

baiduTemplate单元测试用例及功能细则

输出变量

基本输出（自动HTML转义）

`<%=value%>`

支持容错写法

`<%=value;%>`

不转义输出

`<%= value%>`

支持容错写法

`<%=value;%>`

`<%-value%>`

URL转义输出

`<%:u=value%>`

支持容错写法

`<%:u=value;%>`

UI变量在页面标签事件中使用转义

`<%:v=value%>`

支持语法容错

`<%:v=value%>`

支持HTML转义

`<%:h=value%>`

支持容错写法

`<%:h=value;%>`

变量未定义自动输出空

支持多级Object未定义（暂不考虑支持）

模板中直接输出

特殊字符

斜杠

单引、双引

注释

HTML注释支持

`<!--HTML注释方式 -->`

模板自带注释

`<%* 模板自带注释方式 *%>`

JS注释支持

`<% //JS的注释方式 %>`

判断语句

`<% if(value){%> <% }else{ %> <% } %>`

支持容错

`<%if(value)%> <% { %> <% } %><%else%><%{%><%}%>` 每个位置都可隔开

循环语句

`<% for(var i=0;i<list.length;i++){ %> <% } %>`

支持容错语法

`<%for(var i=0;i<list.length;i++)%><% { %> <% } %>`

自定义分割符

支持HTML注释的头

`<! !>`

支持含有正则的元字符

如 `<? ?>`

其他字符支持

如 `${ }`

使用

传入id

`<textarea>`

`<script>`

下次调用缓存

直接传入tpl

返回编译后的函数

调用该函数生成不同的模板

直接传入data，返回编译后的模板

data如果是非object类型

data如果是空 `{ }`

data如果是数据

性能测试对比

分隔符内支持任意JS语法

`<% var test=123;%>`

`<% function test(){ } %>`

nodejs环境支持