**学习笔记**

**1、javascript**

①undefined、null、NaN、0、-0、“”都转化为false。

②&&（与）当运算符两边都为真的时候返回true，否则返回false；||(或)运算符其中一个为真的时返回true，两边都为假的时候返回false。

**2、scss**

①简介：Sass是成熟、稳定、强大的CSS预处理器，而SCSS是Sass3版本当中引入的新语法特性，

完全兼容CSS3的同时继承了Sass强大的动态功能。

②变量：变量用来存储需要在CSS中复用的信息，例如颜色和字体。SASS通过$符号去声明一个变 量。

1. **凡是带有inline的元素都具有文字特性**

**4、ES6**中：（在for循环函数下）（for循环还有一个特别之处，就是设置循环变量的那部分是一个父作用域，而循环体内部是一个单独的子作用域。）

var是属于全局的属性；

let是局部的属性，只在局部有效，带自己的作用域；

**5、事件冒泡和事件捕获**

事件传递有两种方式：冒泡与捕获。

事件传递定义了元素事件触发的顺序。 如果你将 <p> 元素插入到 <div> 元素中，用户点击 <p> 元素, 哪个元素的 "click" 事件先被触发呢？

在 冒泡 中，内部元素的事件会先被触发，然后再触发外部元素，即： <p> 元素的点击事件先触发，然后会触发 <div> 元素的点击事件。

在 捕获 中，外部元素的事件会先被触发，然后才会触发内部元素的事件，即： <div> 元素的点击事件先触发 ，然后再触发 <p> 元素的点击事件。

addEventListener() 方法可以指定 "useCapture" 参数来设置传递类型：

addEventListener(event, function, useCapture);

默认值为 false, 即冒泡传递，当值为 true 时, 事件使用捕获传递。

**6、关于正则表达式（贪婪匹配原则）**

[^]写在括号里面的^代表非，除了的意思

（abc|bcd）“|”代表或，正则表达式中的或

①题目检验一个字符串首尾是否含有数字？ 答：var reg = /^\d[\s\S]\*\d%/g; var str = '123abc123';

**7、css中伪类选择器的先后顺序（爱恨法则）**

link > visited > hover >active （根据样式的先后顺序，覆盖前面的样式）

**8、css中关于页面布局**

两栏布局的时候，如果右面的主区域不能设置宽度的时候，作如下操作：

左侧的区域float：left并且设置宽度；右侧的区域不设置宽度，加上overflow:hidden；

（若想两边有间隙的话，就要设置浮动元素的margin值）

**9、css处理等高**

height设置一个很大的值例如10000，那么就设置内容之外的为负的，

例如margin-bottom： -9990，内容为10px

**10、关于三栏布局**

三栏布局的时候，一般情况下左右两侧为导航栏，中间为内容区。但是搜索引擎在读代码的时候

是从上到下读的，会认为越靠前的越重要（忽略css代码）。所以在处理三栏布局的时候先写主要的内容区，把主内容区两边的区域留出来（margin：0 300px）。然后左右两侧的导航栏就用

定位的方式去将区块放置在左右两边。

**11、关于行高（line-height）（以16px为例）**

行高有四种写法;px、百分比、em、没有单位

px的时候行高就是固定的；百分比和em是字体大小的倍数（2em = 200% = 32px） 先计算再继承

没有单位的时候是先继承再计算，若line-height：2；意味着就是16的两倍32px

**12、关于图片底部白边的处理**

1.将font-size:0 2.将图片的display属性设置为block

**13、关于堆叠上下文**

一、创建堆叠上下文的元素：①html根元素 ②设置z-index的定位元素

二、在z轴上的排列方式

①当z-index：auto的时候不会创建堆叠上下文，为数值的时候（无论正负）都会创建

②有堆叠上下文的元素靠后显示，z-index为负值的时候也不影响排序；没有堆叠上下文的时候 按照z-index的值来显示

**14.js中数据类型的存储（原始值和引用值唯一的不同是赋值的不同）**

①原始值存储在栈中（stack）：栈内存之间的赋值是拷贝的关系，所以当a=10，b=a，a=20的时 候，因为b=a是拷贝的，所以a=20不会影响b的赋值，所有console.log(b)是10。

②引用值存储在堆中（heap）

**15.逻辑运算符**

①&&（全真才为真） 先看第一个表达式转换为布尔值的结果，如果结果为真，那就看第二个表达式转换为布尔

值的结果；若只有两个表达式，第一个为true，第二个也为true，那么

就返回第二个表达式的结 果。如果第一个表达式为false，那么就直接返回0，不继续看第二个表达式。（第一个为真，就返回第二个；第一个为假，那么久直接返回第一个）。(遇到假就停)

②|| 遇到真的就返回真，遇不到就继续往后找。（遇到真就停）

③！ 变布尔值取反

**16.最小特权原则：**函数当中局部作用域中的变量无法在全局中访问。

**最大特权原则：**不管在哪声明变量，在任何位置都能访问，这个叫做最大特权原则。这样对程序并不安全。

**17.关于js的变量提升问题**