

Retry for a live version

This page (<https://medium.com/handzon-coders/criando-um-mini-blog-com-django-1-17ef6adb5072?source=>) is currently offline. However, because the site uses Cloudflare's Always Online™ technology you can continue to surf a snapshot of the site. We will keep checking in the background and, as soon as the site comes back, you will automatically be served the live version. Always Online™ is powered by [Cloudflare](#) | [Hide this Alert](#)

Criando um Mini Blog com Django #1



Erisvaldo Correia

Follow

Sep 13, 2018 · 5 min read



Fala galera tudo bem? Ao longo dessa série vamos entender os passos básicos para criarmos uma aplicação estilo blog utilizando a linguagem Python e a framework Django.

Nessa primeira parte vamos iniciar o projeto e fazer nossa primeira configuração. Irei usar aqui o Python instalado no Windows, mas você pode utilizar o processo em qualquer sistema desde que possua o Python 3 instalado com a ferramenta **PIP** (gerenciador de pacotes Python) configurada.

O que iremos utilizar ao longo do projeto?

- Python 3.6
- Django 2
- VS Code (ou outro editor de sua preferência)
- Terminal de Comando (prompt)
- Bootstrap CSS 4

Instalando o Django.

Para instalar o Django em sua máquina usamos o seguinte comando. Use o prompt de comando no Windows ou o terminal caso esteja no Linux ou Mac:

