

Estrutura básica de um To-Do List

Objetivo

Criar uma página HTML simples usando Django, que simule uma lista de tarefas.

Passo a passo

Passo 1: Criar o projeto e a aplicação

1. Inicie um novo projeto Django:

```
django-admin startproject config .
```

2. Crie uma aplicação chamada `tarefas`:

```
python manage.py startapp tarefas
```

3. Configure a aplicação no `settings.py`:

- Adicione '`tarefas`' em `INSTALLED_APPS`:

```
INSTALLED_APPS = [  
    ...  
    'tarefas',  
]
```

Passo 2: Criando uma view simples

- 1. No arquivo `views.py` da aplicação `tarefas`, crie uma view que renderiza um template com uma lista de tarefas fixa:**

```
from django.shortcuts import render

def lista_tarefas(request):
    tarefas = ["Estudar Django", "Criar um projeto", "Revisar o
conteúdo"]
    return render(request, 'lista_tarefas.html', {'tarefas':
tarefas})
```

- Aqui, estamos criando uma lista fixa de tarefas apenas para exibir na página, sem armazenar em um banco de dados.

Passo 3: Configurando a URL

1. **Crie o arquivo `urls.py`** dentro da aplicação `tarefas` e adicione o caminho para a view:

```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.lista_tarefas, name='lista_tarefas'),
]
```

2. **Inclua o `urls.py` da aplicação `tarefas` no `urls.py` do projeto:**

- No arquivo `urls.py` do projeto `lista_de_tarefas`, adicione:

```
from django.contrib import admin
from django.urls import path, include

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

]

Passo 4: Criando um template simples

1. Crie a pasta de templates para a aplicação `tarefas`:

- Dentro da pasta `tarefas`, crie a pasta `templates`.

2. Crie o arquivo `lista_tarefas.html` dentro de `tarefas/templates/`:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Lista de Tarefas</title>
</head>
<body>
    <h1>Minha Lista de Tarefas</h1>
    <ul>
        {% for tarefa in tarefas %}
            <li>{{ tarefa }}</li>
        {% endfor %}
    </ul>
</body>
</html>
```

- Esse template exibe uma lista de tarefas simples em formato de lista não ordenada (ul).

Passo 5: Testando o projeto

1. Inicie o servidor de desenvolvimento:

```
python manage.py runserver
```

2. **Acesse** <http://127.0.0.1:8000/> no navegador.

- Você deve ver a lista de tarefas fixa: "Estudar Django", "Criar um projeto", "Revisar o conteúdo".

Próximos passos

Na próxima aula:

- Adicionar um modelo **Tarefa** para armazenar as tarefas no banco de dados.
- Ajustar a view para buscar tarefas do banco de dados.
- Criar um formulário para adicionar novas tarefas à lista.