**青盟学习笔记**

1. **学习git**

Git在提交文件的步骤：

1. git add “文件名称（test.html）”
2. git commit -m “备注（创建test.html文件）”
3. Git diff 显示对文件的修改变化内容
4. **补充CSS**
5. 内联样式优先级最高！外部样式与内部样式的优先级就看代码的先后顺序
6. 关于选择器权重：!important > 内联 > id > class > 伪类 > 标签 | 伪元素 > 通配符
7. **常见样式**
8. background-image | 不重复:no-repeat | backgropund-position： 水平 垂直 |
9. **关于CSS**
10. 关于外边距合并实例：

解决： 1.给父元素加一个overflow：hidden；

2给父元素添加一个边框

3父元素或者子元素浮动或者定位也会解决

父元素套一个子元素，涉及到一个BFC的概念。记得差BFC的原理 hasLAyout查询

1. 块元素垂直分布，宽度自适应；行元素水平排列，宽度是内容决定。
2. 相对定位position：relative；相对自身原始位置进行定位，空间不释放；

绝对定位position：absolute；相对于最近的已经定位的祖先元素进行定位，空间释放

1. 浮动会导致空间释放；

1. **关于词法作用域**
2. 词法作用域：是指函数在定义（或声明）它们的作用域里运行，而不是在执行（或 调用）它们的作用域里运行。
3. **关于闭包**
4. 含义：闭包就是两个函数嵌套，在外部函数内声明一个局部变量，在内部函数里调 用外部函数的局部变量，导致外部函数的局部变量不能释放，导致内存泄漏，从而形成 闭包；
5. 解决方法： 立即执行函数（function （） { }）（）；又称即时函数。

**渡一教育**

**编程：高内聚，弱耦合**

1. **关于函数**
2. 函数声明：function test() {}命名小驼峰式
3. 函数表达式（匿名表达式）： 命名函数表达式：var test = function demo() {} 在表达式中调用test，调用demo没有用。Test.name=demo；匿名函数表达式：var test = function（）{}
4. 参数：形式参数 -- 形参 （函数名后面的）：求形参的数值，函数名.length；

实际参数 -- 实参 ： 可以传多个值，找对应的形参传值；若实参的值多余形参的值，剩余的实参数值会被传到arguements的数组里面；（不限制数据类型）

！Arguements存的是实参列表

！Arguements里面的实参跟形参是映射关系，若实参只有一位，形参两位，arguements【1】的值是undefind。

1. return： （1）终止函数 （2）返回值，把一个值返回到函数外部。
2. **js运行三部曲**
   1. **语法分析**
   2. **预编译**
   3. **解释执行**

**关于预编译（函数声明整体提升 | 变量 声明提升）**

1. 系统会把函数声明提升到逻辑的最前面
2. Var a = 2; a是变量声明，所以提升的时候只提升变量a；

3.Imply global暗示全局变量。任何变量，如果变量未声明就赋值，那么就归全局对象所有；4.一切声明的全局变量，全是window的属性；

**预编译发生在函数执行的前一刻：**

1. **创建AO对象（Activation Object）执行期上下文**
2. **找形参和变量声明，将变量和形参名作为AO属性名，值为undefined**
3. **将实参和形参想统一**
4. **在函数体里面找函数声明，值赋予函数体**
5. **全局也有预编译（1）生成一个Go对象，Global Object（2）没有实参，所以忽略第三部**

**九、关于作用域、作用域链**

1. 执行期上下文: 当函数执行时，会创建一个称为执行期上下文的内部对象。一个执行期上下文定义了一个函数执行时的环境，函数每次执行时对应的执行上下文都是独一无二的，所以多次调用一个函数会导致创建多个执行上下文，当函数执行完毕，它所产生的执行上下文被销毁。
2. Scope：每个javascript函数都是一个对象，对象中有些属性我们可以访问，但有些不可以，这些属性仅供javascript引擎存取，[[ scope ]]就是其中一个。[[ scope ]]