#####################################################

# AULA 01

Site: https://www.youtube.com/watch?v=UmljXZIypDc&list=PL-osiE80TeTtoQCKZ03TU5fNfx2UY6U4p

#####################################################

O que fez:

Inicia venv:

Instala django:

Inicia projeto django:

Roda o servidor: python manage.py runserver

#####################################################

# AULA 02

https://www.youtube.com/watch?v=a48xeeo5Vnk&list=PL-osiE80TeTtoQCKZ03TU5fNfx2UY6U4p&index=2

#####################################################

O que fez:

Cria APP: python manage.py startapp blog

Criou pasta: blog com arquivos (views.py)

------------------------------------

projeto\_django / **blog** / **views**.py

------------------------------------

from django.shortcuts import render

from django.http import HttpResponse

# Create your views here.

def **home**(request):

return HttpResponse('<h1>Blog Home</h1>')

------------------------------------

projeto\_django / **blog** -> **urls**.py

------------------------------------

from django.urls import path

from . import views

urlpatterns = [

    path('', views.home, name='blog-home'),

    path('about/', views.about, name='blog-about'),

]

Observação 1: o ponto do from . import significa o diretório atual.

Observação 2: **views.home** significa arquivo **views**.py, função **home**; **views.about** significa arquivo **views**.py, função **about**.

Criamos o arquivo urls.py dentro de projeto\_django/projeto\_django

------------------------------------------

projeto\_django / **projeto\_django** -> urls.py

------------------------------------------

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path('blog/', include('blog.urls')),

]

Está funcionando o site: http://localhost:8000/blog/

A linha:

path('blog/', include('blog.urls')),

quer dizer que sempre que a pessoa entra no domínio blog/ (digita http://localhost:8000/blog/), ela é encaminhada para blog.urls

Mudando a última linha de urlpatterns para blog\_dev, agora acessamos com:

http://localhost:8000/blog\_dev/

http://localhost:8000/blog\_dev/about

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path('blog\_dev/', include('blog.urls')),

]

Para definir a página de http://localhost:8000/

Basta deixar a string do primeiro argumento de path vazia, como:

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path('', include('blog.urls')),

]

#####################################################

# AULA 03 - Templates

https://www.youtube.com/watch?v=qDwdMDQ8oX4&list=PL-osiE80TeTtoQCKZ03TU5fNfx2UY6U4p&index=3

#####################################################

Teplates

Criamos a pasta templates dentro de projeto\_django/blog:

------------------------------------------

projeto\_django / blog / **templates**

------------------------------------------

Dentro de projeto\_django/blog/templates, criamos uma subpasta com o mesmo nome do app(blog) - assim também é feito com a pasta static, abaixo:

------------------------------------------

projeto\_django / blog / **templates** / **blog**

------------------------------------------

Lá, criamos dois arquivos:

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **home.html**

------------------------------------------

e:

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **about.html**

------------------------------------------

No arquivo apps.py (projeto\_django / blog), há a classe:

class BlogConfig(AppConfig):

    name = 'blog'

------------------------------------------

projeto\_django / projeto\_django / **settings.py**

------------------------------------------

Adicionamos uma linha de string à lista INSTALLED\_APPS no arquivo settings.py (projeto\_django / blog)

INSTALLED\_APPS = [

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

]

ficando:

INSTALLED\_APPS = [

    'blog.apps.BlogConfig',

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

]

Da primeira linha (adicionada: '**blog**.**apps**.**BlogConfig**'), **blog** é a **pasta**, **apps** é o **arquivo** (apps.py), e **BlogConfig** é a **classe** (BlogConfig()).

------------------------------------------

projeto\_django / blog / **views.py**

------------------------------------------

Neste arquivo,

Trocamos:

def home(request):

    return HttpResponse('<h1>Blog Home</h1>')

def about(request):

    return HttpResponse('<h1>Blog About</h1>')

Por:

def home(request):

    return render(request, 'blog/home.html')

def about(request):

    return HttpResponse('<h1>Blog About</h1>')

Das funções, que também retornam respostas http, **request** é objeto, e:

**'blog/home.html'** é o arquivo template que está dentro de **projeto\_django/blog/templates/blog/home.html**

**'blog/about.html'** é o arquivo template que está dentro de **projeto\_django/blog/templates/blog/about.html**

Podemos excluir a linha:

from django.http import HttpResponse

Criamos também em views.py, dados falsos em uma lista com dicionários:

posts = [

    {

        'autor': 'CoreyMS',

        'título': 'Primeiro título',

        'conteúdo': 'Primeiro conteúdo',

        'postado\_em': '27 de agosto de 2018'

