

1. 前端常做的优化有哪些？
2. 封装一个函数，返回两个数之间的随机数。
3. splice这个函数的作用是什么？参数列表是？每个参数的意义是？
4. 随机生成一个数组，数组中的元素必须都是number且不重复且在指定的范围内。(数组长度和数组元素的取值范围自定)。
5. var关键词是怎么造成变量作用域提升和降低的。
6. 你觉得学习一门开发语言的步骤应该是怎么样的？
7. 写出一段数组排序的代码。(升、降序不限制，排序方式使用冒泡和选择分别实现)， 并且要求要有优化处 。
8. 封装一个操作cookie的类，实例对象有方法对cookie中的数据进行增删改查。
9. DOM0级的事件绑定和DOM2级的事件绑定的区别是什么。 举个DOM2级的事件绑定中你常做的兼容处 。
10. 描述ajax的运行机制及兼容。(代码)
11. post请求和get请求的区别？各自的使用场景
12. 模拟一个事件委托的应用场景，举出事件委托的好处，分别用JS原生和 JQuery 实现这个事件委托。(先描述场景，再描述事件委托的优点，最后写出两方种方式的代码)。
13. 描述下this这个变 以及各种不同情况的this的指向。(函数内外的this，对象方法中的this，构造函数中的this)。
14. 原型和原型链分别是什么？
15. 面向对象的特点？（请举例说明）

16. 你知道的继承方式有哪几种，分别有什么特点，有什么优缺点？
17. 你对ES6的了解？
18. 在封装类时，如何判断某个属性或者方法应该声明成公有的还是原型的？
19. 什么是JSON？js如何解析JSON？如何将js数据转成JSON？
20. 你在以前写代码的过程中遇到的封装有哪些，你对封装的理解是什么？
21. 什么是跨域，跨域是由什么引起的？。
22. 函数的封装什么时候要传参，什么时候需要返回值？

23.程序结构如下:

```
var i = 1;  
for(i < 10;) {  
  i++;  
}
```

console.log(i); 写出输出结果。

24. var arr = [1,2,3,4,5,6,7,8,9,10];for(var i = 0; i < arr.length; i++)
{

 if(arr[i] % 2 == 0) {
 arr.splice(i,1);

```
} }
```

console.log(arr); 执行结果?

25. 阅读代码:var fn = function() {

```
    alert(2); };
```

```
function Num() {
```

```
    var a = 1;
```

```
    fn = function() {
```

```
        alert(1);
```

```
    };this.fn = function() {
```

```
        alert(3); }
```

```
}
```

```
fn = function() {
```

```
    alert(4);
```

```
};1. fn(); 的执行结果 4
```

2. newNum().fn()的执行结果3 3. newNum();

fn();的执行结果;

26. 阅读代码

```
function Foo(){ getName = function(){ alert(1); }; return this; }
Foo.getName = function () { alert(2); };
Foo.prototype.getName=function () { alert(3); } var getName =
function
```

() { alert(4); }; function getName() { 1. Foo.getName(); 的结果

2. getName(); 的结果3. Foo().getName(); 的结果4. new
Foo().getName(); 的结果

5. new Foo(); 的结果 getName(); 的结果

alert(5) }

27. 阅读代码

```
function fun(n,o){ console.log(o); return { fun:function(m){ return
fun(m,n); } } } var a=fun(0); a.fun(1); a.fun(2); a.fun(3); var
b=fun(0).fun(1).fun(2).fun(3); var c=fun(0).fun(1);
```

作业：你发送http请求，后台返回的数据是一个数组，数组有若干个对象，但是对象的键值对的数量并不统一且并不知道有多少种可能出现的键：

```
[{'name': 'tom'
```

```
},{ 'age': 18
```

```
},{ 'age': 19,
```

```
'sex': '男' ]]
```

可以丰富数据，但是键值对数量都不可统一，而且要注意，键具体有多少个，是什么，都是未知的，且每次重新请求，这个返回值都在改变，也就是说，我们只能知道返回值的数据类型。需求:根据这个返回值的数据，在页面上创建表格。表格要求:表头就是这次请求返回值中的所有对象中出现的所有的键。

表格内容根据数组中的数据渲染。优化：当某个单元格，没有信息时，要用‘-’填充当前单元格。