

template

view から呼び出される html のようなもの .

`{{ object }}` で view から渡された object を表示できる .

`{{ object|filter }}` で object に filter 関数を適用できる .

引数は `{{ object|filter:'arg' }}` のようにコロン区切りで 1 つだけ渡すことができる .

`{% templatetag %}` で `templatetag` という専用の関数を呼び出せる .

`{% extends "template_path" %}` と書くことで `template_path` の template を継承できる .

`{% block block_name %}` と書くことで継承元のテンプレートの `block_name` を上書きできる .

Listing 1 base.html 記述例

```
1 {% load staticfiles %}
2 <html>
3   <head>
4     <link href="{% static 'css/main.css' %}" rel="stylesheet">
5   </head>
6   <body>
7     <div class="contents">{% block contents %}{% endblock %}</div>
8   </body>
9 </html>
```

Listing 2 polls/index.html 記述例

```
1 {% extends "base.html" %}
2
3 {% block contents
4   メインコンテンツ
5 {% endblock contents %}
```

form

post が発生するページでは Cross Site Request Forgery 対策が必要 .

csrf_token タグを置くことで hidden を挿入してくれる .

post 先が同じ view (同じ URL) の場合 , action の中身は空で良い .

Listing 3 form タグ 記述例

```
1 <form method="POST" action="{% url 'post_page_name' %}">
2   {% csrf_token %}
3   {{ form }}
4 </form>
```

テンプレートタグ例

引数がある場合はスペース区切りで渡す .

タグ名	内容	補足
<code>{% extends template_path %}</code>	template の継承	
<code>{% load library %}</code>	拡張 template タグ読み込み	<code>library</code> はスペース区切りで複数記述可
<code>{% block blockname %}</code>	ブロック宣言開始	
<code>{% endblock blockname %}</code>	ブロック宣言終了	<code>blockname</code> はなくてもいいが , 付けることを推奨
<code>{% url url_name %}</code>	URL 名前解決	<code>urls.py</code> で設定した名前を記述 .
<code>{% static path %}</code>	静的ファイルの URL 作成	設定で prefix が変化するので tag を使う .
<code>{% for obj in itor %}</code>	繰り返しブロック開始	<code>{% load static %}</code> が必要
<code>{% endfor %}</code>	繰り返しブロック終了	<code>obj</code> は block ないで使用する名前
<code>{% if condition %}</code>	if ブロック開始	<code>condition</code> は and, or not が使用可能
<code>{% elif condition %}</code>	elif ブロック開始	
<code>{% else %}</code>	else ブロック開始	
<code>{% endif %}</code>	if ブロック終了	
<code>{% csrf_token %}</code>	csrf 対策	form タグの中に置く