

前端面经part one

1、水平垂直居中

```
1  *{
2      margin: 0;
3      padding: 0;
4  }
5  .wrapper{
6      width: 300px;
7      height: 300px;
8      border: 1px solid black;
9      display: flex;
10     justify-content: center;
11     align-items: center;
12 }
13 .content{
14     width: 100px;
15     height: 100px;
16     line-height: 100px;
17     text-align: center;
18     border: 1px solid orange;
19 }
```

```
1  * {
2      margin: 0;
3      padding: 0;
4  }
5  .wrapper {
6      width: 300px;
7      height: 300px;
8      border: 1px solid black;
9      position: relative;
10 }
11 .content {
12     width: 100px;
13     height: 100px;
14     line-height: 100px;
15     text-align: center;
16     border: 1px solid orange;
17     position: absolute;
18     left: 50%;
19     top: 50%;
20     margin: -50px 0 0 -50px;
21 }
```

```

1  * {
2  |   margin: 0;
3  |   padding: 0;
4  | }
5  .wrapper {
6  |   width: 300px;
7  |   height: 300px;
8  |   border: 1px solid black;
9  |   position: relative;
10 | }
11 .content {
12 |   width: 100px;
13 |   height: 100px;
14 |   line-height: 100px;
15 |   text-align: center;
16 |   border: 1px solid orange;
17 |   position: absolute;
18 |   left: 50%;
19 |   top: 50%;
20 |   transform: translate(-50%, -50%);
21 | }

```

2.js基本数据类型

原始值 Number、String、Boolean、undefined、Null

引用值 Object、Array、function.....

3.typeof()的返回值

Number、String、Boolean、undefined、Object (null、array) 、function

4.css选择器及权重比较

! important (infinity)

行间样式 (1000)

id选择器 (100)

class选择器、属性选择器、伪类选择器 (10)

标签选择器、伪元素选择器 (1)

通配符选择器 (0)

注：权重值是256进制。

5.隐藏一个元素的方法

opacity:0 (通过改变元素的透明度，不脱标)

position:absolute (通过定位将元素定到远离页面的位置，脱标)

display:none (通过隐藏盒子属性，脱标 直接从DOM渲染树消失，灭族且毁尸，不留坟。)

visibility:hidden (通过隐藏盒子属性，不脱标 子孙节点消失由于继承了hidden，通过设置 visibility: visible;可以让子孙节点显示)

6.类数组转化为数组

1、Array.prototype.slice.call()

这种方法是借用了数组原型中的slice方法，返回一个数组。slice方法的内部实现：

2、Array.from()

Array.from()是ES6中新增的方法，可以将两类对象转为真正的数组：类数组对象和可遍历（iterable）对象（包括ES6新增的数据结构Set和Map）。

3、扩展运算符（...）

同样是ES6中新增的内容，扩展运算符（...）也可以将某些数据结构转为数组。

7.类数组对象

类数组对象长得像数组，也可以当数组用，但不是数组。

类数组必须有几个组成部分：

属性要为索引（数字）属性，必须有length属性，最好加上push属性，一旦给对象添加splice方法，长得便像数组一样，也可以当数组用。

```
9      <script>
10      const arr = {
11        "0": "a",
12        "1": "b",
13        length: 2,
14        push: Array.prototype.push,
15        splice: Array.prototype.splice,
16      };
17      console.log(arr);
18    </script>
```

8.判断一个对象是不是数组的方法

1、A instanceof B，判断A对象的原型链上是否有B的原型。

例 {} instanceof Array ---> true

2、Object.prototype.toString.call()

返回值 '[object Array]'等

3、constructor方法

例 const arr = 123;

arr.constructor-----

4、Array.isArray()方法

例 const arr = 34;

arr.isArray-----> true/false

9、事件循环 eventloop

相关概念1、执行栈 call stack

执行栈是一个数据结构，用于存放各种函数的执行环境，每个函数执行之前，他的相关信息会被加入到执行栈；函数调用之前，创建执行环境然后加入到执行栈；函数调用后，销毁执行环境。

相关概念2、异步函数

某些函数不会立即执行，需要等到某个时机到达后才会执行。

例 事件处理函数、setTimeout (function () {}, time) 等。

异步函数的执行时机会被宿主环境（js运行的环境）所控制。

相关概念3、事件队列

分类：宏队列、微队列（MutationObserver用于监听某个DOM对象的变化、Promise）先微再宏。

当浏览器线程发生了某些有处理程序的事情，它会将该处理程序加入到一个叫事件处理的内存；当js引擎发现执行栈call stack 里已经没有任何内容后，会将事件队列中的第一个函数加入到执行栈中执行。

事件循环eventloop

js引擎对事件队列的取出执行方式以及与宿主环境（js运行的环境）的配合，称为事件循环。