

## Django urls.py views.py template || djangoning asosiy qismi || 4-dars

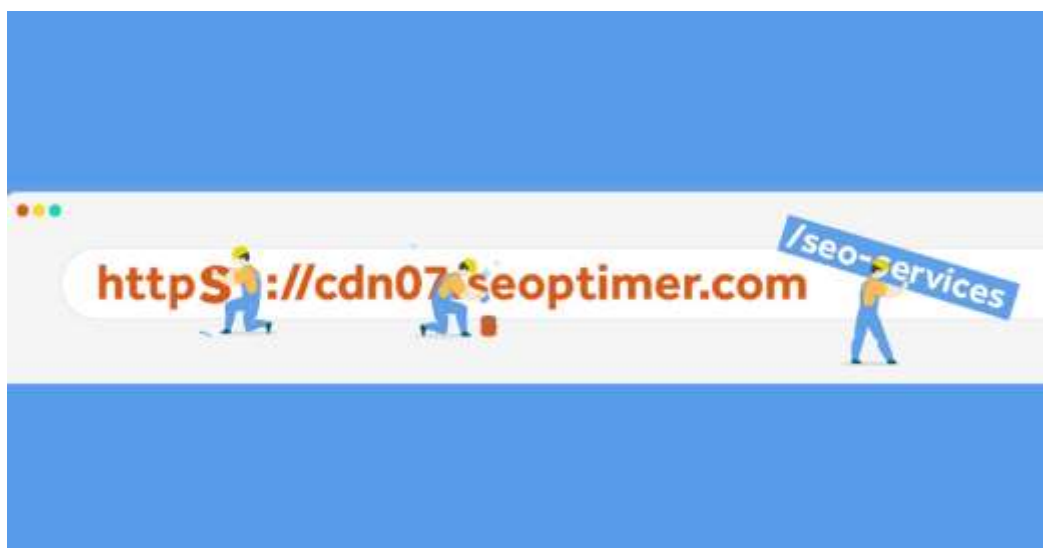
Bu maqolada sizlar bilan urls views template larni o'tamiz. Bugungi mavzu eng asosiy va eng katta mavzu bo'lib urls views va template mavzularini o'z ichiga oladi. Chunki bu mavzular bir biriga chambarchas bog'liq bo'lib bir birisiz sizga tushuntirishimning iloji yo'q.



Reja:

- [urls.py](#)
- [views.py](#)
- template
- ishga tushirish

### urls.py



[urls.py](#)

Birinchi bo'lib django projectimiz bor bo'lgan papkaga kiramiz va projectimizning ichidagi projectimiz nomi bilan bir hil nomli projectga kiramiz. Menda project nomi library edi. `library>library>urls.py` ga kiramiz.

Ushbu fayl kodi(library>library>urls.py):

```
"""library URL Configuration

The `urlpatterns` list routes URLs to views. For more information please
see:
    https://docs.djangoproject.com/en/3.1/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path

urlpatterns = [
    path('admin/', admin.site.urls),
]
```

Shu faylni kod editorda ochib olamiz va yuqori qismdagi komentaryni o'qiymiz. Komentda siz uchun kerakli bo'lgan bilimlar mavjud. Komentning pastki qismida boshqa `app`ga qanday qilib ulashni ko'rsatib o'tgan.

Siz qilishingiz kerak:

django.urls dan include funksiyasini path funksiyasi bilan birgalikda chaqirib olishingiz.

urlpatterns ning ichiga `path('', include('lib_app.urls', namespace='app'))` yozishingiz zarur.

Bundan so'ng sizning `urls.py` faylingiz:

```
"""library URL Configuration

The `urlpatterns` list routes URLs to views. For more information please
see:
    https://docs.djangoproject.com/en/3.1/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
```

```

Class-based views
1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
Including another URLconf
1. Import the include() function: from django.urls import include, path
2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('lib_app.urls', namespace='app'))
]

```

Siz bu amalingiz bilan include funksiyasini chaqirdingiz va `lib_app` papkadagi `urls.py` faylini ishga tushirdingiz. Hozirida esa bunday fayl mavjud emas chunki bu faylni o'zimiz hosil qilib olamiz. Qachonki domendan so'ng yoki localhostdan so'ng hechqanday manzil bo'lmasa `urls.py` faylimiz `lib_app` papkadagi `urls.py` ni ishga tushiradi.