A versão que será usada é o Django 2.0.5

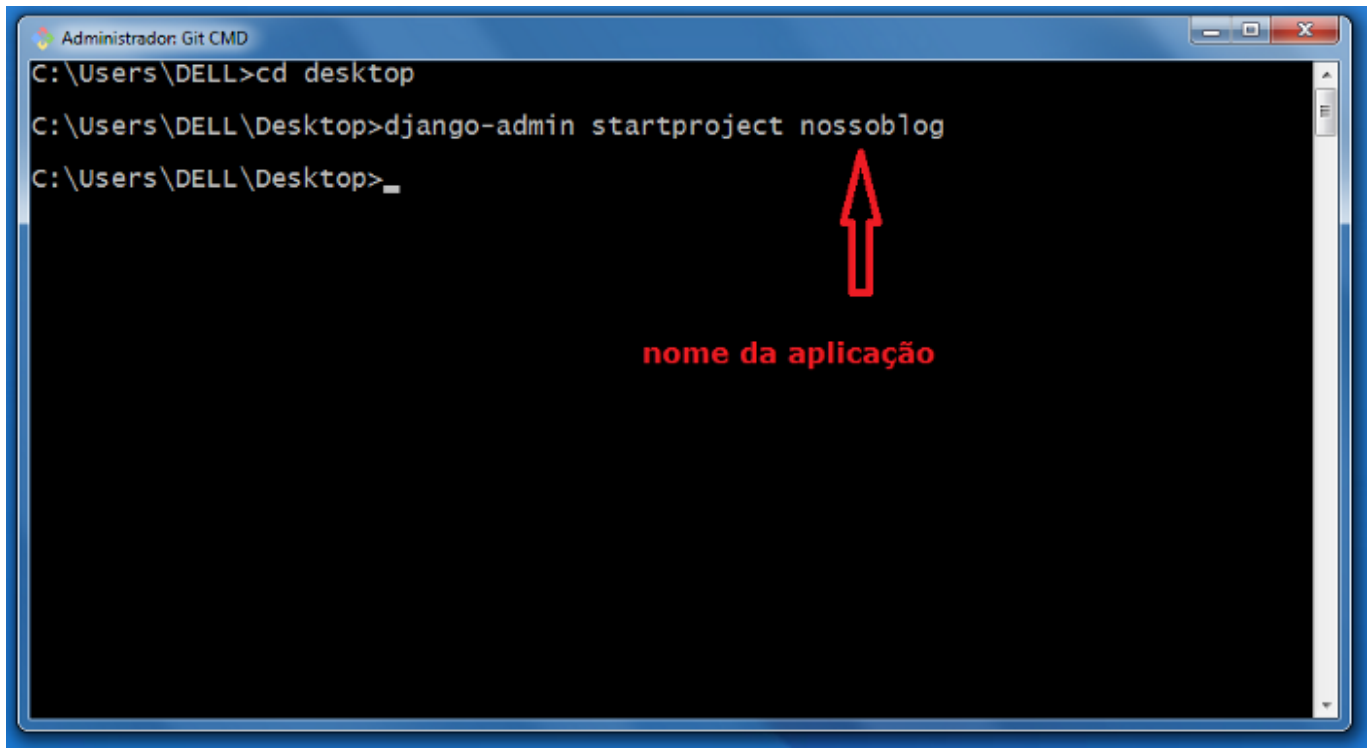
```
pip install django
```

O Django é um framework Python que nos possibilita um rápido desenvolvimento de aplicações para web. Ele trabalha no padrão de aplicações de três camadas. Quando instalamos o Django ele nos habilita um comando para que possamos criar um projeto com base em sua estrutura.

Criando nosso projeto:

Com o prompt de comando aberto, navegue até um diretório de sua preferência para criarmos nosso projeto (no meu caso irei utilizar a Área de Trabalho mesmo) e digite o seguinte comando:

```
django-admin startproject nossoblog
```



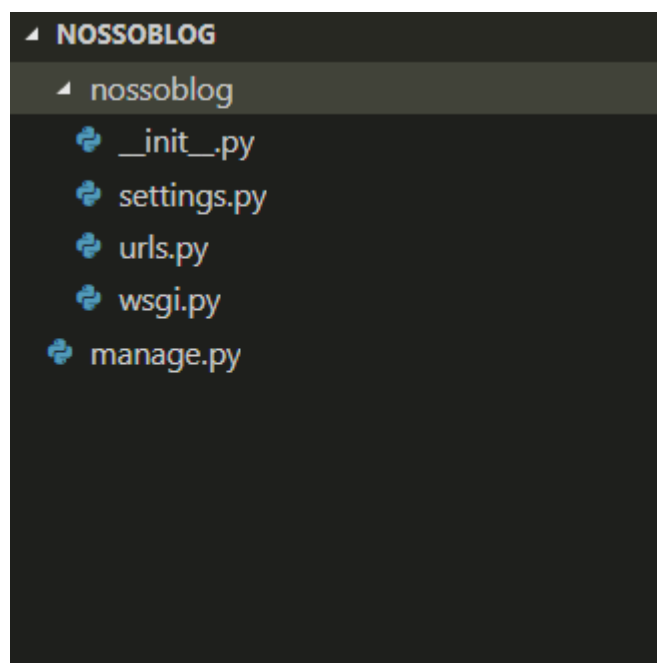
```
Administrador: Git CMD
C:\Users\DELL>cd desktop
C:\Users\DELL\Desktop>django-admin startproject nossoblog
C:\Users\DELL\Desktop>_
```

nome da aplicação

navegando para o diretório e criando o projeto

Note que usamos o comando “*django-admin*” para acessarmos o Django e informamos que queremos iniciar um novo projeto com o nome de “*nossoblog*” usando o complemento “*startproject*”.

