☒ B.export default {  
  data(){  
    return { 变量: 值 }  
  },  
  watch:{  
    变量(){  
      要执行的操作  
    }  
  }  
}

☐ C.export default {  
  data(){  
    return { 变量: 值 }  
  },  
  watches:{  
    变量():=>{  
      要执行的操作  
    }  
  }  
}

☐ D.export default {  
  data(){  
    return { 变量: 值 }  
  },  
  watches:{  
    变量(){  
      要执行的操作  
    }  
  }  
}

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】Vue2中定义监视函数: export default { data(){ return { 变量: 值 } }, watch:{ 变量(){ 要执行的操作 } }}——所以，只有D正确【审核人】魏丽君，【录题人】魏丽君

## 18、【单选题】

| 得分：0分

下列关于TypeScript模块化开发描述正确的是（）

☐ A.模块中只抛出一个东西:  
  export default { 一个东西 };

其他模块中引入时:  
import { 别名 } from "相对路径"

- ☐ B.模块中只抛出一个东西:  
export default 一个东西;  
其他模块中引入时:  
import { 别名 } from "相对路径"
- ☒ C.模块中抛出多个东西:  
export { 原名1, 原名2, ... };  
其他模块中引入时:  
import { 别名1, 别名2, ... } from "相对路径"
- ☐ D.模块中抛出多个东西:  
export { 原名1, 原名2, ... };  
其他模块中引入时:  
import { 原成员名1, 原成员名2, ... } from "相对路径"

【正确答案】D

【答题时间】2021-11-25 08:14:50

【答案解析】(1). 模块中只抛出一个东西: export default 一个东西 import 别名 from "相对路径" ——所以A和B都错 (2). 模块中同时抛出多个东西: export { 东西1, 东西2, ... } import { 东西1, 东西2, ... } from "相对路径" //{}中的成员名, 必须能够在原export的{}内找到对应成员——所以C错误, D正确  
【审核人】魏丽君, 【录题人】魏丽君

## 19、【单选题】

得分: 2分

下列关于Vuex中三个“传送门”函数用法描述正确的是 ()

- ☐ A....mapMutations(["函数名"]), 应该放在组件对象的computed:{}中
- ☐ B....mapActions(["函数名"]), 应该放在组件对象的computed:{}中
- ☐ C....mapState(["函数名"]), 应该放在组件对象的methods:{}中
- ☒ D....mapState(["函数名"]), 应该放在组件对象的computed:{}中

【正确答案】D

【答题时间】2021-11-25 08:14:50

【答案解析】mapState()传送来的是state中的变量, 但是, 只能获取变量值, 不能修改变量值, 所以mapState()传来的是计算属性, 应该放在computed()中。——所以C错误, D正确mapMutations()和mapActions()传送来的是mutations和actions中的函数, 所以, 应该放在methods中。所以A和B都错误  
【审核人】魏丽君, 【录题人】魏丽君

## 20、【单选题】

得分: 2分

下列关于VUE3新生命周期钩子函数描述不正确的是 ()

- ☒ A.第一个阶段是create阶段。该阶段前后分别触发的生命周期钩子函数为: beforeCreate和created
- ☐ B.第二个阶段是mount阶段。该阶段前后分别触发的生命周期钩子函数为: onBeforeMount和onMounted
- ☐ C.更新阶段前后分别触发的生命周期钩子函数为: onBeforeUpdate和onUpdated
- ☐ D.销毁阶段前后分别触发的生命周期钩子函数为: onBeforeUnmount和onUnmounted

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】新声明周期钩子函数: 背过: beforeCreate -> setup() created -> setup()——所以A不正确, 该选 beforeMount -> onBeforeMount mounted -> onMounted ——所以B对, 不选 beforeUpdate -> onBeforeUpdate updated -> onUpdated——所以C对, 不选 beforeDestroy -> onBeforeUnmount destroyed -> onUnmounted——所以D对, 不选【审核人】魏丽君, 【录题人】魏丽君



## 21、【单选题】

| 得分: 0分

下列关于jquery中事件绑定描述正确的是: ()

