

PROGRAMAÇÃO BACK-END

The Django logo, featuring the word "django" in a white, lowercase, sans-serif font with a small dot over the 'i', set against a dark green rectangular background.

The web framework for
perfectionists with deadlines.

Django Custom Users





PROGRAMAÇÃO BACK-END





Alteração da tabela de usuários

Quando inicializa-se uma nova base, ao rodar os comandos `python manage.py makemigrations` e `python manage.py migrate`, o Django cria no banco de dados algumas tabelas a mais que servem para gerenciar os usuários, grupos, permissões, etc.

PROGRAMAÇÃO BACK-END

A figura a seguir mostra os **nomes** destas **tabelas**:

- >  auth_group
- >  auth_group_permissions
- >  auth_permission
- >  auth_user
- >  auth_user_groups
- >  auth_user_user_permissions

- >  django_admin_log
- >  django_content_type
- >  django_migrations
- >  django_session

PROGRAMAÇÃO BACK-END

A tabela `auth_user`

A tabela `auth_user` possui campos específicos que são usados:

```
CREATE TABLE IF NOT EXISTS public.auth_user
(
    id integer NOT NULL GENERATED BY DEFAULT AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),
    password character varying(128) COLLATE pg_catalog."default" NOT NULL,
    last_login timestamp with time zone,
    is_superuser boolean NOT NULL,
    username character varying(150) COLLATE pg_catalog."default" NOT NULL,
    first_name character varying(150) COLLATE pg_catalog."default" NOT NULL,
    last_name character varying(150) COLLATE pg_catalog."default" NOT NULL,
    email character varying(254) COLLATE pg_catalog."default" NOT NULL,
    is_staff boolean NOT NULL,
    is_active boolean NOT NULL,
    date_joined timestamp with time zone NOT NULL,
    CONSTRAINT auth_user_pkey PRIMARY KEY (id),
    CONSTRAINT auth_user_username_key UNIQUE (username)
)
```

Muitas vezes precisa-se utilizar outros campos além das que estão nesta tabela.

PROGRAMAÇÃO BACK-END

Vamos supor que se quer acrescentar os seguintes campos na tabela **auth_user**:

- ✓ endereço varchar(255)
- ✓ cidade varchar(100)
- ✓ estado varchar(100)
- ✓ cep varchar(20)

PROGRAMAÇÃO BACK-END

Tem-se duas maneiras de se preparar:

1º) Caso já foi executado os comandos:
`python manage.py makemigrations` e
`python manage.py migrate`

2º) Caso já foi criado as tabelas do banco mas não foram executados os comandos anteriores descritos.

PROGRAMAÇÃO BACK-END

Para o 1º Caso, deve ser efetuado os seguintes procedimentos (recomenda-se fazer antes uma cópia de segurança da pasta **my_ads**, para recuperação caso ocorra algum problema):

1) Apague os arquivos de migração: Vá até a pasta **migrations** da app em seu projeto (exceto migrations padrão do Django, como auth, contenttypes, etc.) e apague todos os arquivos de migração (exceto **__init__.py**).

PROGRAMAÇÃO BACK-END

2) Apague as entradas de migrações no banco de dados: No banco de dados, as migrações aplicadas estão registradas na tabela **django_migrations**.

Utilize o Pgadmin4 e execute o comando SQL:

```
DELETE FROM django_migrations WHERE app =  
'dvdrentalapp';
```

ou

3) Deletar as tabelas do Django utilizando o Pgadmin4

PROGRAMAÇÃO BACK-END

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (34)**
 - actor
 - address
 - auth_group
 - auth_group_permissions
 - auth_permission
 - auth_user
 - auth_user_groups
 - auth_user_user_permissions
 - basket_a
 - basket_b
 - category