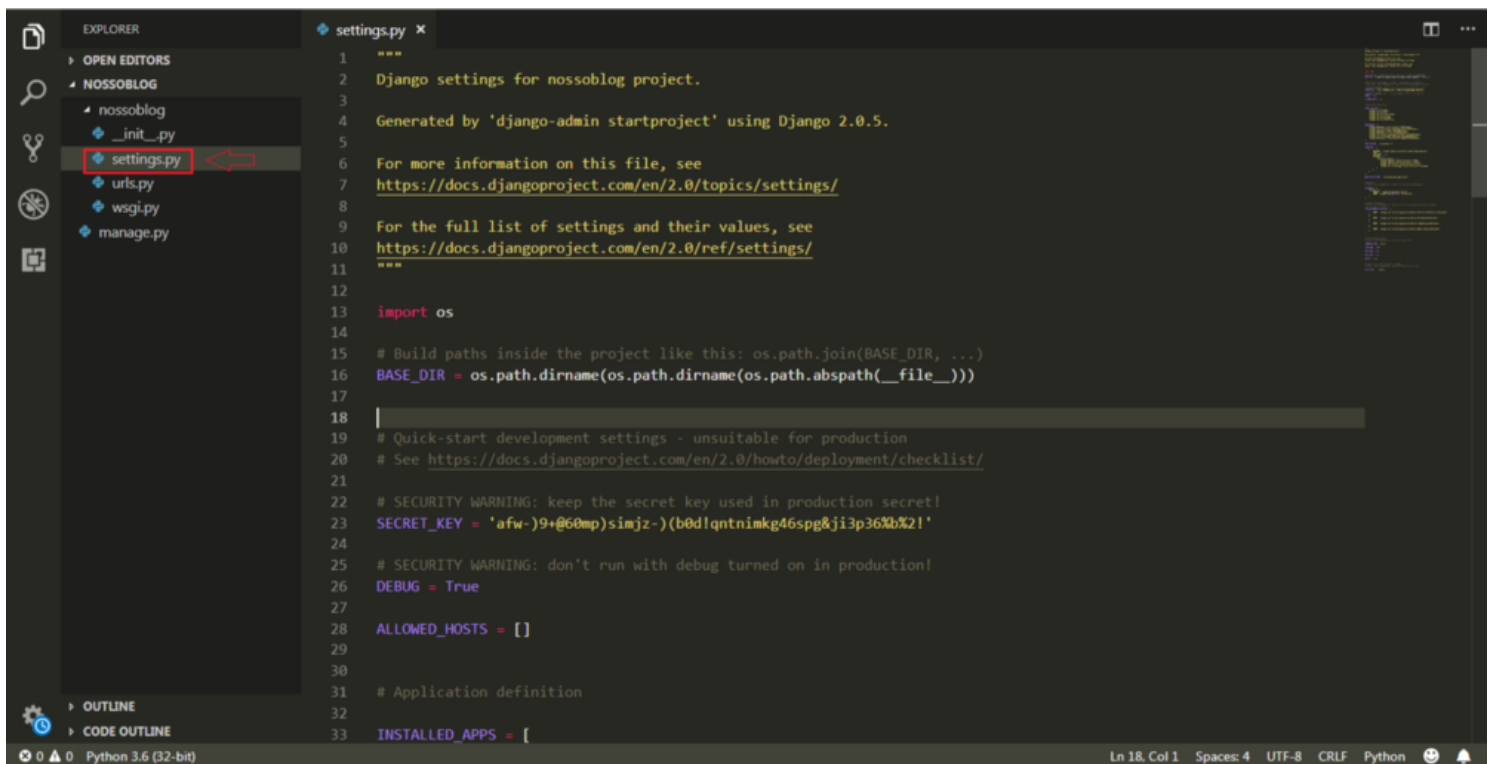
No diretório que criamos nosso projeto, o Django cria uma pasta com o nome que demos, e dentro dela encontramos um arquivo chamado “*manage.py*” e uma outra pasta com o mesmo nome do projeto. Vejamos a estrutura do projeto criado:



Estrutura criada pelo Django

Essa estrutura básica criada pelo Django nos permite construir nossas aplicações de forma rápida e simples. Dentro da pasta principal do nosso projeto como observamos encontramos um arquivo chamado “*manage.py*”. É com esse arquivo que iremos dizer ao Django a maior parte das ações a serem realizadas conforme construimos nossas aplicações. Ele é um utilitário de linha de comando que interage entre o projeto e o framework.