- ☐ A.Jq中即使绑定事件处理函数时, 使用有名称的函数, 也可以重复绑定多个事件处理函数!
- ☐ B.\$元素.事件名(处理函数) 底层相当于.onclick=function(){...}
- ☐ C.DOM中如果绑定事件处理函数时, 使用有名称的函数, 则可以重复绑定多个事件处理函数!
- ☒ D.Jq中绑定事件处理函数时, 使用有名称的函数, 则该事件处理函数只能绑定一次

【正确答案】 A

【答题时间】 2021-11-25 08:14:50

【答案解析】 jq中事件绑定: (1). 标准写法: 用on代替addEventListener, 用off代替removeEventListener(2). 简写: 如果事件名属于常用的事件列表中, 可简写为: \$元素.事件名(处理函数)底层相当于addEventListener(), 而不是.onclick=function(){ ... }, 所以A错误差异: (只记现象, 不用深究)(1). DOM中: 如果绑定事件处理函数时, 使用有名称的函数, 则该函数只能绑定一次! ——所以C错误(2). Jq中: 即使绑定事件处理函数时, 使用有名称的函数, 也可以重复绑定多个! 移除时, 也可以都移除! ——所以B错误。D正确。【审核人】魏丽君, 【录题人】魏丽君

## 22、【单选题】

| 得分: 0分

vuex中actions中异步函数内, 调用mutations中函数的方法是 ()

- ☐ A.context.commit("方法名",实参值)
- ☐ B.this.commit("方法名",实参值)
- ☐ C.state.方法名(实参值)
- ☒ D.this.方法名(实参值)

【正确答案】 A

【答题时间】 2021-11-25 08:14:50

【答案解析】 vuex中actions中异步函数内, 调用mutations中函数的方法, 只能用context.commit("方法名", 实参值)所以, 只有A正确, 其余都错误【审核人】魏丽君, 【录题人】魏丽君

## 23、【单选题】

| 得分: 0分

下列关于计算后的样式描述错误的是: ()

- ☐ A.元素.style.css属性方式, 也只能获得元素的内联样式
- ☐ B.只要获取一个元素的完整的css属性都必须获得计算后(computed)的样式
- ☒ C.获得完整的计算后的样式的集合对象: var style=getComputedStyle(元素对象)
- ☐ D.通过getComputedStyle()获得的计算后的样式中所有css属性均可修改

【正确答案】 D

【答题时间】 2021-11-25 08:14:50

【答案解析】 问题: style.css属性方式, 也只能获得元素的内联样式——A正确, 不选解决: 只要获取一个元素的完整的css属性都必须获得计算后(computed)的样式。——B正确, 不选(1). 什么是计算后的样式: 最终应用到这个元素上的所有样式的总和。(F12->Elements->Computed)(2). 如何: 2步: a. 先获得完整的计算后的样式的集合对象: var style=getComputedStyle(元素对象) ——D正确, 不选强调: getComputedStyle()函数是浏览器内置函数, 可直接使用, 不能改名b. 从获得的样式集合对象style中获取计算后的某一个css属性值style.css属性(4). 计算后的样式, 不仅收集这个元素的所有样式到一个对象中集中保存, 而且还会将相对单位(em)以及不是数值的属性值, 换算回绝对单位和数值方式。(5). 所有通过计算后的样式获得css属性都是只读的! 不能修改! 因为计算后的样式来源不确定, 一旦修改, 很可能牵一发而动全身! ——所以C错误【审核人】魏丽君, 【录题人】魏丽君

## 24、【单选题】

| 得分：0分

下列关于TypeScript中变量和数据类型描述错误的是（）

- ☐ A.如果TypeScript中定义变量时，不显式指定数据类型，会报错
- ☒ B.any类型的变量，可以保存任意类型的数据
- ☐ C.如果TypeScript中定义变量时，不显式指定数据类型，则默认以首次赋值的数据类型为变量的数据类型
- ☐ D.TypeScript中定义变量时，可以不显式指定数据类型

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】TypeScript中定义变量时，可以不显式指定数据类型。——所以A正确，不选如果TypeScript中定义变量时，不显式指定数据类型，则默认以首次赋值的数据类型为变量的数据类型——所以B正确，不选；D错误，应该选any类型的变量，可以保存任意类型的数据——C正确，不选【审核人】魏丽君，【录题人】魏丽君