Dashboard Properties SQL Statistics Dependencies

Acione esta opção e confirme

Name	postgres
<input type="checkbox"/> actor	postgres
<input type="checkbox"/> address	postgres
<input checked="" type="checkbox"/> auth_group	postgres
<input checked="" type="checkbox"/> auth_group_permissions	postgres
<input checked="" type="checkbox"/> auth_permission	postgres
<input checked="" type="checkbox"/> auth_user	postgres
<input checked="" type="checkbox"/> auth_user_groups	postgres
<input checked="" type="checkbox"/> auth_user_user_permissions	postgres
<input type="checkbox"/> basket_a	postgres
<input type="checkbox"/> basket_b	postgres
<input type="checkbox"/> category	postgres
<input type="checkbox"/> city	postgres
<input type="checkbox"/> country	postgres
<input type="checkbox"/> customer	postgres
<input type="checkbox"/> departments	postgres
<input type="checkbox"/> distinct_demo	postgres
<input checked="" type="checkbox"/> django_admin_log	postgres
<input checked="" type="checkbox"/> django_content_type	postgres
<input checked="" type="checkbox"/> django_migrations	postgres
<input checked="" type="checkbox"/> django_session	postgres
<input type="checkbox"/> employee	postgres

2) Selecione a aba Properties

3) Marque as tabelas que serão deletadas

1) Selecione Tables()

PROGRAMAÇÃO BACK-END

- 4) Realizar a preparação dos conteúdos arquivos necessários (explicados mais adiante)**
- 5) executar as funções `makemigrations` e `migrate` para realizar as alterações.**

PROGRAMAÇÃO BACK-END

Para o 2º Caso, não é necessário nenhuma alteração, somente a preparação dos arquivos normalmente e depois executar as funções **makemigrations** e **migrate**.

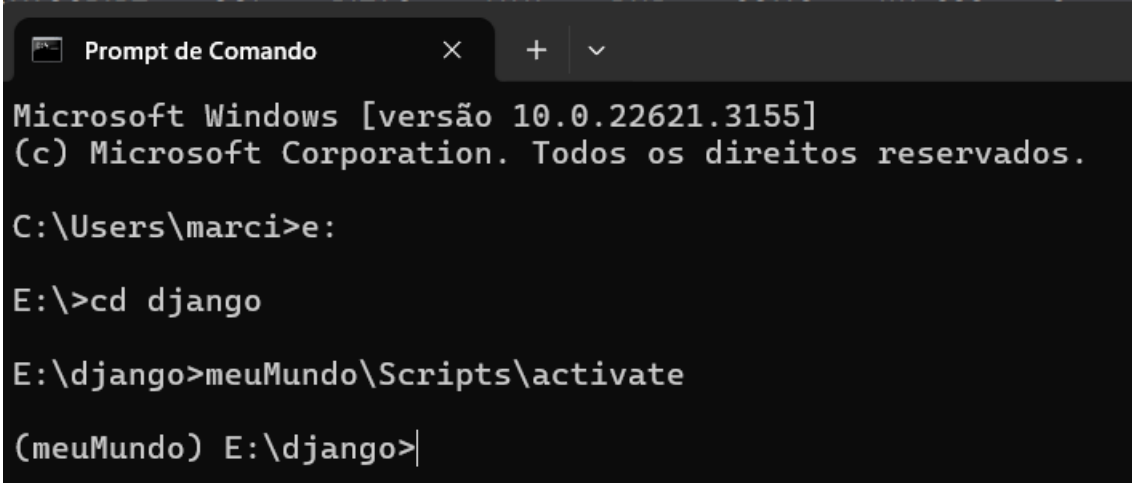
Obs importante: Antes de executar os comandos **makemigrations** e **migrate**, deve ser dado o **activate** e iniciar o banco.

PROGRAMAÇÃO BACK-END

Iniciando

PROGRAMAÇÃO BACK-END

Rodar o script activate.bat



```
Prompt de Comando
Microsoft Windows [versão 10.0.22621.3155]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\marci>e:

E:\>cd django

E:\django>meuMundo\Scripts\activate

(meuMundo) E:\django>
```


Este passo é importante para configurar as variáveis de ambiente.

