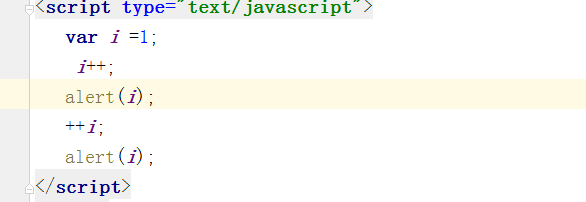
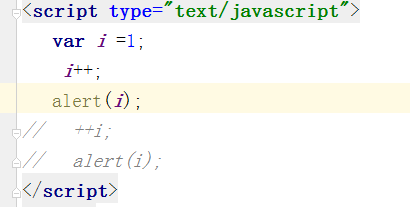
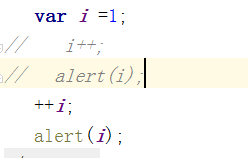
周会



这样的输出结果分别是2和3



这样的运行结果是2



这样的运行结果是2

之前输出的结果是2和3是因为将i++后的值赋予了++i，应该分别打印，出现的结果都是2(不参与运算的情况)。

在参与运算的情况

var i = 1; var i =1;

i = i++ +1; i = ++i +1；

alert(i); i=2 alert(i); i=3;

1.attr和prop的区别：

attr适用于自定义标签属性,返回的是字符串。

Prop适用于自带属性,返回的是true或false.

2.关于libraries 的规范：

1. 文件名和类名一致:
2. 采用大驼峰命名
3. 加载时也必须与类名一致（大驼峰）
4. 用类时小写

例如EDK项目中的：CronSchedule.php文件名 ,类名和文件名一致，类名也是 CronSchedule,采用的是大驼峰命名。加载类时：$this->load->library(array(‘CronSchedule’))。 使用类时：$this->cronschedule(小写)。

3.关于config的规范

1. 文件名小写
2. 用的时候也是小写
3. 加载时也小写且不带后缀

例如在EDK项目中:

autoload.php 文件名小写 ，在配置时 $autoload[‘config’] = array(‘小写并且不带后缀名’)；

4.关于helper的规范：

1. 自定义时小写(文件名小写,单词间用下划线隔开)
2. 加载时用的\_helper前面的名称
3. MY\_(扩展自己的helper,也可以在配置中更改前缀).

在EDK项目中：

1. action\_helper.php 文件名（自定义小写，单词件用下划线隔开）。
2. 文件名为action\_helper.php加载时 $this->load->helper(‘action’) ,去掉\_helper.php 后缀。
3. 如果是MY\_这样的前缀(属于扩展自己的helper) ,如需要修改，在application/config/config.php 修改前缀

5.关于model的规范:

1. 类名和文件名一致
2. 单词间下划线隔开
3. 加载的时候全部小写.

在EDK项目中:

类名和文件名一致，用大驼峰的命名方法 ，加载时 $models = array（‘follow\_model（小写）’）；

在使用时 $this->CI->(model(小写)).

1. Views的规范(文件名小写)。

在EDK项目中: 例如 package文件夹下面的 create.php(文件名小写)。

1. 关于导出（一个方法处理一种类型的数据）.