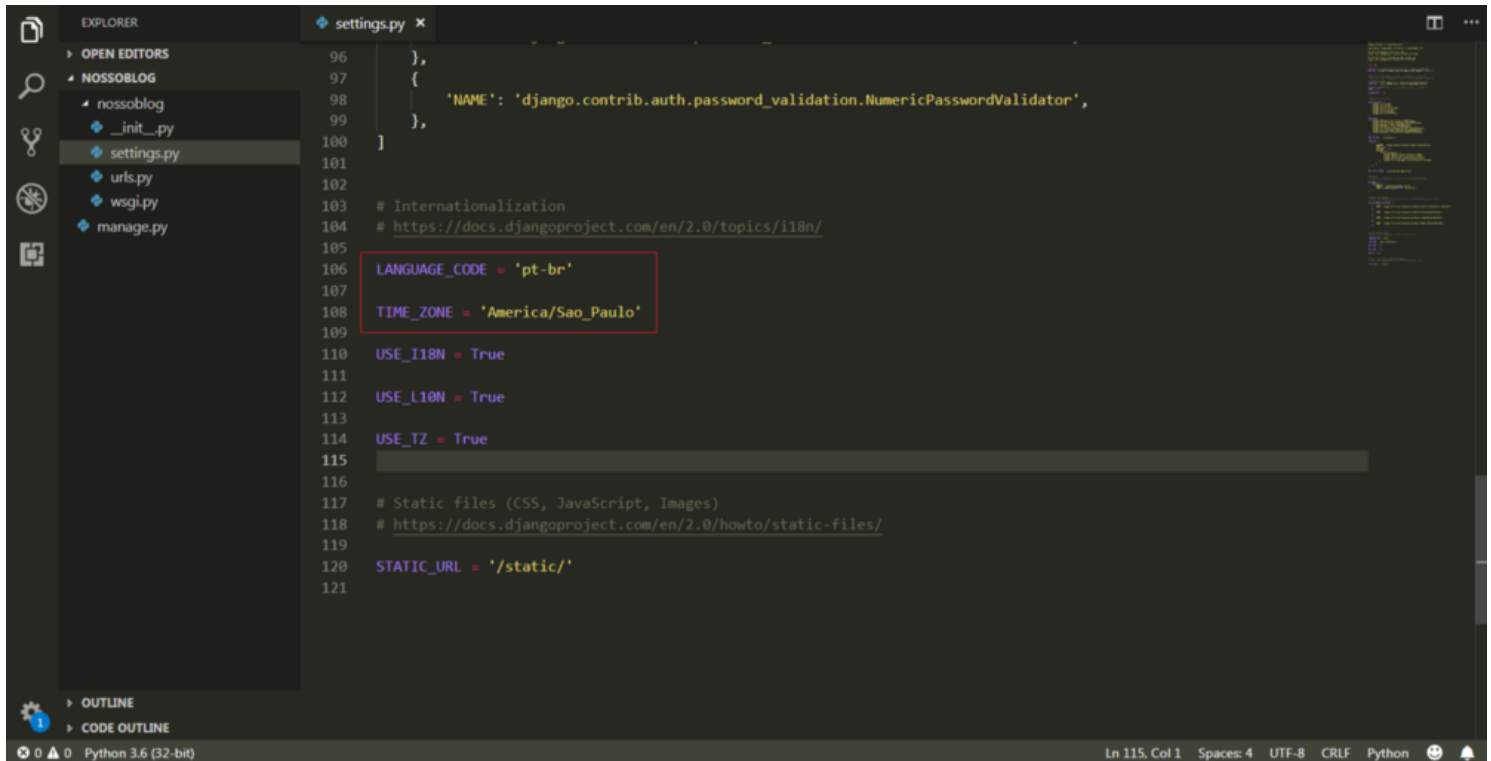
Vamos agora fazer duas configurações básicas em nosso projeto: Ajustar o idioma padrão no qual estaremos desenvolvendo e o localização de fuso horário. Para isso vamos utilizar o editor de códigos para podermos editar o arquivo “*settings.py*” que se encontra no diretório de configuração do nosso projeto.



