

190508 - Template

HttpResponse는 활용도가 떨어짐. Template을 만들어서 렌더링 된 화면을 사용자에게 보여주자.

1. Template 활용해보기

1. render 사용

```
def index(request):  
    # parameter 1 : request, 2 : 템플릿 이름  
    return render(request, 'index.html')
```

2. 해당 app 폴더 내에, templates 라는 폴더 생성

```
(django-venv) ssoso27:~/workspace/PROJECT01/home $ mkdir templates
```

3. template 생성

```
(django-venv) ssoso27:~/workspace/PROJECT01/home/templates $ touch index.html
```

2. Template에 변수 넣어보기 (By. DTL)

DTL : Django Template Language

1. 변수

1. 사용

1. {{ 변수명 }}
2. ex) {{ my_list.0 }} {{ my_dict.key }} {{ first_name }}

2. 태그

1. 사용

1. {% 태그 %}

2. 용도

1. 콘텐츠 출력
2. 제어구조 사용 (if, for)
3. db에서 콘텐츠 가져옴
4. 다른 템플릿 태그에 액세스

3. 필터

1. 사용

1. {{ django|title }}
2. {{ my_dict|date:"Y-m-d" }}

4. 주석

1. 사용

1. {# 주석이당 #}

render 함수의 3번째 파라미터로 dict를 보낼 수 있음.

[home/views.py]

```
def dinner(request):
    menus = ['pizza', 'bob', 'chicken']
    picked = random.choice(menus)
    return render(request, 'dinner.html', {'menus' : menus, 'picked' : picked})
```

1) render로 받은 dict를, {{key name}} 으로 가져올 수 있음.

[home/templates/dinner.html]

```
...
    <h1>Today's dinner is {{picked}} </h1>
...
```

2) for문

```
...
{% for menu in menus %}
    <p>{{menu}}</p>
{% endfor %}
...
```

3) if 문

```
<p>IF</p>
{% if picked == 'chicken' %}
    <p>JMT</p>
{% else %}
    <p> this is not chicken. this is {{picked}} </p>
{% endif %}
```

Today's dinner is chicken

[Print menu]

- pizza
- bob
- chicken

[IF]

치킨 존맛탱

3. Variable Routing

path에 <변수명> 으로 넣어주고, view 메소드가 해당 변수를 파라미터로 받게끔 한다.

1. views.py

```
def hello(request, name):  
    return render(request, 'hello.html', {'name' : name})
```

2. urls.py

```
urlpatterns = [  
    ...  
    path('home/hello/<name>/', views.hello), # default type : str  
    path('home/cube/<int:num>/', views.cube), # 숫자는 int:으로 표시해준다. (타입 지정)  
]
```

3. hello.html

```
<h1>안녕 {{ name }} </h1>
```

안녕 소희

3. DTL 심화

1. {% empty %}

빈 값인지 조건을 걸어줌

```
{% for user in empty_list %}
    <li>{{ user }}</li>
{% empty %}
    현재 가입한 유저가 없습니다.
{% endfor %}
```

2. forloop

for loop 돈 횟수

```
{% for menu in chinese_foods %}
    {{ forloop.counter }} 회 :
    {% if forloop.first %}
        짜장면 + 고춧가루 <br>
    {% else %}
        {{ menu }} <br>
    {% endif %}
{% endfor %}
```

3. length filter

길이 알려주는 필터

```
{% for name in fruits %}
    {% if name|length > 5 %}
        {{ name }} : 글자가 너무 길니다. <br>
    {% else %}
        {{ name }} : {{ name|length }} <br>
    {% endif %}
{% endfor %}
```

4. Form

1) GET을 위한 form

1. [ping.html]

GET /home/pong/?data=소희 를 위한 form

```
<form action='/home/pong/'>
    이름 : <input type="text" name="data"/>
    <input type="submit" value="Submit"/> # => pong method 호출
</form>
```

2. [views.py]

```
def pong(request):
    data = request.GET.get('data') # request에서 query 받아옴
    return render(request, "pong.html", {'data' : data})

# request.GET : <QueryDict: {'data': ['소희']}>
```