## 25、【单选题】

| 得分：0分

下列关于Vue3事件修饰符描述错误的是（）

- ☐ A.事件.enter 按回车时才执行
- ☒ B.事件.prevent 阻止默认行为
- ☐ C.事件.stop 阻止冒泡
- ☐ D.事件.13 按回车时才执行

【正确答案】D

【答题时间】2021-11-25 08:14:50

【答案解析】事件名.enter，表示按回车时才执行。Vue3中不能使用.13来限制键盘事件了。——所以A错误，该选，D正确，不选事件名.stop 阻止冒泡，等效于e.stopPropagation()——所以B正确，不选事件名.prevent 阻止默认行为，等效于e.preventDefault()——所以C正确，不选【审核人】魏丽君，【录题人】魏丽君

## 26、【单选题】

| 得分：2分

下列关于vuex描述正确的是（）

- ☒ A.组件中，只要希望使用vuex中state中的变量，可以使用mapState()函数来获取。
- ☐ B.组件中，只要希望使用vuex中actions中的函数，可以使用mapMutations()函数来获取。
- ☐ C.组件中，只要希望使用vuex中mutations中的函数，可以使用mapAction()函数来获取。
- ☐ D.组件中，只要希望使用vuex中state中的变量，可以使用mapAction()函数来获取。

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】vuex提供了一套“传送门”函数:mapState()专门负责获取state中的变量——所以A错误，C正确mapMutations()专门负责获取mutation中的函数——所以B错误mapActions()专门负责获取action中的函数——所以D错误【审核人】魏丽君，【录题人】魏丽君

## 27、【单选题】

| 得分：2分

关于reg.exec()函数返回值的描述正确的是 ( )

- ☐ A.arr=[ "content": "敏感词内容", "index": 敏感词的位置i ]
- ☐ B.arr=[ "敏感词1 ", "敏感词2 ", "敏感词3" ,...]
- ☐ C.arr=[ " index ": "敏感词内容", "0": 敏感词的位置i ]
- ☒ D.arr=[ "0": "敏感词内容", "index": 敏感词的位置i ]

【正确答案】 D

【答题时间】 2021-11-25 08:14:50

【答案解析】 reg.exec()函数的返回值: i. 如果找到, 返回一个数组: arr=[ "0": "敏感词内容", "index": 敏感词的位置i ]——只有C正确【审核人】魏丽君, 【录题人】魏丽君

## 28、【单选题】

得分: 2分

下列关于vuex中mutations中方法内, 如何访问state中变量, 描述正确的是 ( )

- ☐ A.this.变量名()
- ☐ B.this.state.变量名
- ☒ C.state.变量名
- ☐ D.this.变量名

【正确答案】 C

【答题时间】 2021-11-25 08:14:50

【答案解析】 下列关于vuex中mutations中方法内, 要想访问state中的变量, 必须使用state.变量名的方式【审核人】魏丽君, 【录题人】魏丽君

## 29、【单选题】

得分: 0分

如果外壳组件中包含多个插槽标签<slot>, 将来在其他组件中使用该外壳时, 以下描述正确的是 ( )

- ☐ A.可能变化的多处HTML片段, 应该放在<template>元素内, 不用加任何特殊属性, <template>就可以按顺序匹配外壳组件中的<slot>标签位置。
- ☒ B.可能变化的多处HTML片段, 应该放在<slot>元素内, 必须加#slot名, <slot>才可根据#slot名匹配外壳组件中同名的<slot>位置
- ☐ C.可能变化的多处HTML片段, 可以省略<template>标签。
- ☐ D.可能变化的多处HTML片段, 应该放在<template>元素内, 必须加#slot名, <template>才可根据#slot名匹配外壳组件中同名的<slot>位置

【正确答案】 D

【答题时间】 2021-11-25 08:14:50

【答案解析】 如果外壳组件包含多段可能变化的区域, 则外壳组件中用<slot name="插槽名">来标记不同的插槽位置。当其他组件使用外壳组件时, 必须用<template #插槽名>将一段HTML片段与外壳组件中的同名<slot>对应。——所以只有B正确【审核人】魏丽君, 【录题人】魏丽君

