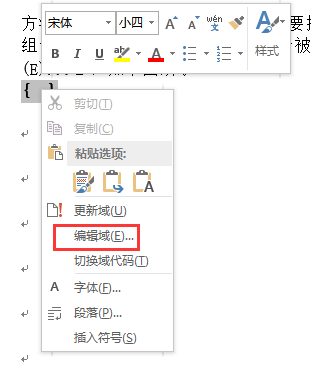
# 模板编辑步骤

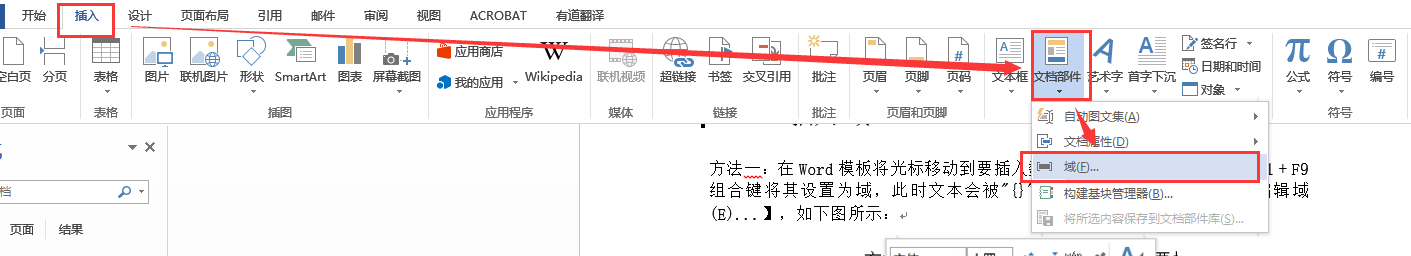
巡检平台中的巡检文档生成需要依赖文档模板的编辑，需要提前按要求制作好想要生成的文档模板，这里对模板的制作步骤进行简要说明。

## 插入域

方法一：在Word模板将光标移动到要插入数据的地方，然后键盘使用Ctrl + F9组合键将其设置为域，此时文本会被"{}"包围，接着鼠标右键选择【编辑域(E)...】，如下图所示：

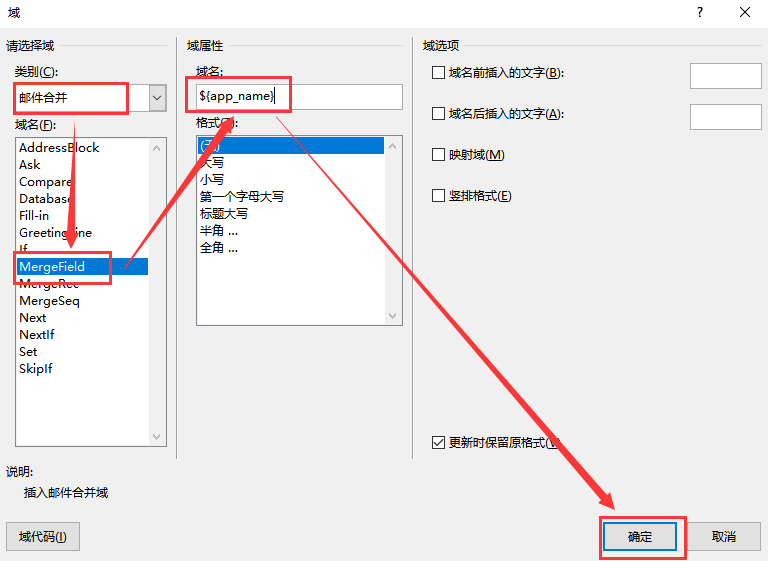


方法二：在Word模板将光标移动到要插入数据的地方，依次点击word导航栏中的“插入”-“文档部件”-“域”，此时会直接打开“编辑域”窗口，如下图：



## 编辑域

在弹出的“编辑域”对话框中，类别选择“邮件合并”，域名选择 "MergeField"，域属性中的域名填入模版表达式，点击【确定】按钮，如下图：（注意：之后如果要对某个域进行修改，也需要在这里打开编辑域进行修改，只在页面上修改不生效）



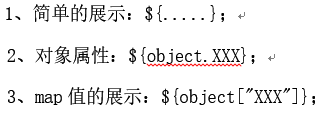
以上操作即完成了在模板文档中的指定位置插入了后端数据中的一个“app\_name”属性数据的操作。下面将对模版表达式的语法进行说明。

# 模板表达式语法及使用

使用只是开始，语法才是修行。这里的语法是兼容freemarker的，下面对常用语法的使用进行简单说明和展示。（目前后台只封装了一个“list”属性的集合数据，集合中就是巡检结果提交上来的JSON格式的数据，所以我们只需要写一个集合遍历，之后在集合遍历中通过各种语法结构，输出JSON中key对应的数据即可，之后的样例展示也使用的是这个集合数据展示，供参考）

## 属性展示

**语法：**

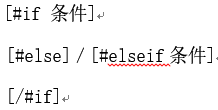


**说明：**这里是三种数据展示方式。

**样例：**略（参考之后语法的样例）

## 条件判断

**语法：**



**说明：**if 与java的运算符相差不多，相等可直接使用“==”，字符串也不例外。注：当使用大于号时，应用（）包起表达式，或者使用gt，否则if语句将提前结束，会报语句不完整报错。在文档中，一个“域”内可以写多个表达式，这里的if语句也可以放到一个“域”内编辑，也可以分开成多个域编辑，之后的语法也一样，根据实际需求进行编辑。

**样例1：**对象非空判断，如下：

（建议在之后的每个属性输出前都进行非空判断，防止程序因为某个属性不存在而报错）

«[#if list??]»