Arquivo settings aberto no vscode

Indo até o final do código, iremos encontrar os dois pontos que precisaremos modificar por hora: vamos alterar a linguagem de ‘**en-us**’ para ‘**pt-br**’ e o localização de hora de ‘**UTC**’ para ‘**America/Sao_Paulo**’. *Obs: Você pode alterar o Time Zone para aquele que lhe for melhor.*

```
...  
LANGUAGE_CODE = 'pt-BR'  
  
TIME_ZONE = 'America/Sao_Paulo'  
...
```



Essa troca nas configurações não são obrigatórias, mas auxiliam no momento em que estamos lidando com textos e partes que usam de data e hora em nosso projeto. Feito as alterações, salve o arquivo e vamos observar o “Hello World” do Django funcionando.

Executando nosso projeto pela primeira vez!

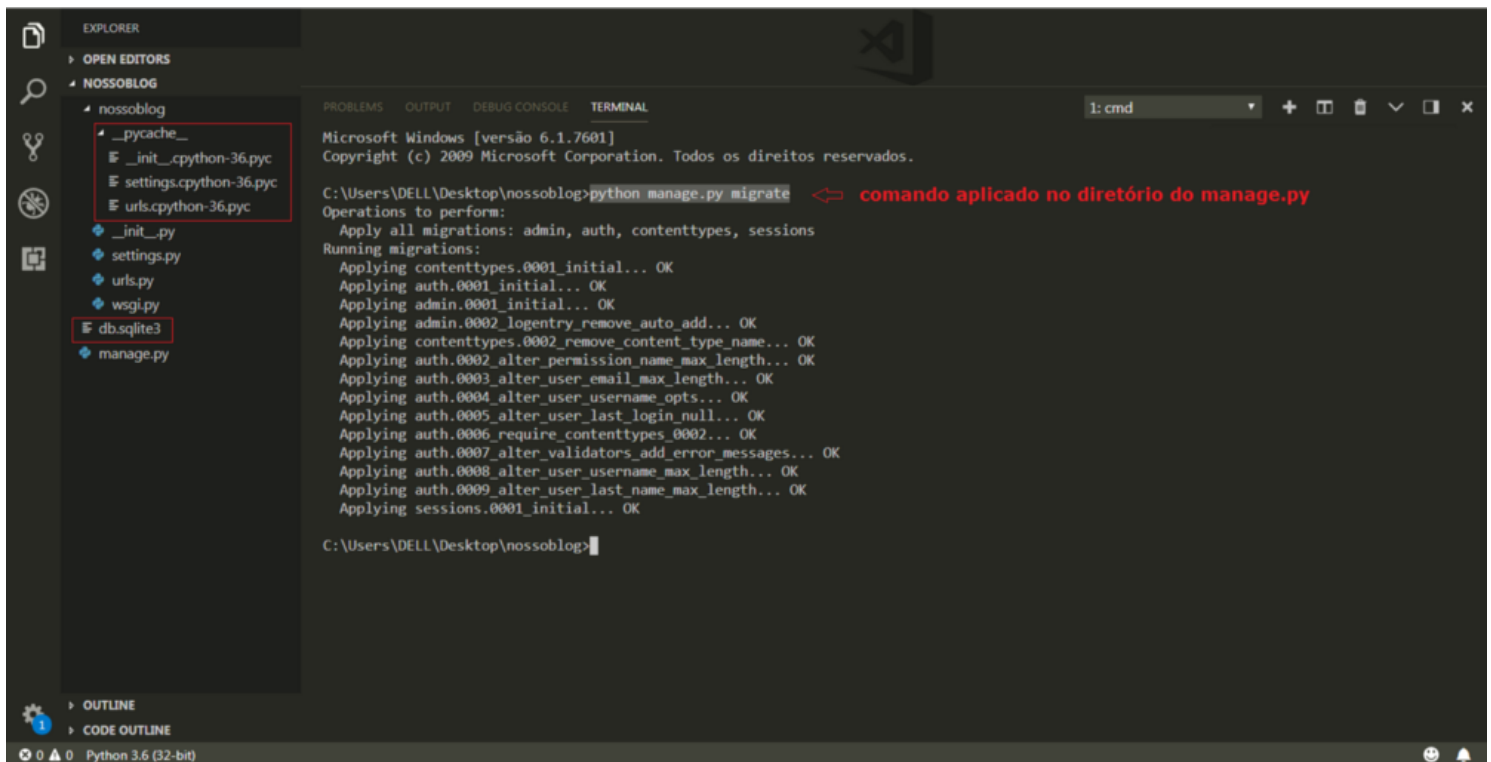
Utilize o terminal de comandos acoplado ao seu editor de código ou o prompt do sistema operacional mesmo. Navegando para dentro da pasta principal do nosso projeto ficando no diretório onde o arquivo **manage.py** se encontra. Por padrão o Django vem configurado para utilizar uma estrutura de banco de dados, a saber o **SQLite**. Você pode alterar essas configurações para trabalhar com outros bancos de dados. Por hora para esse nosso projeto iremos utilizar a estrutura padrão nos dada pelo Django.