`lib_app` papkada `urls.py` faylini hosil qilib olamiz. Bu faylni hohlasangiz hosil qilib olib so'ngra library papkasidagi `urls.py` ichidagi kodni nusxalab olib o'tishingiz ham mumkin yoki `urls.p` faylini o'zini nusxalab qo'yning.

So'ngra `app` ning ichidagi `urls.py` ning bazi qismlarni o'chirib yuboramiz. `urlpatterns` listini tozalab olamiz.

`from django.contrib import admin` ni o'chirib yuboramiz chunki asosiy `urls.py` faylimizda chaqirib qo'yganmiz. Bu faylda esa kerak emas.

Bundan so'ng bizning `app` mizning `urls.py` faylimiz manabunday holda bo'lishi kerak:

```

"""Library URL Configuration

The `urlpatterns` list routes URLs to views. For more information please
see:
    https://docs.djangoproject.com/en/3.1/topics/http/urls/
Examples:
Function views
1. Add an import: from my_app import views
2. Add a URL to urlpatterns: path('', views.home, name='home')
Class-based views
1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
Including another URLconf
1. Import the include() function: from django.urls import include, path
2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
"""
from django.urls import path

urlpatterns = [
]

```

Endi o'zgartirishlar kiritamiz:

- `app` mizning nomi yozib olamiz `app_name='lib_app'`
- `urls.py` ning yuqorisidagi komentda yozib qo'ygan qanday qilib `views` ga ulashni o'qib chiqing
- komentda yozib qo'ygan `from . import views` yozing. Bu bilan siz `views.py` ni chaqirib olgan bo'lasiz
- `urlpatterns` ning ichiga `path` funksiyasini ishlatamiz. `path` funksiyasining ichiga birinchi bo'lib linkni yozamiz, siz qaysi link bo'lganda bu funksiya ishlasin? Agar asosiy ya'ni home sahifasini linkini yozmoqchi bo'lsangiz qo'shtirnoqlarni bo'sh qoldiring
- ikkinchi elementiga siz `views`dan funksiyani yozasiz men `views.home` funksiyasini yozdim
- uchunchi elementiga bu qanday sahifa ekanligini yozamiz
- ya'ni `name='home'` =>

```
path('', views.home, name='home')
```

appdagi `urls.py` ning kodi:

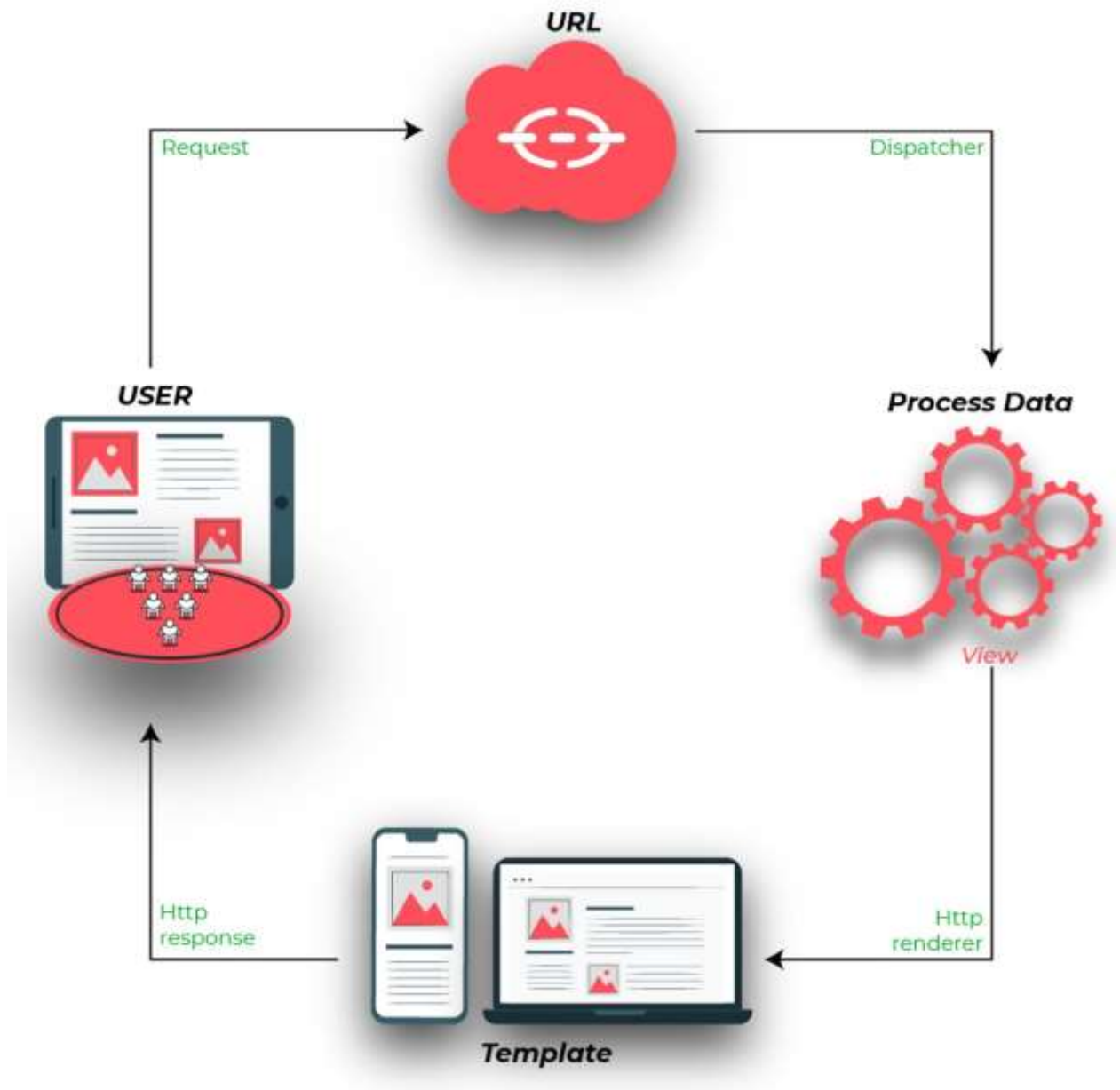
```
"""library URL Configuration

The `urlpatterns` list routes URLs to views. For more information please
see:
    https://docs.djangoproject.com/en/3.1/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.urls import path
from . import views
app_name = 'lib_app'
urlpatterns = [
    path('', views.home, name='home')
]
```