存在list属性

«[#else]»

不存在list属性

«[/#if]»

**样例2：**这里的条件判断有没有名称是“list”的数据，并且list的元素数量大于0，如下：

«[#if list?? && (list?size>0)]»

List大小为：«${list?size}»

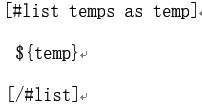
«[#else]»

List为空

«[/#if]»

## 集合/数组的遍历

**语法：**



**说明：**这个语法表示循环遍历一个名为“temps”的集合，每次取出一个元素赋值给temp，并输出temp，即：循环遍历temps中的每个元素并输出。在文档中，一个“域”内可以写多个表达式，但这里建议拆分成多个“域”编辑，方便集合中多个元素的输出，也方便模板语法阅读。

**样例1：**这里遍历list集合，输出其中每个元素的“app\_ip”属性数据（注意：模板编辑的方法，[参考1](#_模板编辑步骤)）

这里我们先进行对象判空

«[#if list?? && (list?size>0)]»

开始遍历

«[#list list as item]»

这里输出元素字段前也增加判空操作：«[#if item.app\_ip??]${item.app\_ip}[/#if]»

«[/#list]»

«[/#if]»

**样例2：**集合的嵌套遍历，集合中的某一个属性集合仍为集合，我们可以在集合中再遍历这个属性集合，如下：

«[#if list?? && (list?size>0)]»

«[#list list as item]»

«[#if item.es\_indices?? && (item.es\_indic»

«[#list item.es\_indices as es\_item]»

«[#if es\_item["pri.store.size"]??]${es\_it»

«[/#list]»

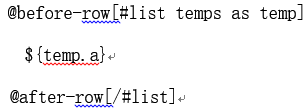
«[/#if]»

«[/#list]»

«[/#if]»

## 表格数据插入

**语法：**



**说明：**表格数据的插入事实上也是使用的集合遍历，区别在于，只需要在表格的第一列和最后一列加上集合遍历即可，这里注意@before-row和@after-row的使用。

**样例1：**将集合list中的es\_indices集合，按表格的方式输出，并过滤输出pri.store.size字段值符合条件的行，模板编辑如下：

«[#if list?? && (list?size>0)]»

«[#list list as item]»

«[#if item.es\_indices?? && (item.es\_indic»

|  |  |  |
| --- | --- | --- |
| pri.store.size字段 | health字段 | status字段 |
| «@before-row[#list item.es\_indices as es\_»«${es\_item["pri.store.size"]}» | «${es\_item.health}» | «${es\_item.status}»«@after-row[/#list]» |

«[/#if]»

«[/#list]»

«[/#if]»

## 其它语法的使用

以上内容仅对常见语法进行了说明演示，已经可以解决80%的模板需求，如在使用的过程种需要用到其它语法，可以参考：<https://www.jianshu.com/p/1ff072643b75>地址中语法说明进行使用。

# 常见问题

## 生成文档报错

这里可能有两种原因：

1. 文档模板编辑有问题：这个问题最大原因可能就是文档里输出了后台解析不到的数据，所以为了防止这类问题，在编辑模板时，一定要在属性输出前，对属性增加非空判断。
2. 上传的数据格式有问题：由于我们目前后台处理的数据都是前台采集上传上来的，我们对采集上来的数据格式进行了约定，上传的数据必须符合JSON格式，这里我们可以使用JSON在线验证工具对采集的数据进行验证，json验证工具：<https://www.baidufe.com/fehelper/json-format/index.html> ，下面展示一个标准JSON格式的数据供参考：

{

"IP\_Address": "10.1.226.112",

"Hostname": "fqzapp1",

"OS\_version": "Linux fqzapp1 2.6.32-642.el6.x86\_64",

"CPU\_cores": "56",

"Max\_user\_processes": "10240",

"Openfile": "65535",

"Mem\_use\_meminfo\_total": "128557",

"Mem\_used": "113057",

"Mem\_free": "15499",

"disk": " Filesystem Type Size Used Avail Use% Mounted on /dev/mapper/VolGroup-lv\_root ext4 317G 88G 213G 30% / tmpfs tmpfs 68G 140k 68G 1% /dev/shm /dev/sda2 ext4 500M 42M 432M 9% /boot /dev/sda1 vfat 210M 271k 210M 1% /boot/efi /dev/mapper/VolGroup-lv\_home ext4 53G 106M 50G 1% /home cgroup\_root tmpfs 68G 0 68G 0% /cgroup /soft/rhel-server-6.8-x86\_64-dvd.iso iso9660 3.9G 3.9G 0 100% /soft/iso ",

"es\_indices": [

{

"pri.store.size": "650b",

"health": "yellow",

"status": "open",

"index": "twitter",

"pri": "5",

"rep": "1",

"docs.count": "0",

"docs.deleted": "0",

"store.size": "650b"

},

{

"pri.store.size": "650b",

"health": "yellow",

"status": "open",

"index": "twitter",

"pri": "5",

"rep": "1",

"docs.count": "0",

"docs.deleted": "0",

"store.size": "650b"

}

]

}