



PYTHON
ACADEMY

COMO CONECTAR O DJANGO (PYTHON) AO MYSQL

Guia prático: conecte Django 5.1 ao MySQL. Configure DATABASES, use mysqlclient, Docker, migrations. Integração completa com MariaDB também suportada.

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) MySQL e MariaDB totalmente suportados. Django 5.1 requer MySQL 8.0+ ou MariaDB 10.5+.

Salve salve Pythonista!

O Django é um framework de desenvolvimento web em Python que permite a criação de aplicações robustas e escaláveis.

Uma das principais tarefas ao construir uma aplicação Django é conectar o framework a um banco de dados para armazenar as informações da sua aplicação.

Neste artigo, vamos explorar como conectar o Django a um Banco de Dados **MySQL**, um sistema de gerenciamento de banco de dados relacional extremamente popular.

Executando o MySQL localmente com Docker

Antes de conectar o Django ao MySQL, é necessário ter o banco de dados em execução localmente.

Uma forma fácil e conveniente de fazer isso é utilizando o Docker.

O Docker permite criar e executar aplicativos dentro de containers, de forma isolada e sem a necessidade de instalação do software diretamente no sistema operacional.

Para iniciar um container do MySQL através do Docker, você pode executar o seguinte comando no terminal:

```
docker run \
  -p 3306:3306 \
  --name mysql-container \
  -e MYSQL_ROOT_PASSWORD=sua_senha \
  -d mysql:latest
```

Esse comando irá baixar a imagem mais recente do MySQL, criar um container com o nome “mysql-container” e definir a senha do usuário root.

Além disso, ele mapeará a porta 3306 do container para a mesma porta do host local.

Configurando o Django para se conectar ao MySQL

Após ter o MySQL em execução localmente, é necessário realizar as configurações apropriadas no Django para que ele possa se conectar ao banco de dados.

Vamos considerar que você já tenha um projeto Django criado.

Para que o Django possa se comunicar com o MySQL, é necessário instalar a biblioteca `mysqlclient`.

Para isso, execute o seguinte comando no terminal:

```
pip install mysqlclient
```



Django 5.1: Alternativamente, use `pip install pymysql` (Python puro, mais fácil de instalar mas menos performante).

Em seguida, no arquivo `settings.py` do seu projeto Django, encontre a seção `DATABASES` e substitua as configurações padrão pelo seguinte código:

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'seu_banco_de_dados',
        'USER': 'seu_usuario',
        'PASSWORD': 'sua_senha',
        'HOST': 'localhost',
        'PORT': '3306'
    }
}
```

Certifique-se de substituir 'seu_banco_de_dados', 'seu_usuario' e 'sua_senha' pelos valores correspondentes ao seu ambiente local.

Caso você tenha configurado o MySQL em uma porta diferente, lembre-se de alterar o valor de `PORT` de acordo.

Após realizar as alterações nas configurações do Django, é necessário criar as tabelas do banco de dados.

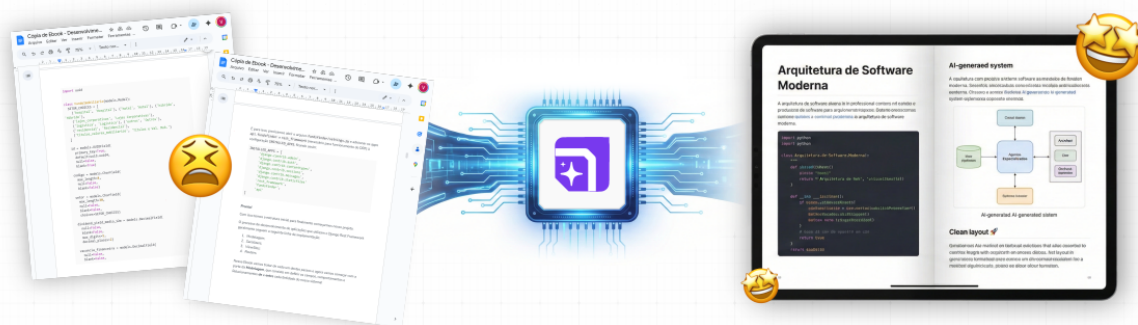
Execute o seguinte comando no terminal para aplicar essas alterações:

```
python manage.py migrate
```



Estou desenvolvendo o [Ebookr.ai](https://ebookr.ai), uma plataforma que transforma suas ideias em ebooks profissionais usando IA — com geração de capa, infográficos e exportação em PDF. Te convido a conhecer!


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, exploramos como conectar o Django a um banco de dados MySQL.

Primeiramente, vimos como executar o MySQL localmente utilizando o Docker.

Em seguida, realizamos as configurações necessárias no Django para estabelecer a conexão com o banco de dados.

Por fim, criamos um exemplo de aplicação Django que utiliza modelos para se comunicar com o MySQL.

Ao seguir essas instruções, você será capaz de utilizar o Django em conjunto com o MySQL para desenvolver poderosas aplicações web de forma simples e eficiente.

Agora você está pronto para começar a construir seus próprios projetos Django utilizando o MySQL como seu banco de dados de escolha.

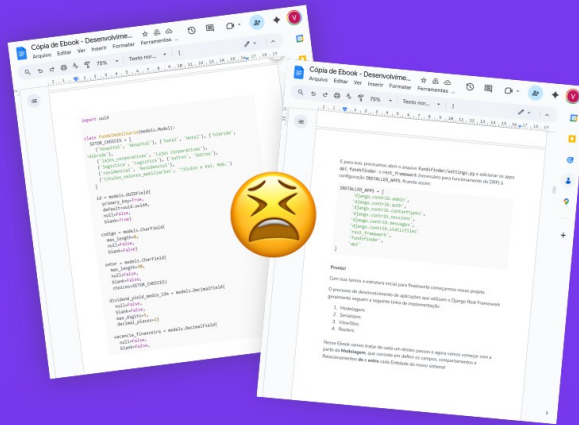
Nos vemos no próximo artigo!

Não se esqueça de conferir!

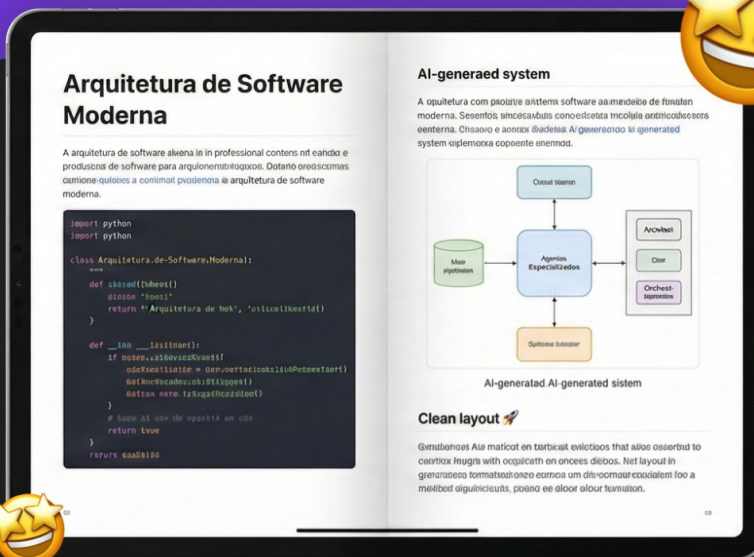


Ebookr

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