Siz bu kodlaringiz bilan:

- shu katalogdagi `views` nomli faylni chaqirib oldingiz
- `app` ning nomini kiritdingiz. Bu `urls.py` yanada yaxshi ishlashini taminlaydi. `app_name` djangoning yangi versiyalarida keng qo'llanilmoqda
- `path` funksiyasida bo'sh qolgan qo'shtirnoqlar bu asosiy sahifa ishga tushirganda ya'ni domendan so'ng boshqa manzil bo'lmasa `views`dagi `home` funksiyasi ishga tushadi
- `name='home'` deganimizning sababi keyinchalik asosiy sahifaga o'tayotganimizda giperlink dan foydalanayotganingizda bo'sh joy qoldirsangiz asosiy sahifaga o'tmaydi o'sha paytda maxsus funksiya ishtirokida `"home"` deb yozsangiz ishlaydi

## views.py



## views.py

appmizning `views.py` fayliga kiramiz

kodi:

```
from django.shortcuts import render
# Create your views here.
```

`views` faylimizda `render` funksiyasi avvaldan chaqirilgan bo'ladi. Chunki biz `render` funksiyadan foydalanamiz. Django iloji boricha dasturchilarga yordam berishga urinadi.

appmizning `urls.py` faylida `views` dagi `home` funksiya ishga tushsiz degan edik. Endilikda esa bu funktsiyani hosil qilishimiz kerak.

Buning uchun `views.py` fayliga:

`home` nomli funksiya yaratib olamiz. Bu funksiya `request` qabul qiladi

`render` funksiyasini qaytaramiz

`render` funksiyasining ichiga birinchi bo'lib `request` keyin ikkinchi elementiga qaysi web sahifa('index.html') ko'rinishini yozamiz.

Bundan so'ng `views` faylimiz:

```
from django.shortcuts import render
# Create your views here.
def home(request):
    return render(request, 'index.html')
```

Qanday `index.html`?! deb o'ylanyapsizmi.

## Template



### Template

Avvalgi darslarda template mavzusini o'tamiz degan edim.

`template` hosil qilish:

- Buning uchun `settings.py` ga kiring.
- yuqori qisda os kutubxonasini chaqirib olish `import os`
- `TEMPLATES` ga tushung va `DIRS` nomli list ichiga yozing.

```
os.path.join(BASE_DIR, 'templates')
```

- `lib_app` library papkalarini yoniga templates nomli papka hosil qiling.