## 30、【单选题】

得分: 0分

对一个全由数字组成的数组中所有元素值\*2, 直接修改原数组, 下列写法正确的是 ( )

- ☐ A.arr.forEach(function(elem,i,arr){ elem=arr[i]\*2 })

- ☐ B.arr.forEach(function(elem,i,arr){ arr[i]=elem\*2 })
- ☒ C.arr.forEach(function(elem){ return elem\*2 })
- ☐ D.arr.forEach(function(elem){ elem\*=2 })

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】因为按值传递，如果数组中所有元素都是原始类型的数字，则回调函数参数elem，每次获得的仅是数组中每个数字元素值的副本。所以，只要修改elem，都无法直接修改原数组中的元素值。所以A和D错误。B选项forEach不支持return，写了也没用，所以B也错误。C，因为直接拿arr变量中保存的数组对象地址加下表[i]修改原数组中的元素位置，所以C正确。【审核人】魏丽君，【录题人】魏丽君

### 31、【单选题】

得分：2分

常见解决跨域问题的方式描述正确的是（）

- ☐ A.CORS方式跨域，客户端和服务端代码都要修改
- ☐ B.JSONP方式跨域，只需要客户端代码修改，服务器端代码无需修改，即可完成
- ☒ C.http-proxy代理方式跨域，只需要客户端代码修改，服务器端代码无需修改，即可完成
- ☐ D.http-proxy代理方式跨域，需要客户端和服务端代码都要修改

【正确答案】C

【答题时间】2021-11-25 08:14:50

【答案解析】最常用三种解决跨域方式: CORS，只需要服务端修改代码，客户端代码不用修改——所以A错误JSONP，需要客户端和服务端代码都要修改——所以B错误http-proxy，只需要客户端代码修改，服务器端代码不用修改——所以C正确，D错误【审核人】魏丽君，【录题人】魏丽君

### 32、【单选题】

得分：2分

下列关于Vue懒加载中preload和prefetch说法正确的是（）

- ☐ A.prefetch提前下载，并提前渲染到网页中给人看
- ☒ B.preload是提前下载，并提前渲染到网页中给人看
- ☐ C.prefetch下载快慢会影响首页加载速度
- ☐ D.preload下载快慢与首页加载速度无关

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】preload是提前下载，并提前渲染到网页中给人看。Preload下载速度直接影响首屏加载速度——所以A正确，B错误prefetch是只异步下载，但不立刻渲染到网页中给人看。下载快慢不影响首页加载速度。——所以B，C错误。【审核人】魏丽君，【录题人】魏丽君

### 33、【单选题】

得分：2分

下列关于事件委托三步描述错误的是：（）

- ☒ A.事件委托中this依然可获得最初点击的子元素对象
- ☐ B.在父元素上绑定一次处理函数即可
- ☐ C.事件委托中，都要先验证目标元素e.target的特征是否符合要求

- ☐ D.事件委托中不要用this获得当前元素对象。而要换成e.target

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】3步:事件绑定在父元素上, ——所以A正确, 不选this不再指子元素, 而是指父元素——所以B错误, 选解决: 事件委托中不要用this获得当前元素对象。而要换成e.target。——所以C正确, 不选e.target, 是事件对象中自带的, 永久保存最初的目标元素的属性。不随冒泡而改变! 问题: 也可能点在子元素上或其它不想要的子元素上, 也会出发事件解决: 事件委托中, 都要先验证目标元素e.target的特征是否符合要求! 只有符合要求的子元素, 才能继续执行处理函数。如果发现, e.target是不符合要求的元素, 就什么都不干! ——所以D正确, 不选【审核人】魏丽君, 【录题人】魏丽君

### 34、【单选题】

| 得分: 0分

下列关于取消冒泡正确的是 ()

- ☒ A.e.target
- ☐ B.e.cancelBubble()
- ☐ C.e.stopPropagation()
- ☐ D.e.preventDefault()

【正确答案】C

【答题时间】2021-11-25 08:14:50

【答案解析】取消冒泡: e.stopPropagation() 停止蔓延/冒泡——只有A正确B, e.target用来获得正在触发事件的目标元素——与冒泡无关, B不正确。C, 没有这个函数——C不正确D, preventDefault()是阻止元素身上自带的默认行为, 不是阻止冒泡。——D不正确。【审核人】魏丽君, 【录题人】魏丽君