Antes de rodarmos nossa aplicação pela primeira vez, precisamos fazer com que a estrutura básica inicial gerada pelo Django para o banco de dados seja aplicada. Para

isso em nosso terminal iremos usar o seguinte comando:

```
python manage.py migrate
```

Esse comando irá aplicar as regras da estrutura do banco de dados e criará um subdiretório chamado “__pycache__” dentro da pasta de configuração do projeto e um arquivo na raiz do nosso projeto chamado “db.sqlite3”. Esse arquivo é o que irá conter todos os dados de nosso projeto e aplicações.



```
Microsoft Windows [versão 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\DELL\Desktop\nossoblog>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying sessions.0001_initial... OK

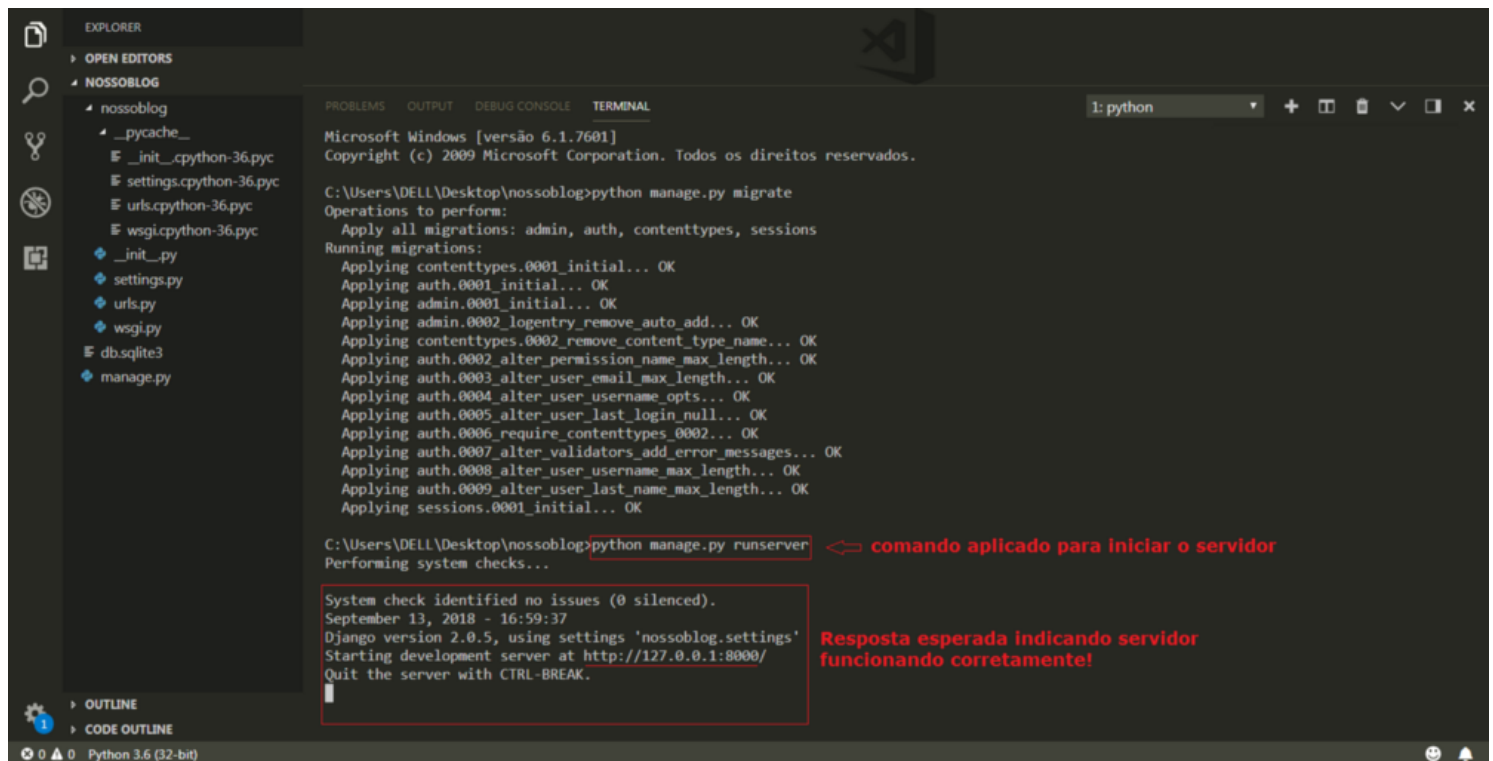
C:\Users\DELL\Desktop\nossoblog>
```

