



PYTHON
ACADEMY

SEU PRIMEIRO PROJETO DJANGO EM 15 MINUTOS

Tutorial completo: crie seu primeiro projeto Django 5.1 em 15 minutos.
Aprenda instalação, estrutura de arquivos, manage.py, runserver e
comandos essenciais para começar no desenvolvimento web Python.

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

Gere ebooks como este com



em <https://ebookr.ai>

Crie ebooks profissionais incríveis em minutos com IA



Chega de formatar texto no Google Docs, Word ou ferramentas que só te fazem perder tempo...

E deixe que nossa IA faça o trabalho pesado!



Capas gerados por IA



Infográficos feitos por IA



Edite em Markdown em Tempo Real



Adicione Banners Promocionais

TESTE AGORA

PRIMEIRO CAPÍTULO 100% GRÁTIS

✓ **Artigo atualizado para Django 5.1** (Dezembro 2025) Aprenda a criar projetos com a versão mais recente do Django, com suporte total a async e novas features.

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.

Versão atual (Dezembro 2025): Django 5.1 LTS - com suporte estendido até 2026.

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:

```
.
├── manage.py
└── primeiro_projeto
    ├── asgi.py
    ├── __init__.py
    ├── settings.py
    ├── urls.py
    └── wsgi.py
```

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.

💡 **Django 5.1:** ASGI é o padrão recomendado para novos projetos. Permite views assíncronas (`async def`), WebSockets e melhor performance em operações I/O.

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:

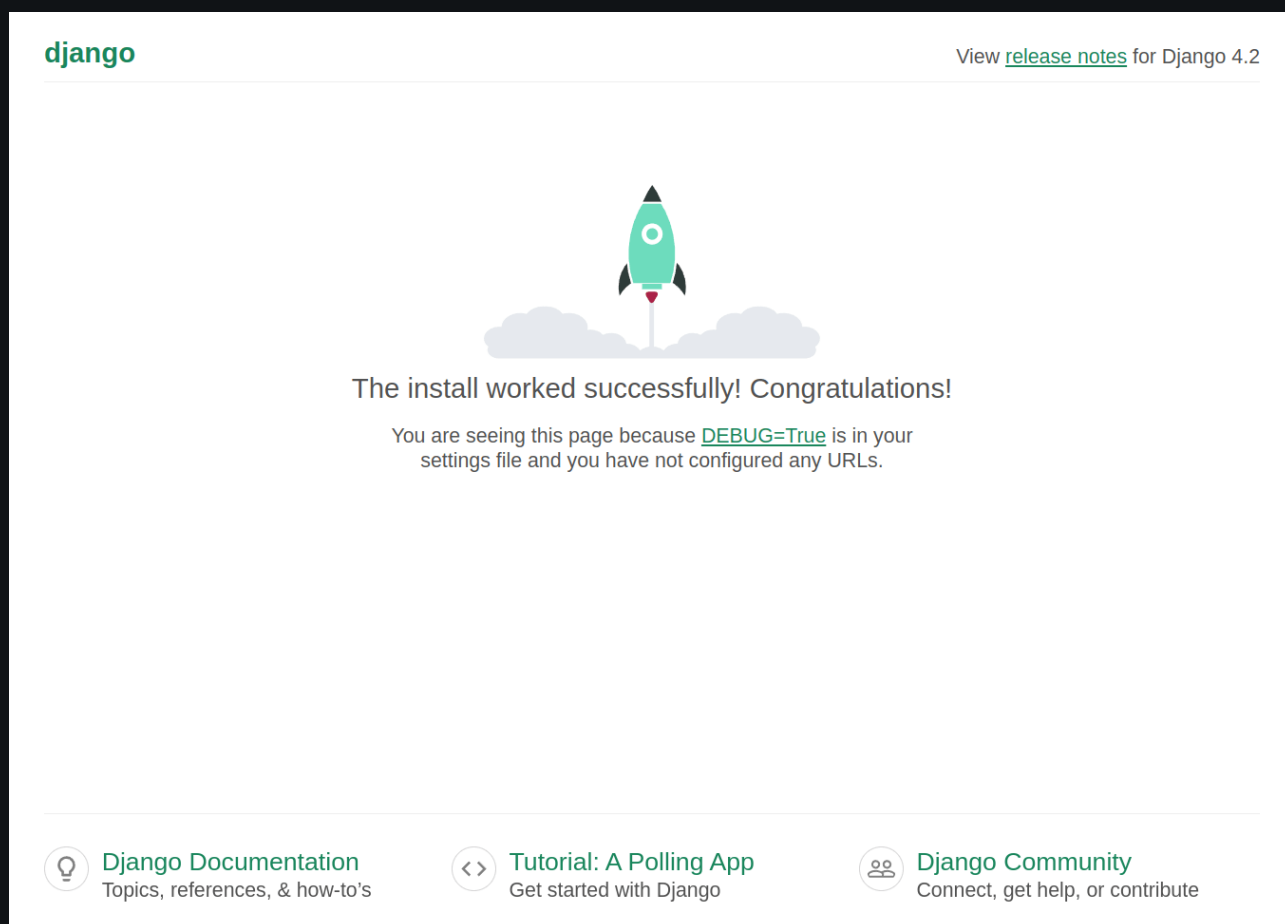
```
python manage.py runserver
```

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.



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 construindo o **Ebookr.ai**, uma plataforma onde você cria ebooks profissionais com IA sobre qualquer assunto — do zero ao PDF pronto, com capas e infográficos gerados automaticamente. Dá uma olhada!



Crie Ebooks profissionais incríveis em minutos com IA



Chega de formatar texto no Google Docs, Word ou ferramentas que só te fazem perder tempo...

... e deixe que nossa IA faça o trabalho pesado!

TESTE AGORA! PRIMEIRO CAPÍTULO 100% GRÁTIS 

 Capas gerados por IA

 Adicione Banners Promocionais

 Edite em Markdown em Tempo Real

 Infográficos feitos por IA

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.

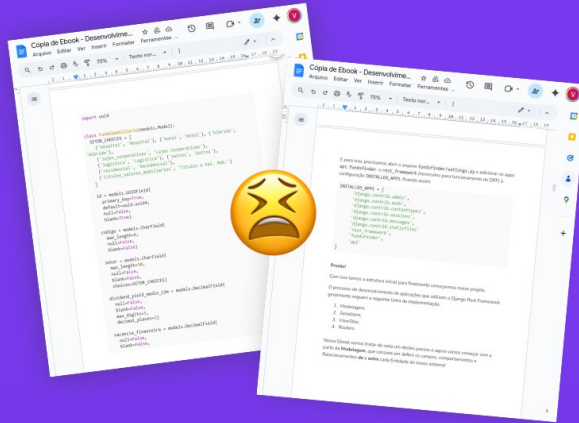
Tamo junto!

Não se esqueça de conferir!

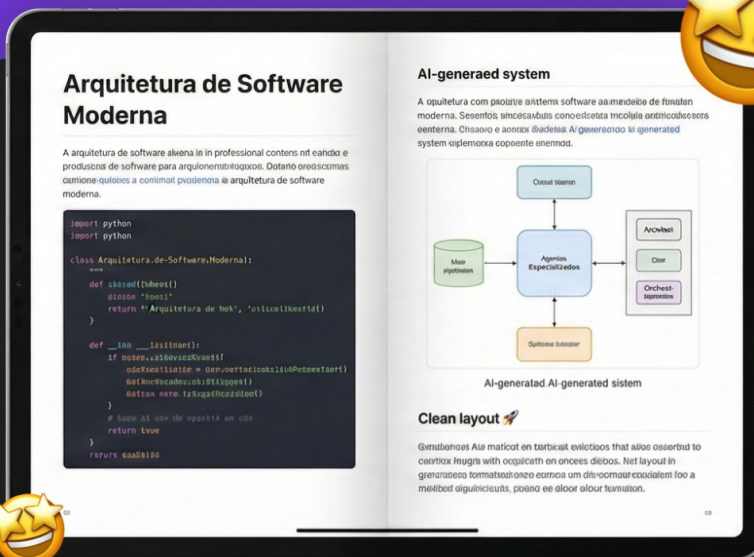


Ebookr

Crie Ebooks profissionais em minutos com IA



Chega de formatar código no Google Docs ou Word



Capas gerados por IA



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