### 35、【单选题】

| 得分: 2分

关于Vue3中界面所需的变量, 描述正确的是 ()

- ☐ A.Vue3中setup()函数中, 要想使用界面所需的变量, 必须加this.变量名
- ☒ B.setup()中定义的变量, 必须放在最后return的{}对象中, 才能被界面识别并使用
- ☐ C.只要再setup()函数中定义的一切变量, 都能在界面中使用, 并受监控
- ☐ D.Vue3中界面所需变量, 必须写在data:{}中

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】Vue3中界面所需变量:setup(){ var 变量名=ref(变量值); ——所以A错误 var 函数名={()=>{ //直接操作变量名.value, 不用加this... ——所以B错误 } return { 变量名, 函数名 } }//必须放在return中的变量和函数, 才能被界面识别。——所以C错误, D正确【审核人】魏丽君, 【录题人】魏丽君

### 36、【单选题】

| 得分: 0分

以下异步延迟懒加载配置方式正确的是()

- ☐ A.const routes=[
- ```
...
{
  path:"/details/:lid",
  component: () => import(
    '@views/Details.vue'
  ),
```

```
    props:true
  ]

☒ B.const routes=[
  ...
  {
    path:"/details/:lid",
    component: () => import(
      /* webpackChunkName: "../views/Details" */
    ),
    props:true
  }
]

☐ C.const routes=[
  ...
  {
    path:"/details/:lid",
    component: () => import(
      /* webpackChunkName: "details" */
      './views/Details.vue'
    ),
    props:true
  }
]

☐ D.const routes=[
  ...
  {
    path:"/details/:lid",
    component(){
      /* webpackChunkName: "details" */
      './views/Details.vue'
    },
    props:true
  }
]
```

【正确答案】 C

【答题时间】 2021-11-25 08:14:50

【答案解析】 异步延迟懒加载的配置方式:router/index.js中开头, 不要import除主页组件之外的其他页面组件。然后, 修改路由字典中:const routes=[... path:"/details/:lid", component: () => import(——所以, 只有B正确 /* webpackChunkName: "details" */ './views/Details.vue'), props:true] 【审核人】 魏丽君, 【录题人】 魏丽君

37、【单选题】

得分: 2分

Vue3核心绑定原理 ()

- ☐ A.访问器属性+虚拟DOM树
- ☒ B.proxy代理对象+虚拟DOM树
- ☐ C.proxy代理对象+真实DOM树
- ☐ D.访问器属性+proxy代理对象

【正确答案】 B

【答题时间】 2021-11-25 08:14:50

【答案解析】 VUE3绑定元素: ES6的proxy对象+虚拟DOM树 ——所有只有C正确 【审核人】 魏丽君, 【录题人】 魏丽君

38、【单选题】

得分: 2分

下列关于<style scoped>描述正确的是 ()

- ☐ A.只要加上scoped属性, 一定不会发生组件间样式冲突
- ☐ B.scoped属性会自动为当前组件的唯一父元素添加随机class名, 来避免组件间样式冲突

- ☐ C.将来避免组件间样式冲突，都必须用scoped属性
- ☒ D.scoped属性会自动为当前组件中每个元素都添加相同随机自定义属性。但是不同组件的随机自定义属性名各不相同，所以，可避免组件间样式冲突

【正确答案】D

【答题时间】2021-11-25 08:14:50

【答案解析】避免组件间样式冲突: 2种方式: 1. <style scoped>。Scoped自动为当前组件中每个元素添加相同的随机自定义属性。但是不同组件的随机自定义属性又各不相同。同时，修改组件中的选择器，自动添加当前组件对应的自定义属性选择器。以此来避免组件间样式冲突——所以B错误，C正确2. 但是scoped无法对外部引入的css文件中的选择器添加自定义属性选择器，所以scoped对于外部引入的css文件，无法防止组件间样式冲突。——所以A错误3. 最好的避免组件间样式冲突的方式，是我们手工为当前组件的唯一父元素添加唯一class名，然后，定义css样式中的选择器时，让当前组件的所有选择器都以这个唯一父元素开头。因为不同组件的唯一父元素的class名各不相同，所以可以避免组件间样式冲突——所以D错误。【审核人】魏丽君，【录题人】魏丽君