Banco de dados básico criado com as tabelas padrões do Django.

Uma vez aplicado a criação do nosso banco de dados, podemos executar o comando que inicia o servidor de desenvolvimento para visualizarmos nosso projeto. Note que até agora apenas fizemos uma simples configuração, e aplicamos poucos comandos. Todo o trabalho praticamente de construção do projeto está pela responsabilidade do Django que tem justamente essa finalidade: *facilitar nosso desenvolvimento*.

Agora ainda no diretório do arquivo **manage.py**, execute no terminal o comando a seguir:

python manage.py runserver



```
Microsoft Windows [versão 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\DELL\Desktop\nossoblog>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying sessions.0001_initial... OK

C:\Users\DELL\Desktop\nossoblog>python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
September 13, 2018 - 16:59:37
Django version 2.0.5, using settings 'nossoblog.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

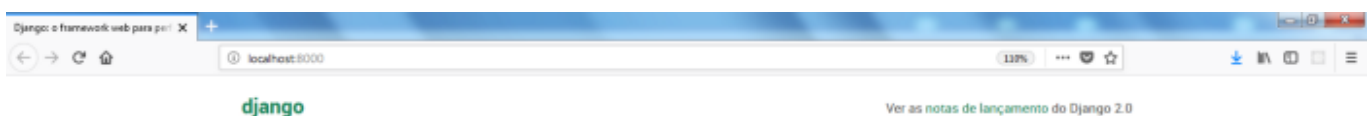
comando aplicado para iniciar o servidor

Resposta esperada indicando servidor funcionando corretamente!

servidor django iniciado com sucesso

Ele executará a verificação de todo nosso projeto e aplicações (se já tivermos alguma) e irá iniciar o servidor local para que possamos interagir pelo navegador através do endereço: <http://localhost:8000/>

A tela aberta no navegador é pré-configurada pelo Django e nos mostra que o servidor foi iniciado corretamente e que nosso projeto esta funcionando. Com ela podemos visualizar todas as funcionalidades de nossas aplicações que viermos a desenvolver. Essa primeira tela, podemos chamar de Hello World do Django. Ela aparece em português pois configuramos o arquivo *settings.py* para desenvolver nosso projeto nessa linguagem.



A instalação foi com sucesso! Parabéns!

Você está vendo esta página pois possui `DEBUG=True` no seu arquivo de configurações e não configurou nenhuma URL.



Documentação do Django
Tópicos, referências, & how-to's



Tutorial: Um aplicativo de votação
Comece a usar Django



Comunidade Django
Conecte-se, obtenha ajuda ou contribua

Nos próximos textos, vamos iniciar nossa aplicação do blog e começarmos a entender os principais conceitos de funcionalidades da framework Django!

Até o próximo texto!

Python3 Django Tutorial

About Help Legal

Get the Medium app

