



PYTHON  
ACADEMY

# SEU PRIMEIRO PROJETO DJANGO EM 15 MINUTOS

Nesse ebook você vai iniciar a sua carreira no Desenvolvimento de Aplicações Web utilizando o framework Django em apenas 15 minutos.

[PYTHONACADEMY.COM.BR](https://pythonacademy.com.br)

Este ebook foi gerado por



# Crie Ebooks técnicos incríveis em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs

Deixe que nossa IA faça o trabalho pesado

 Syntax Highlight

 Adicione Banners Promocionais

 Edite em Markdown em Tempo Real

 Infográficos feitos por IA

**TESTE AGORA** 

 PRIMEIRO CAPÍTULO 100% GRÁTIS

Salve salve Pythonista!

**Django** é um framework de desenvolvimento web em Python que permite criar aplicações robustas e escaláveis.

Se você é um desenvolvedor iniciante ou está procurando expandir suas habilidades em Python, o Django é uma ótima escolha.

Neste artigo, vamos abordar os primeiros passos para iniciar seu projeto Django, desde a instalação até a execução do servidor local.

## Instalação do Django

Antes de começar a desenvolver com Django, é necessário fazer a instalação do framework.

Felizmente, isso é um processo simples e rápido.

O Django pode ser instalado usando o pip, o gerenciador de pacotes do Python.

E é claro que você vai querer utilizar um Ambiente Virtual para isso ([clique aqui para saber mais sobre Ambientes Virtuais](#)).

Abra seu terminal e execute o seguinte comando:

```
pip install django
```

Isso irá instalar a versão mais recente do Django em seu ambiente de desenvolvimento.

# Criando um novo projeto Django

Após a instalação do Django, você está pronto para criar seu primeiro projeto.

O Django fornece um utilitário de linha de comando chamado `django-admin` para ajudar na criação de projetos.

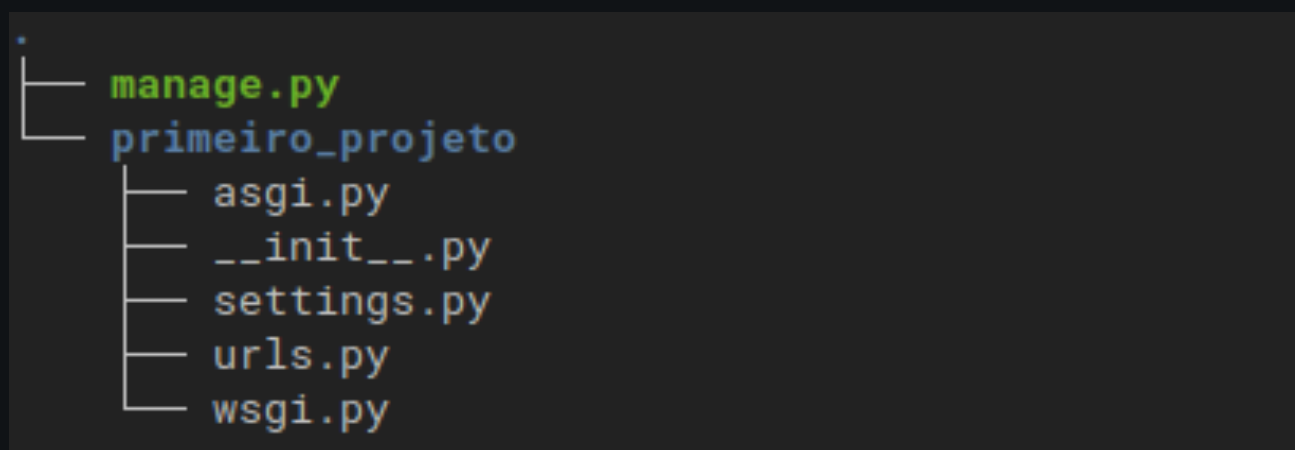
Execute o seguinte comando:

```
django-admin startproject seu_projeto
```

Isso criará um novo diretório chamado `seu_projeto` com a estrutura inicial do projeto.

Dentro dele, você encontrará um arquivo chamado `manage.py`, que será seu melhor amigo durante o desenvolvimento. Vamos falar mais sobre ele posteriormente.