39、【单选题】

得分：0分

配置淘宝镜像语句正确的是（）

- ☐ A.npm config set registry http://registry.npm.taobao.org
- ☐ B.npm config set register http://registry.npm.taobao.org
- ☒ C.npm config registry http://registry.npm.taobao.org
- ☐ D.npm config set url http://registry.npm.taobao.org

【正确答案】A

【答题时间】2021-11-25 08:14:50

【答案解析】isNaN检测是否为NaN, 会调用Number函数将undefined转为数值型为NaN【审核人】魏丽君，【录题人】魏丽君

40、【单选题】

得分：0分

\$mount("#app")函数主要功能描述正确的是（）

- ☐ A.主要用于创建Vue类型子对象
- ☐ B.主要用于扫描真实DOM树，创建虚拟DOM树，并将Vue对象中内容挂载到页面中
- ☐ C.主要用于强制刷新页面
- ☒ D.主要用于当地址栏参数值发生变化时，自动更新页面。

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】\$mount()主要用于扫描真实DOM树，创建虚拟DOM树，并将Vue对象中内容挂载到页面中——只有B正确创建Vue类型子对象，vue2用的是new Vue(), vue3用的是createApp(App)。——A错误强制刷新页面，可以用location.reload(true)——C错误地址栏参数发生变化，自动更新页面，自动重新发送请求，用的是watch监视函数——D错误【审核人】魏丽君，【录题人】魏丽君

41、【单选题】

得分：0分

下列关于class中共有属性描述正确的是（）

- ☐ A.直接放在class中的属性，自动就是共有属性
- ☐ B.以上都不对
- ☐ C.在构造函数里this.属性名=属性值，这个属性自动就是共有属性

- ☒ D.使用static关键字标记的静态属性，都是保存在原型对象中的

【正确答案】B

【答题时间】2021-11-25 08:14:50

【答案解析】(1). 直接放在class中的属性，绝不是共有属性，而会成为将来每个子对象的自有属性——所以A错误(2). 构造函数妈妈中定义的属性，会成为将来子对象身体里的自有属性——所以B错误(3). static标记的静态属性，都是保存在构造函数对象身上。只能用构造函数.静态属性方式访问。——所以C错误所以选D【审核人】魏丽君，【录题人】魏丽君

42、【多选题】

| 得分：0分

下列关于TypeScript中接口定义和使用不正确的是（）

- ☒ A.interface I接口名{
 属性名:数据类型=属性值
 ...
 方法名(形参:数据类型, ...){ 函数体 }
}
- ☒ B.class 类型名 implements I接口名{
 属性名:数据类型=值
 ...
 方法名(形参:数据类型, ...):返回值类型{
 方法实现
 }
}
- ☒ C.class 类型名 extends I接口名{
 ...
 constructor(){
 super();
 }
 ...
}
- ☐ D.interface I接口名{
 属性名:数据类型
 ...
 方法名(形参:数据类型, ...):返回值类型;
}

【正确答案】A,C

【答题时间】2021-11-25 08:14:50

【答案解析】当希望开发人员必须按照设计人员的要求实现类型定义时，就要用接口来约束。a. 先定义接口： interface I接口名{ 属性名:数据类型 ... 方法名(形参:数据类型, ...):返回值类型; }——所以A正确，不选。B错误，该选 b. 定义class实现接口的要求： class 类型名 implements I接口名{ 属性名:数据类型=值 ... 方法名(形参:数据类型, ...):返回值类型{ 方法实现 } }——所以C正确，不选，D错误，该选【审核人】魏丽君，【录题人】魏丽君

43、【多选题】

| 得分：0分

关于Vue3中计算属性描述正确的是（）

- ☒ A.将一个函数变为计算属性:
computed("计算属性名", function(oldVal){
 inserted(oldVal){
 return 计算后的新值
 }
})
- ☒ B.将一个函数变为计算属性:
计算属性名=computed(()=>{
 return 计算后的新值
})

