

# Template Kısmı

14 Ağustos 2016 Pazar 10:30

- ★ Blog kısmı altında **templates diye bir klasör** oluşturuyoruz. İsmi ve yeri bu olmak zorunda değil ama değiştirmek için settings kısmında değişiklik yapmak lazım. Buraya direk bu isimle ve yerine uygun olarak girersek herhangi bir settings ayarıyla uğraşmadan direk erişilebilir olur.

- ★ **Template adresini** Djangoya tanıtmak için :
  - Settings.py altında :
    - 'DIRS': [**os.path.join(BASE\_DIR, 'templates')**]
  - Bu kısma yolumuzu belirtiyoruz.

- ★ Klasör altında başka bir **html sayfası** oluşturuyoruz.

- ★ Bu html sayfasına **yönlendirme** yapabilmek için :

- def home(request):  
    **return render(request, "home.html")**

- ★ Templates klasörünün altında başka bir klasör oluşturup burdan yönlendirme yapmak istersek şöyle yapabiliriz:

- def home(request):  
    return render(request, "**new/home.html**")

- ★ View üzerinden **template e parametre göndermek** için :

- View kısmında 3. parametre olarak bir dict gönderiyoruz :
  - def home(request):  
    return render(request, "home.html", {"name" : "Serhat"})
- Html kısmında ise {{ }} içerisine parametreyi giriyoruz :
  - Hello **{{ name }}**

- ★ **Db den veri çekmek ve Filtreleyerek veri göndermek** :

- **from blog.models import Post**  
  
def home(request):  
    **#posts = Post.objects.all() -> Tüm verileri gönderir**  
    **kayit\_sayisi = Post.objects.count() -> Verileri sayar**  
    **posts = Post.objects.filter( -> filtre uygulayarak veri gönderir**  
        **is\_published = True**  
    **)**  
    return render(request, "home.html", {  
        "name" : "Serhat",  
        "posts" : posts,  
        " kayit\_sayisi" : kayit\_sayisi  
    })

- ★ **Filtreleyerek saymak** için :

- def home(request):

```

    #posts = Post.objects.all()
    kayit_sayisi = Post.objects.count()
    posts = Post.objects.filter(
        is_published = True
    )
    kayit_sayisi2 = posts.count()
    return render(request, "home.html", {
        "name" : "Serhat",
        "posts" : posts,
        "kayit_sayisi" : kayit_sayisi,
        "kayit_sayisi2" : kayit_sayisi2
    })

```

★ **Len() fonksiyonu yerine count() fonksiyonunu kullanmanın amacı :**

- Len() fonksiyonu db den tüm veriyi çekip listeleyip sayar.
- Count() fonk ise db üzerinde sayım yapar ve sadece int değeri verir.

★ **Foreign key ile birbirine bağlı iki modeli birlikte kullanmak için :**

- View kısmı :
  - def index(request):
 

```

              return render_to_response("index.html", {
                  "places" : Place.objects.all(),
              })
          
```

- HTML kısmı :
  - **{% for p in places %}**

```

        {{ p.name }}
        {{ p.description }}

```

```

        {% for r in p.review_set.all %}
            {{ r.comment }}
        {% endfor %}

```

```

    {% endfor %}

```