Na imagem abaixo, você pode ver uma representação visual da estrutura inicial do projeto:



Agora vamos entender cada um desses arquivos:

- O diretório raiz do projeto contém o arquivo `manage.py` e um subdiretório com o mesmo nome do seu projeto.
- O subdiretório com o nome do seu projeto contém os arquivos de configuração principais.
- O arquivo `settings.py` contém as configurações do projeto, como banco de dados, aplicativos instalados, caminhos para arquivos estáticos, entre outros.
- O arquivo `urls.py` é usado para mapear as URLs da sua aplicação para as funções que manipularão as requisições.
- O arquivo `wsgi.py` é usado para iniciar o servidor WSGI (Web Server Gateway Interface) quando você deseja implantar o projeto em um servidor de produção.
- O arquivo `asgi.py` é usado para iniciar o servidor utilizando ASGI (Asynchronous Server Gateway Interface) quando você deseja implantar o projeto em um servidor de produção, utilizando servidores web assíncronos.

## O poder do `manage.py`

Agora, vamos falar um pouco mais sobre o `manage.py`.

Esse arquivo é a espinha dorsal do seu projeto Django, e você irá usá-lo com frequência, pode acreditar 👍

O `manage.py` permite executar uma variedade de comandos úteis, como migrações de banco de dados, execução de testes, criação de superusuários e muito mais.



Para usar o `manage.py`, abra seu terminal, navegue até o diretório raiz do seu projeto e execute os comandos a seguir.

## Executando o servidor local

Para testar seu projeto localmente, sem a necessidade de implantá-lo em um servidor de produção, você pode executar o servidor de desenvolvimento embutido do Django através do comando `runserver`.

Para isso, execute o seguinte em seu terminal:

```
python manage.py runserver
```

Isso iniciará o servidor na porta padrão `8000`. Você verá uma saída semelhante à seguinte:

```
Starting development server at http://127.0.0.1:8000/  
Quit the server with CONTROL-C.
```

Agora, abra seu navegador e acesse `http://127.0.0.1:8000/`.

Você verá a seguinte página de boas-vindas do Django, indicando que seu servidor está funcionando corretamente.



The install worked successfully! Congratulations!

You are seeing this page because `DEBUG=True` is in your settings file and you have not configured any URLs.



**Django Documentation**  
Topics, references, & how-to's



**Tutorial: A Polling App**  
Get started with Django



**Django Community**  
Connect, get help, or contribute

## Outros comandos úteis

Além do comando `runserver`, o `manage.py` oferece várias outras funcionalidades úteis para o desenvolvimento do seu projeto Django.

Vejamos alguns exemplos:

**Migrações:** O Django possui um sistema de migrações embutido, que permite gerenciar o estado do seu banco de dados. Para criar e aplicar migrações, execute os seguintes comandos:

```
python manage.py makemigrations
python manage.py migrate
```

Isso irá 1) criar as migrações do sistema e 2) aplicar em seu banco de dados (sem precisar de uma linha de SQL para isso!).

**Criar um superusuário:** Durante o desenvolvimento, é comum precisar de um usuário com permissões administrativas para acessar a área admin do Django.

Para criar um superusuário, execute o seguinte comando:

```
python manage.py createsuperuser
```

**Executando testes:** O Django fornece um framework de testes integrado. Você pode escrever testes para verificar o comportamento do seu aplicativo e, para executá-los, use o seguinte comando:

```
python manage.py test
```

Esses são apenas alguns exemplos dos muitos comandos que o `manage.py` oferece.



*Estou desenvolvendo o **DevBook**, uma plataforma que usa IA para gerar ebooks técnicos profissionais. Não deixe de conferir!*



## Crie Ebooks técnicos incríveis em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs

Deixe que nossa IA faça o trabalho pesado

Syntax Highlight

Adicione Banners Promocionais

Edite em Markdown em Tempo Real

Infográficos feitos por IA

TESTE AGORA! PRIMEIRO CAPÍTULO 100% GRÁTIS

## Conclusão

Neste artigo, você aprendeu sobre os primeiros passos para iniciar seu projeto Django.

Vimos como instalar o Django, criar um novo projeto e explorar a estrutura inicial.

Além disso, você foi apresentado ao poderoso `manage.py`, que permite executar uma variedade de comandos úteis.

O Django é uma ferramenta poderosa para o desenvolvimento web em Python.

Agora que você tem uma base sólida, você pode começar a desenvolver suas próprias aplicações web usando Django.

Aproveite todo o potencial deste framework e não hesite em explorar a vasta documentação disponível para descobrir ainda mais recursos.

Parabéns pelo seu primeiro projeto Django! Que você tenha sucesso em todas as suas empreitadas no mundo do desenvolvimento web.

Tamo junto!

Não se esqueça de conferir!



DevBook

# Crie Ebooks técnicos em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs



 Syntax Highlight

 Infográficos feitos por IA

 Adicione Banners Promocionais

Deixe que nossa IA faça o trabalho pesado

 Edite em Markdown em Tempo Real

**TESTE AGORA** 

 PRIMEIRO CAPÍTULO 100% GRÁTIS