

1. 什么是跨域
2. 各种跨域实现方法
3. this 作用域
4. 单列以及多列布局
5. BFC
6. w3c 盒子模型和 IE 盒子模型
7. 给定一个元素获取它相对于视图窗口的坐标
8. offsetHeight, scrollHeight, clientHeight 分别代表什么
9. CSS 如何实现居中，以多种方案解决
10. transition 的属性值和应用
11. rem 和 em 的区别
12. js 的闭包问题以及闭包继承
13. 严格模式的特性
14. ES6 的 promise
15. 原型与原型链
16. 如何实现图片滚动懒加载
17. 输入一个 url 后发生了什么？
18. 说出 es6 的三个特性
19. CSS 中的长度单位
20. 如何实现一个矩形的椭圆边角
21. vue 的优点和特性，如何实现双向绑定
22. UMD 规范和 ES6 模块化，Commonjs 的对比

23. 实现一个常见的移动端三栏布局。上下定高且固定 fix ,中间栏自适应高度且可以滚动。

24. resize 和 scroll 事件的性能优化

25. 类型转换

一段js 代码，如下：

```
(!(~+[])+{})[-[~+""]][+[]][~+[]] + ~~!+[]+({}+[])[~!+[]]~+[]，试着去运行它
```

26. 前缀表达式的中缀形式

27. 链式调用，ECMA262 曾有收到过这样的提议 “Optional Chaining”：当我们需要查

找一个很深层的属性时，经常需要确保每一级父节点的存在，

```
var street = user && user.address && user.address.street;
```

建议使用 “Optional Chaining Operator” 来简化类似的操作：

```
var street = user?.address.street;
```

一旦 “?.” 前面的表达式返回 null 或 undefined，则整个表达式返回 undefined，

请实现一个函数完成类似的功能。

28. 如何实现缓存机制

29. 如果服务器端更新脚本，如何保证客户端不受缓存机制影响，实时更新脚本

30. 函数作用域以及块作用域，对比 ES5 以及 ES6 在块级作用域上的差别

31. for 函数中的 i 变量为什么不能保存？原理是什么？

32. HTTP 有几种请求头？option 请求头的作用

33. 原生 ajax 代码流程。ajax 中有个 async 属性，有了解吗？

34. http 和缓存有关的状态码，以及所有的 http 状态码

35. get 和 post 区别

36. nginx 的好处？nginx 和 node 的比较

37. flex 特性

38. canvas 切图的原理

39. javascript 模块化规范