- ☐ C.只要希望根据一个现有数据，计算获得新结果值，都可用计算属性
- ☐ D.只要希望将一个不能直接给人看的原始数据，加工后再显示给人看，就可用计算属性。

【正确答案】B,C,D

【答题时间】2021-11-25 08:14:50

【答案解析】Vue中，只要希望根据一个现有数据，计算获得新结果值，都可用计算属性——所以B正确但是，Vue3中没有过滤器了，而是由计算属性同时承担过滤器的任务——所以A正确Vue3中如何定义计算属性：`import { defineProperty, ..., computed } from 'vue' export default defineComponent({ setup() { 计算属性名=computed(()=>{ //计算过程 return 返回值 }) })`——所以C错误，D正确【审核人】魏丽君，【录题人】魏丽君

44、【多选题】

得分：0分

下列关于在vue3中如何发送ajax请求，说法正确的是（）

- ☐ A.在main.ts中配置好axios.defaults对象的baseURL，不必放入Vue的原型对象，其他组件就可通过import引入main.ts中配置好的axios对象，直接使用。
- ☒ B.使用Vue.prototype.axios=axios将axios对象放入vue类型的原型对象中
- ☒ C.在onMounted()生命周期钩子函数中，可使用this.axios.get()方式想服务器端发送get请求
- ☐ D.在onMounted()生命周期钩子函数中，直接使用axios.get()方式想服务器端发送get请求

【正确答案】A,D

【答题时间】2021-11-25 08:14:50

【答案解析】Vue3中axios使用步骤：1. main.ts中引入axios:import axios from "axios" 配置基础路径: axios.defaults.baseURL="服务器端基础路径" //因为vue3中已经不能直接使用Vue关键字，所以，不能将axios放入Vue类型原型对象中——所以A错误，B正确2. 其他组件中: import axios from "axios" //单例模式，导致这里import引入的就是main.ts中的同一个axios对象 export default defineComponent({ setup() { onMounted() { axios.get(...).then(result=>{ ... }) //因为没有放在原型对象中，只是用import引入的一个普通对象，所以不用加this。所以C错误，D正确 } }) } }) 【审核人】魏丽君，【录题人】魏丽君

45、【多选题】

得分：0分

下列关于表单元素过滤器描述正确的是：（）

- ☒ A.:text 专门选择 <input type="text">的元素，但是:text不能单独使用，必须前加input:text才能选择文本框元素
- ☒ B.input 只是一个元素选择器，只能选择input元素而已
- ☐ C.:hidden 只能选择<input type=" hidden" >的隐藏域元素，不能选择其他元素
- ☐ D.:input 可选择所有表单元素(input select textarea button)

【正确答案】B,D

【答题时间】2021-11-25 08:14:50

【答案解析】:input 可选择所有表单元素(input select textarea button)——所以A正确而input 只是一个元素选择器，只能选择input元素而已——所以C错误b. 其实每种input的type属性都对应着一种专门的选择器，比如：1). :text 专门选择 <input type="text">的元素——所以B错误。:text单独就可匹配文本框。2). :hidden: 即可选择<input type=" hidden" >的元素又可选择display:none的元素) ——所以D错误【审核人】魏丽君，【录题人】魏丽君

46、【多选题】

得分：0分

关于Vue3中自定义指令定义正确的是（）

- ☒ A.组件内:
`export default defineComponent({ directives:{`

- ```

 "指令名":{
 mounted(elem){ //对elem执行原生DOM操作 }
 }
 }
})

☐ B.组件内:
export default defineComponent({
 directives:{
 "指令名":{
 inserted(elem){ //对elem执行原生DOM操作 }
 }
 }
})

☒ C.main.ts中
var app=createApp(App);
app.directive("指令名",{
 mounted(elem){
 //对elem执行原生DOM操作
 }
})

☒ D.main.ts中
Vue.directive("指令名",{
 inserted(elem){ 对elem执行原生DOM操作 }
})