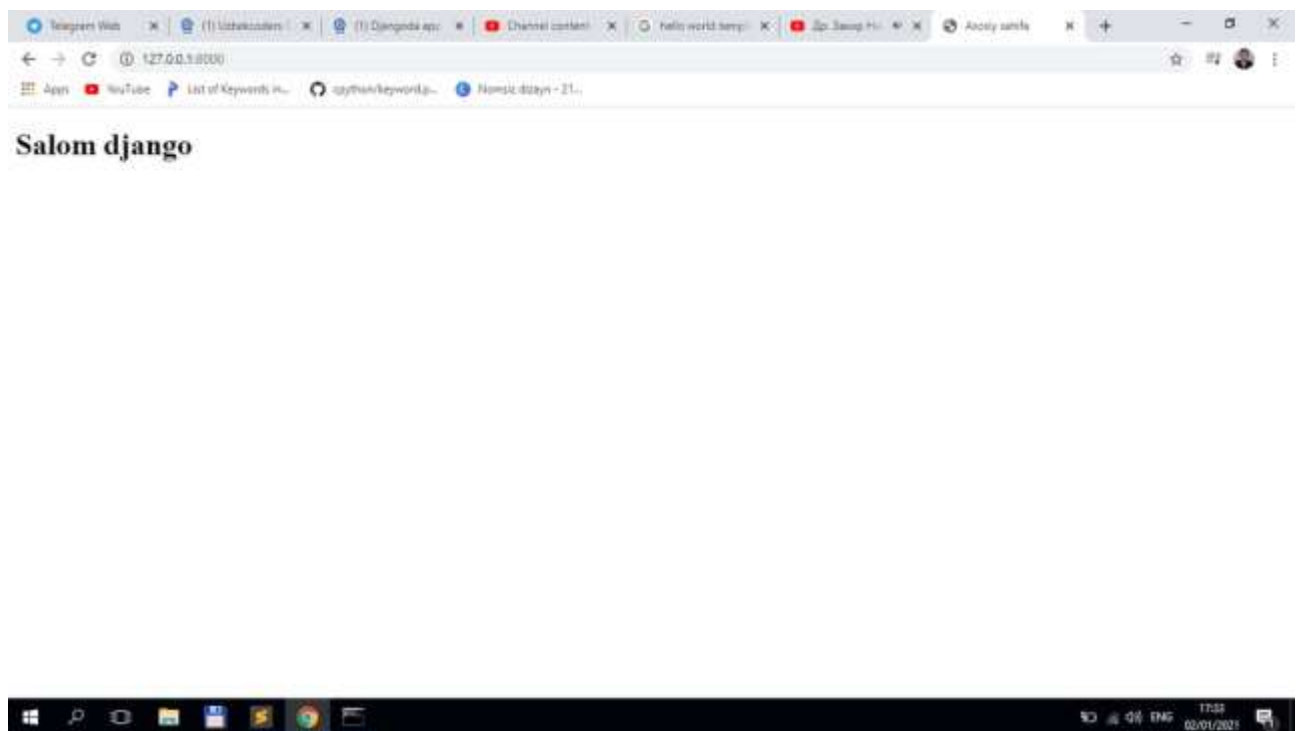
Istasangiz `DIRS` listini ichidagi `'templates'` ni o'rniga boshqa so'z yozing va o'sha nomdagi papka hosil qiling ammo yozilmagan qoidalarga ko'ra teplate papkaning nomi templates bo'lishi kerak. Bu sizning kodingizni boshqa dasturchilar ishlatayotganda osonroq bo'ladi.

Bu templates papkani ichiga `index.html` fayl hosil qiling

`index.html` ichiga kod yozing. Masalan:

```
<!DOCTYPE html>
<html>
<head>
<title>Asosiy sahifa</title>
</head>
<body>
<h1>Salom django</h1>
</body>
</html>
```

Bundan so'ng runserver qiling(serverni ishga tushiring).



runserverdan so'ng

**Xulosa:** Bu maqolada djangoning asosiy qismini aytib o'tdik. Sizlar [urls.py](https://urls.py) dagi linkni o'zgartirib ko'ring ya'nada chuqurroq bilib olasiz. Raxmat hammaga. Keyingi darslarda ko'rishguncha !!!

**Maqola muallifi: Bunyodbek Abdurazzoqov**