

Elias Santos

Aula: Implementando o base.html para usar duas configurações diferentes.

Vamos observar novamente o arquivo base.html:

```
base.html x
projeto > base > global > base.html > ...
1 {% include 'global/partials/head.html' %}
2 {% include 'global/partials/menu.html' %}
3 <h1>{% block texto %}{% endblock texto %}</h1>
4
5 {% include 'global/partials/paragrafo.html' %}
6 {% include 'global/partials/paragrafo.html' %}
7 </body>
8 </html>
9
```

Temos dois includes iniciais que servem para todas as páginas:

- head.html
- menu.html

Os includes finais do paragrafo.html foi um teste.

O block texto está sendo usado por todos.

Então a questão é: montar no modelo que tenha configurações específicas para cada página. Aqui faremos um exemplo com duas configurações:

- 1. um padrão que será usado no home e no contato
- 2. um padrão que será usado no blog

Começamos esse exemplo, modificando o base.html. Veja:

```
base.html x
projeto > base > global > base.html
1 {% include 'global/partials/head.html' %}
2 {% include 'global/partials/menu.html' %}
3
4 <article class="container bg-primary text-bg-primary">
5 | {% block texto %}{% endblock texto %}
6 </article>
7
8 <article class="container">
9 | <h1>{% block titulo %}{% endblock titulo %}</h1>
10 | <div style="background-color:#FBF2EF">
11 | | {% block info %}{% endblock info %}
12 | </div>
13 </article>
14
15 </body>
16 </html>
17
```

Veja que agora há duas divisões do tipo article.

Configurações diferentes onde a primeira será a utilizada para o home, e o contato

Enquanto a seguinte será usada para o blog.

Obs: Fiz um css in-line no segundo exemplo, mas o ideal é que esteja no arquivo css

```
base.html x index.html x
projeto > blog > templates > blog > index.html
1 {% extends 'global/base.html' %}
2
3 {% block titulo %}
4 Assuntos Blog
5 {% endblock titulo %}
6
7 {% block info %}
8 <ol>
9 | <li>Python</li>
10 | <li>Html</li>
11 | <li>Css</li>
12 | <li>Bootstrap</li>
13 </ol>
14 <!-- abaixo a solução definitiva para o link de artigos -->
15 <a class="btn btn-info" href="{% url 'artigos' %}">Artigo</a>
16
17 {% endblock info %}
```

Ao lado a mudança no primeiro arquivo do blog que é o index.html

Veja que mudamos também o link de artigos para uma solução mais estável.

Esta url precisa ter sido definida antes no arquivo urls.py do blog, como o arquivo abaixo:

```
base.html x index.html x urls.py x
projeto > blog > urls.py > ...
1 from django.urls import path
2
3 from . import views
4 urlpatterns = [
5 | path('', views.blog, name='blog'),
6 | path('artigos/', views.artigos, name='artigos'),
7 ]
```