

水滴公司前端笔试题

1. 一次完整 http 请求，期间一共需要握手____次

- A: 3 次
- B: 4 次
- C: 6 次
- D: 7 次

2. 在 js 语言中,0.1+0.2 ____ 0.3

- A: >
- B: <
- C: ==

3. 测试文本的 font-size 为____px

```
<style>
* {
  font-size: 12px !important;
}
#test {
  font-size: 14px;
}
div {
  font-size: 18px;
}
.test1 {
  font-size: 20px;
}
.test2 {
  font-size: 22px;
}
.test1.test2 {
  font-size: 24px;
}
</style>
<div id="test" class="test1 test2">测试</div>
```

4. 红色区域的大小为____

```
.box {  
  width: 200px;  
  height: 200px;  
  padding: 20px;  
  border: 20px solid black;  
  margin: 20px;  
  background: red;  
  box-sizing: border-box;  
}
```

5. 控制台输出结果为____

```
console.log(foo());  
function foo () {  
  return bar();  
  var bar = function () { return 5; }  
  var bar = () => 6;  
  var bar = (function(){ return 7; })()  
  function bar () { return 8; }  
}
```

- A: 5
- B: 6
- C: 7
- D: 8
- E: TypeError

6. 控制台输出结果为____

```
localStorage.setItem('show', false)  
console.log(localStorage.show || '显示')
```

7. 控制台输出结果为____

```
setTimeout(function() {  
  console.log(1);  
}, 0);  
  
new Promise(function (resolve, reject) {  
  resolve(2);  
  console.log(0)  
}).then(console.log);  
  
console.log(3);
```

8. 控制台输出结果为____

```
var a = 5;  
function todo() {  
  var a = 9;  
  return function() {  
    a = 7;  
  };  
}  
todo()();  
console.log(a);
```

9. 控制台输出结果____

```
(function() {  
  var greet = 'Hello Shuidi';  
  var toGreet = [].filter.call(greet, function(element, index) {  
    return index > 5;  
  });  
  console.log(toGreet);  
})();
```

10. 代码题：点击 li 元素的时候输出其索引值 0,1,2

```
<ul id="test">
  <li>这是第一条</li>
  <li>这是第二条</li>
  <li>这是第三条</li>
</ul>
```

11. 代码题：实现 get 函数

```
let obj = { foo: { bar: { name: 'biz' }}}
get(obj, 'foo.bar.name') // 输出：biz
obj = {}
get( obj, 'foo.bar.name') // 输出：undefined
get( obj, 'foo.bar.name', 'biz') // 输出：biz
// 函数如下：
function get (obj, path, defaultValue) {
  // todo
}
```

12. 代码题：请解码被多次编码的 URL

例子：https%253A%252F%252Fwww.baidu.com%252F%253Ftest%253D1

条件：被编码次数未知

结果：https://www.baidu.com/?test=1

13. 加分题（非必做，可以多种思路，可口述）

```
[{
  key: 0.123123,
  child:{
    key: 0.3213313,
    child:{
      key: 0.123123,
      child: {
        key: 0.723793,
        ...
      }
    }
  }
}, {
  key: 0.788798,
  child:{
    key: 0.312308,
    child:{
      key: 0.123123,
      child: {
        key: 0.3213313,
        child: {
          key: 0.866668
        }
      }
    }
  }
}]
```

上面的数据结构中，不同层级的 key 可能会相同。实现一个方法，调用时更新上面数组的 key 值，使所有的 key 对应的值变更为新的随机数，并且保证更新前相同的 key 更新为新值后也依然相同即可