    },

    {

        'autor': 'PedroMFN',

        'título': 'Segundo título',

        'conteúdo': 'Segundo conteúdo',

        'postado\_em': '11 de janeiro de 2021'

    }

]

E passamos essa lista para o template blog/home.html, modificando de:

def home(request):

    return render(request, 'blog/home.html')

Para (criamos um dicionário dentro da função home, com a chave chave\_posts; e passamos esse dicionário como 3º argumento da função render):

def home(request):

    context = {

        'chave\_posts': posts

    }

    return render(request, 'blog/home.html', context)

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **home.html**

------------------------------------------

<body>

    <h1>Blog home!!!!</h1>

    {% for post in chave\_posts %}

        <h2>{{ post.título }}</h2>

        <p>Por {{ post.autor }} em {{ post.postado\_em }}</p>

        <p>{{ post.conteúdo }}</p>

    {% endfor %}

</body>

------------------------------------------

projeto\_django / blog / **views.py**

------------------------------------------

Mandamos um dicionário ({'título':'Sobre'}) para a template about:

def home(request):

    dicionário = {

        'chave\_posts': posts

    }

    return render(request, 'blog/home.html', dicionário)

def about(request):

    return render(request, 'blog/about.html', {'título':'Sobre'})

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **home.html**

projeto\_django / blog / **templates** / **blog** / **about.html**

------------------------------------------

Deixamos o título da página dinâmico com:

<head>

    <meta charset="UTF-8">

    {% if título %}

        <title>Meu blog - {{ título }}</title>

    {% else %}

        <title>Meu blog</title>

    {% endif %}

    <title>Document</title>

</head>

## Herança de templates

Usado quando há repetição de código HTML, definindo-se um único:

Cria-se dentro de projeto\_django/blog/templates/blog o arquivo (template) base.html

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **base.html**

------------------------------------------

E colocamos o cabeçalho repetido dos templates home.html e about.html em base.html

<!DOCTYPE html>

<html lang="pt-br">

<head>

    <meta charset="UTF-8">

    {% if título %}

        <title>Meu blog - {{ título }}</title>

    {% else %}

        <title>Meu blog</title>

    {% endif %}

    <title>Document</title>

</head>

<body>

    {% block conteúdo %}{% endblock %}

</body>

</html>

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **home.html**

Exclui tudo que é repetido.

{% extends "blog/base.html" %}

{% block conteúdo %}

    <h1>Blog home!!!!</h1>

    {% for post in chave\_posts %}

        <h2>{{ post.título }}</h2>

        <p>Por {{ post.autor }} em {{ post.postado\_em }}</p>

        <p>{{ post.conteúdo }}</p>

    {% endfor %}

{% endblock conteúdo %}

projeto\_django / blog / **templates** / **blog** / **about.html**

------------------------------------------

Exclui tudo que é repetido.

{% extends "blog/base.html" %}

{% block conteúdo %}

    <h1>Blog about!!!!</h1>

{% endblock conteúdo %}

26:25

## Colocando o Bootstrap

https://getbootstrap.com/docs/4.0/getting-started/introduction/#starter-template

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **base.html**

------------------------------------------

Colocar no Head:

 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

Colocar a linha

{% block bloco\_conteúdo %}{% endblock %}

dentro de uma div de classe "container", como;

    <div class="container">

        {% block bloco\_conteúdo %}{% endblock %}

    </div>

Colocar no body:

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl" crossorigin="anonymous"></script>

### Coloca code snippets

<body>

    <!-- BARRA DE NAVEGAÇÃO -->

    <header class="site-header">

        <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">

          <div class="container">

            <a class="navbar-brand mr-4" href="/">Django Blog</a>

            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarToggle" aria-controls="navbarToggle" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

            </button>

            <div class="collapse navbar-collapse" id="navbarToggle">

              <div class="navbar-nav mr-auto">

                <a class="nav-item nav-link" href="/">Home</a>

                <a class="nav-item nav-link" href="/about">About</a>

              </div>

              <!-- Navbar Right Side -->

              <div class="navbar-nav">

                <a class="nav-item nav-link" href="#">Login</a>

                <a class="nav-item nav-link" href="#">Register</a>

              </div>

            </div>

          </div>

        </nav>

      </header>

    <!-- CONTEÚDO -->

    <!-- Aqui, os templates filhos vão sobrescrever -->

    <!-- <div class="container">

        {% block bloco\_conteúdo %}{% endblock %}

    </div> -->

    <main role="main" class="container">

        <div class="row">

          <div class="col-md-8">

            {% block bloco\_conteúdo %}{% endblock %}

          </div>

          <div class="col-md-4">

            <div class="content-section">

              <h3>Our Sidebar</h3>

              <p class='text-muted'>You can put any information here you'd like.

