

Aula 02: Rotas em Django, Models e Variáveis de Ambiente para Banco de Dados

Introdução

Nesta aula, vamos aprofundar nossos conhecimentos em Django, abordando a criação de rotas mais complexas, a definição de models para o banco de dados e a configuração de variáveis de ambiente para gerenciar as credenciais do banco de dados.

1. Rotas em Django

1.1. Criando Rotas Adicionais

Vamos adicionar mais rotas ao nosso aplicativo. No arquivo `urls.py` do `meu_app`, adicione as seguintes rotas:

```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.index, name='index'),
    path('about/', views.about, name='about'),
    path('contact/', views.contact, name='contact'),
]
```

1.2. Criando as Views Correspondentes

No arquivo `views.py` do `meu_app`, adicione as views correspondentes:

```
from django.shortcuts import render

def index(request):
    return render(request, 'index.html')

def about(request):
    return render(request, 'about.html')

def contact(request):
    return render(request, 'contact.html')
```

2. Models em Django

2.1. Definindo um Model

Vamos criar um model para armazenar informações de um blog. No arquivo `models.py` do `meu_app`, adicione o seguinte código:

```
from django.db import models

class Post(models.Model):
    title = models.CharField(max_length=100)
    content = models.TextField()
    created_at = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.title
```

2.2. Migrando o Banco de Dados

Após definir o model, precisamos criar as migrações e aplicá-las ao banco de dados:

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

3. Variáveis de Ambiente para Banco de Dados

3.1. Instalando o `python-decouple`

Para gerenciar variáveis de ambiente, vamos utilizar o pacote `python-decouple`. Instale-o com o seguinte comando:

```
pip install python-decouple
```

3.2. Configurando o `.env`

Crie um arquivo `.env` na raiz do seu projeto e adicione as credenciais do banco de dados:

```
DB_NAME=meu_banco
DB_USER=meu_usuario
DB_PASSWORD=minha_senha
DB_HOST=localhost
DB_PORT=5432
```

3.3. Atualizando o `settings.py`

No arquivo `settings.py`, importe e utilize as variáveis de ambiente:

```
from decouple import config

DATABASES = {
```

```
    'default': {  
        'ENGINE': 'django.db.backends.postgresql',  
        'NAME': config('DB_NAME'),  
        'USER': config('DB_USER'),  
        'PASSWORD': config('DB_PASSWORD'),  
        'HOST': config('DB_HOST'),  
        'PORT': config('DB_PORT'),  
    }  
}
```

Conclusão

Nesta aula, aprendemos a criar rotas adicionais, definir models para o banco de dados e configurar variáveis de ambiente para gerenciar as credenciais do banco de dados. Com esses conhecimentos, você está mais preparado para desenvolver aplicações Django mais complexas e seguras.