<http://blog.csdn.net/niuch1029291561/article/details/8190617>

版权声明：本文为博主原创文章，未经博主允许不得转载。

**判断对象是否存在**，若成立说明存在 <#if blockObjList ??></#if>

       <#if blockObjList ??>  <#else>  </#if>

**判断知否和某一个值相等**(先判断是否存在，在判断相等，如果该指定的参数不存在还比较相等的话就回出错) <#if componentId ?? &&  componentId==1></#if>

**list循环**(循环遍历list并给遍历的对象取别名attribute以方便后面对象调用)，if\_exists在调用一个变量需要先判断是否为空不然为空时会出错,\_index为取出行号索引从0开始,?size可以判断list大小  <#list lockObj.components              as  attribute>  ${attribute.contenturl?if\_exists} ${attribute\_index}  </#list>    {lockObj.components ?size}

**ftl.map     
    <#list Map?keys as s>     
        ${Map[s]}     
    </#list>**

**变量与指定值相加减**，需要先判断该变量是否为数字类型，?number 然后转换为数字类型再做相减运算(变量number与指定的2相减) ${blockWidth?number-2}

**日期转换**

        时间格式化的方法代码   
        最好加上 判断下c时候存在     
       <#if c??>     
             ${c.addtime?string("yyyy-MM-dd HH:mm:ss")     
       </#if>    
  
**判断长度并截取代码**      <#if c.titleH1??>     
          <#if c.titleH1?length lt 33>     
              ${c.titleH1}

**Ftl变量转换代码**      ?number转为数字     
      ?c转为字符串     
      ?string转为字符串     
      ?currency（货币）     
     ?percent(百分比)     
      trim 删除字符串首尾空格 ${“  String ”?trim} 结果为String     
      split使用指定的分隔符将一个字符串拆分为一组字符串     
     <#list “This|is|split”?split(“|”) as s>     
        ${s}     
     </#list>

${“strabg”?replace(“ab”,”in”)} 结果为string     
${“string”?contains(“ing”)?string} 结果为true     
注意：布尔值必须转换为字符串才能输出     
${“string”?index\_of(“in”) 结果为3    
${“string”?index\_of(“ab”) 结果为-1    
length返回字符串的长度 ${“string”?length}结果为6    
lower\_case将字符串转为小写     
${“STRING”?lower\_case}à结果为string     
upper\_case将字符串转为大写     
${“string”?upper\_case}à结果为STRING     
ends\_with 判断某个字符串是否由某个子串结尾，返回布尔值。     
${“string”?ends\_with(“ing”)?string} 返回结果为true     
注意：布尔值必须转换为字符串才能输出     
html 用于将字符串中的<、>、&和“替换为对应得&lt;&gt;&quot:&amp     
index\_of（substring,start）在字符串中查找某个子串，返回找到子串的第一个字符的索引，如果没有找到子串，则返回-1。     
Start参数用于指定从字符串的那个索引处开始搜索，start为数字值。     
如果start大于字符串长度，则start取值等于字符串长度，如果start小于0， 则start取值为     
${‘str’?substring(0)}à结果为str     
${‘str’?substring(0,1)}à结果为s     
  
2.cap\_first 将字符串中的第一个单词的首字母变为大写。     
${‘str’？cap\_first}à结果为Str     
  
3.uncap\_first将字符串中的第一个单词的首字母变为小写。     
${‘Str’？cap\_first}à结果为str     
  
4.capitalize将字符串中的所有单词的首字母变为大写     
${‘str’？ capitalize}à结果为STR     
date,time，datetime将字符串转换为日期     
例如：     
<#assign date1=”2009-10-12”?date(“yyyy-MM-dd”)>     
<#assign date2=”9:28:20”?time(“HH:mm:ss”)>     
<#assign date3=” 2009-10-12 9:28:20”?time(“HH:mm:ss”)>     
${date1}à结果为2009-10-12    
${date2}à结果为9:28:20    
${date3}à结果为2009-10-12 9:28:20    
  
?number转为数字   
?c转为字符串   
?string转为字符串   
?currency（货币）   
?percent(百分比)   
trim 删除字符串首尾空格 ${“  String ”?trim} 结果为String   
split使用指定的分隔符将一个字符串拆分为一组字符串   
<#list “This|is|split”?split(“|”) as s>   
${s}   
</#list>   
${“strabg”?replace(“ab”,”in”)} 结果为string   
${“string”?contains(“ing”)?string} 结果为true   
注意：布尔值必须转换为字符串才能输出   
${“string”?index\_of(“in”) 结果为3   
${“string”?index\_of(“ab”) 结果为-1   
length返回字符串的长度 ${“string”?length}结果为6   
lower\_case将字符串转为小写   
${“STRING”?lower\_case}à结果为string   
upper\_case将字符串转为大写   
${“string”?upper\_case}à结果为STRING   
ends\_with 判断某个字符串是否由某个子串结尾，返回布尔值。   
${“string”?ends\_with(“ing”)?string} 返回结果为true   
注意：布尔值必须转换为字符串才能输出   
html 用于将字符串中的<、>、&和“替换为对应得&lt;&gt;&quot:&amp   
index\_of（substring,start）在字符串中查找某个子串，返回找到子串的第一个字符的索引，如果没有找到子串，则返回-1。   
Start参数用于指定从字符串的那个索引处开始搜索，start为数字值。   
如果start大于字符串长度，则start取值等于字符串长度，如果start小于0， 则start取值为   
${‘str’?substring(0)}à结果为str   
${‘str’?substring(0,1)}à结果为s   
  
2.cap\_first 将字符串中的第一个单词的首字母变为大写。   
${‘str’？cap\_first}à结果为Str   
  
3.uncap\_first将字符串中的第一个单词的首字母变为小写。   
${‘Str’？cap\_first}à结果为str   
  
4.capitalize将字符串中的所有单词的首字母变为大写   
${‘str’？ capitalize}à结果为STR   
  
date,time，datetime将字符串转换为日期   
例如：   
<#assign date1=”2009-10-12”?date(“yyyy-MM-dd”)>   
<#assign date2=”9:28:20”?time(“HH:mm:ss”)>   
<#assign date3=” 2009-10-12 9:28:20”?time(“HH:mm:ss”)>   
${date1}à结果为2009-10-12   
${date2}à结果为9:28:20   
${date3}à结果为2009-10-12 9:28:20   
  
宏定义hash内置函数代码   
Hash的内置函数     
1.hash?keys 返回hash里的所有key,返回结果为sequence     
  
2.hash?values 返回hash里的所有value,返回结果为sequence     
例如：     
<#assign user={“name”:“hailang”, “sex”:“man”}>     
<#assign keys=user?keys>     
<#list keys as key>     
${key}=${user[key]}     
</#list>    
  
Hash的内置函数   
1. hash?keys 返回hash里的所有key,返回结果为sequence   
  
2.hash?values 返回hash里的所有value,返回结果为sequence   
例如：   
<#assign user={“name”:“hailang”, “sex”:“man”}>   
<#assign keys=user?keys>   
<#list keys as key>   
${key}=${user[key]}   
</#list>

参考文献  <http://blog.csdn.net/hbhgjiangkun/article/details/7283634>