                <ul class="list-group">

                  <li class="list-group-item list-group-item-light">Latest Posts</li>

                  <li class="list-group-item list-group-item-light">Announcements</li>

                  <li class="list-group-item list-group-item-light">Calendars</li>

                  <li class="list-group-item list-group-item-light">etc</li>

                </ul>

              </p>

            </div>

          </div>

        </div>

      </main>

## Colocando arquivos CSS

Cria a pasta static em projeto\_django/blog

------------------------------------------

projeto\_django / blog / **static**

------------------------------------------

Assim como no template, criar uma subpasta com o mesmo nome do app (blog)

------------------------------------------

projeto\_django / blog / **static** / **blog**

------------------------------------------

dentro de projeto\_django/blog/static/blog, criamos um arquivo css chamado main.css

------------------------------------------

projeto\_django / blog / **static** / **blog** / **main.css**

------------------------------------------

Lá colocamos os snippets(https://github.com/CoreyMSchafer/code\_snippets/blob/master/Django\_Blog/snippets/main.css).

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **base.html**

------------------------------------------

Chamamos o arquivo main.css no template base.html, Procedimento:

1) No topo:

{% load static %}

2) No <head>:

    <link rel="stylesheet" href="{% static 'blog/main.css' %}">

Reinicializar o servidor:

python manage.py runserver

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **base.html**

------------------------------------------

Colocar o snippets de article.html (https://github.com/CoreyMSchafer/code\_snippets/blob/master/Django\_Blog/snippets/article.html).

no template home.html.

## DEIXANDO AS ROTAS FLEXÍVEIS EM base.html

Usamos as rotas de:

------------------------------------------

projeto\_django / blog / **urls.py**

------------------------------------------

urlpatterns = [

    path('', views.home, name='blog-home'),

    path('about/', views.about, name='blog-about'),

]

em:

------------------------------------------

projeto\_django / blog / **templates** / **blog** / **base.html**

------------------------------------------

As hrefs:

href="/"

href="/"

href="/about"

Viram:

href="{% url 'blog-home' %}"

href="{% url 'blog-home' %}"

href="{% url 'blog-about' %}"

Substituir os href="..." no base.html

De:

    <!-- BARRA DE NAVEGAÇÃO -->

    <header class="site-header">

        <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">

          <div class="container">

            <a class="navbar-brand mr-4" href="/">Django Blog</a>

            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarToggle" aria-controls="navbarToggle" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

            </button>

            <div class="collapse navbar-collapse" id="navbarToggle">

              <div class="navbar-nav mr-auto">

                <a class="nav-item nav-link" href="/">Home</a>

                <a class="nav-item nav-link" href="/about">About</a>

              </div>

              <!-- Navbar Right Side -->

              <div class="navbar-nav">

                <a class="nav-item nav-link" href="#">Login</a>

                <a class="nav-item nav-link" href="#">Register</a>

              </div>

            </div>

          </div>

        </nav>

      </header>

Para:

<!-- BARRA DE NAVEGAÇÃO -->

    <header class="site-header">

        <nav class="navbar navbar-expand-md navbar-dark bg-steel fixed-top">

          <div class="container">

            <a class="navbar-brand mr-4" href="{% url 'blog-home' %}">Django Blog</a>

            <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarToggle" aria-controls="navbarToggle" aria-expanded="false" aria-label="Toggle navigation">

            <span class="navbar-toggler-icon"></span>

            </button>

            <div class="collapse navbar-collapse" id="navbarToggle">

              <div class="navbar-nav mr-auto">

                <a class="nav-item nav-link" href="{% url 'blog-home' %}">Home</a>

                <a class="nav-item nav-link" href="{% url 'blog-about' %}">About</a>

              </div>

              <!-- Navbar Right Side -->

              <div class="navbar-nav">

                <a class="nav-item nav-link" href="#">Login</a>

                <a class="nav-item nav-link" href="#">Register</a>

              </div>

            </div>

          </div>

        </nav>

      </header>

34:20