```

【正确答案】 A,C

【答题时间】 2021-11-25 08:14:50

【答案解析】 全局自定义指令: main.ts中:var app=createApp(App);app.directive("指令名",{ mounted(elem){ //对elem执行原生DOM操作 }}}——所以B正确组件内自定义指令export default defineComponent({ directives:{ "指令名":{ mounted(elem){ //对elem执行原生DOM操作 } } } })——所以D正确【审核人】魏丽君，【录题人】魏丽君

## 47、【多选题】

| 得分: 0分

关于ES5严格模式下说法正确的是: ()

- ☐ A.禁止给未声明的变量赋值
- ☒ B.静默失败升级为错误
- ☒ C.匿名函数自调或普通函数调用中的this指向window
- ☐ D.必须用let代替var声明变量

【正确答案】 A,B

【答题时间】 2021-11-25 08:14:50

【答案解析】 总结: ES5严格模式4项规定(1). 禁止给未声明的变量赋值——所以A正确(2). 静默失败升级为错误——所以C正确(3). 匿名函数自调或普通函数调用中的this, 不再指向window, 而是值为undefined——所以D错误(4). 禁用了arguments.calleelet是ES6的规定, 不是ES5的。【审核人】魏丽君，【录题人】魏丽君

## 48、【多选题】

| 得分: 0分

关于ref和toRefs两个函数区别, 描述正确的是 ()

- ☐ A.ref函数专门用于为对象中每个成员, 分别添加监控, 后续可解构出每个受监控的成员单独使用。
- ☒ B.ref函数专门用于将一个不受监控的数据, 变为整体受监控的数据
- ☒ C.toRefs函数专门用于将一个不受监控的数据, 变为整体受监控的数据
- ☐ D.toRefs函数专门用于为对象中每个成员, 分别添加监控, 后续可解构出每个受监控的成员单独使用。

【正确答案】B,D

【答题时间】2021-11-25 08:14:50

【答案解析】问题: 直接放在`setup()`中的普通变量, 默认不受监控。解决: 可用`ref()`包裹, 让该变量变为受监控的——所以, A正确, D错误问题: 虽然`ref()`包裹一个对象时, 整个对象是受监控的。但是, 解构出对象中每个成员单独使用时, 每个成员不是受监控的。解决: 使用`toRefs()`将原对象中每个成员都变为受监控的。再解构出每个成员, 就都是受监控的了。——所以, B错误, C正确【审核人】魏丽君, 【录题人】魏丽君

#### 49、【多选题】

得分: 2分

下列关于VUE中绑定语法描述正确的是: ()

- ☒ A.VUE中`{{}}`内不能放分支结构
- ☐ B.VUE中`{{}}`内不能放访问数组元素的语句
- ☒ C.VUE中`{{}}`内不能放循环结构
- ☐ D.VUE中`{{}}`内不能放函数调用语句

【正确答案】A,C

【答题时间】2021-11-25 08:14:50

【答案解析】Vue的`{{xxx}}`中和ES6的模板字符串的``{xxx}``中要求一样a. 可以放什么: 变量、有返回值的函数调用、运算和js表达式、创建对象、访问数组的元素——所以A, B错误b. 不可以放什么: 分支和循环等程序结构, 以及没有返回值的函数调用或js表达式——所以C和D对【审核人】魏丽君, 【录题人】魏丽君

#### 50、【多选题】

得分: 0分

下列关于TypeScript中访问控制修饰符描述正确的是 ()

- ☐ A.`public` 公有属性, 程序任何位置都可所以访问
- ☒ B.`protected` 受保护的属性只能在当前class内访问, 出了当前class范围, 就不能用
- ☒ C.`private` 私有属性只能在当前class内访问, 出了当前class范围, 就不能用
- ☐ D.TypeScript中如果没有写访问控制修饰符, 则该属性默认为`private`, 私有属性

【正确答案】A,C

【答题时间】2021-11-25 08:14:50

【答案解析】`public` 公有属性, 程序任何位置都可所以访问——所以A正确`protected` 受保护的属性在当前class及其子类型class中可用, 全局不可访问到。——所以B错误 `private` 私有属性只能在当前class内访问, 出了当前class范围, 就不能用——所以C正确TypeScript中如果没有写访问控制修饰符, 则该属性默认为`public`, 私有属性——所以D错误【审核人】魏丽君, 【录题人】魏丽君