PROGRAMAÇÃO BACK-END

Iniciar Servidor Postgresql

PROGRAMAÇÃO BACK-END

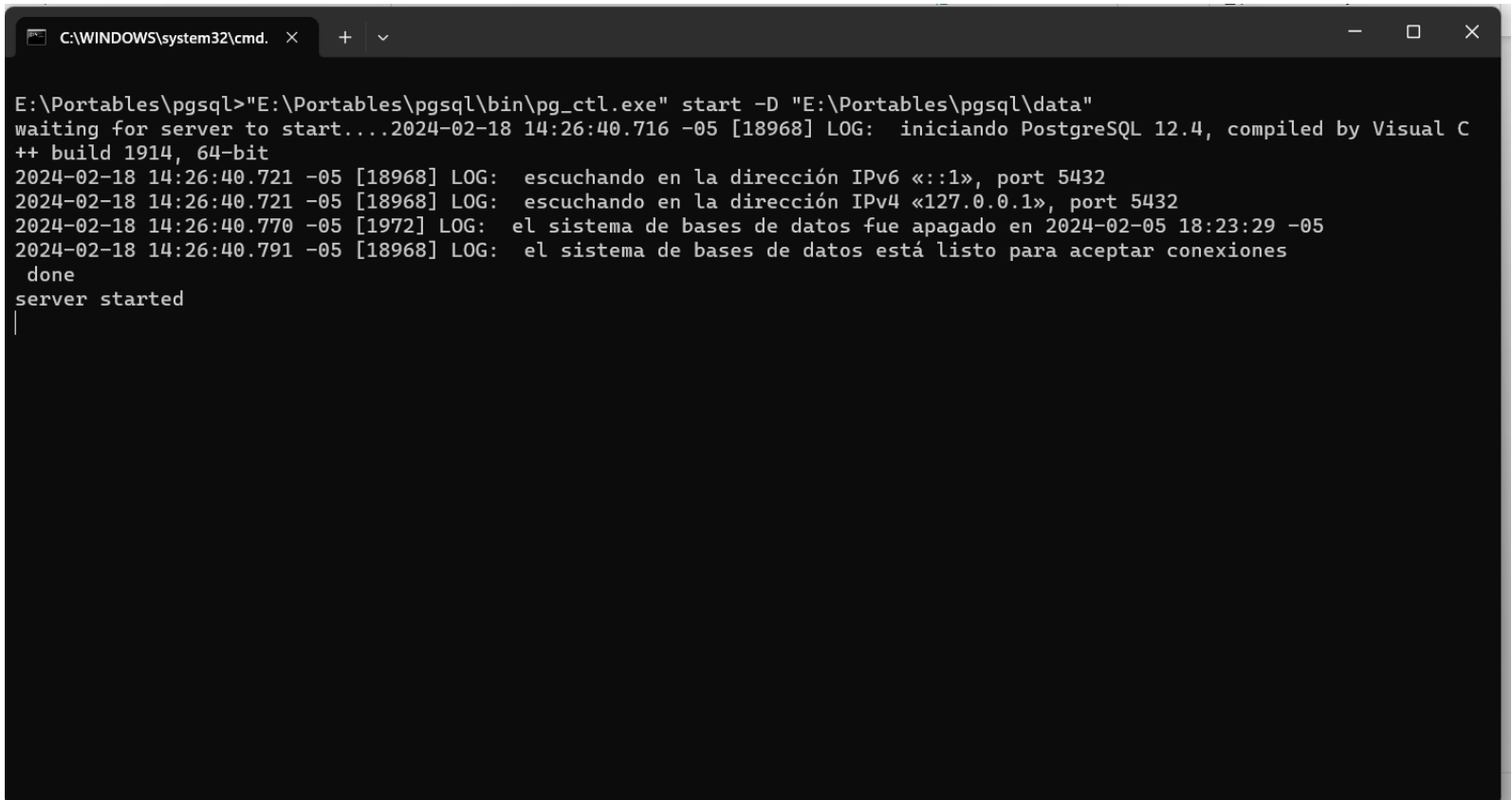
Na pasta `../pgsql` execute o arquivo **PostgreSQL-Start.bat** para iniciar o servidor



Nome	Data de modificação	Tipo	Tamanho
bin	11/08/2020 08:41	Pasta de arquivos	
data	18/02/2024 16:26	Pasta de arquivos	
doc	11/08/2020 08:41	Pasta de arquivos	
include	11/08/2020 08:41	Pasta de arquivos	
lib	11/08/2020 08:41	Pasta de arquivos	
pgAdmin 4	11/08/2020 08:41	Pasta de arquivos	
share	11/08/2020 08:41	Pasta de arquivos	
StackBuilder	11/08/2020 08:41	Pasta de arquivos	
Make Cluster.bat	09/01/2017 14:12	Arquivo em Lotes ...	1 KB
PgAdmin4.bat	28/04/2017 22:29	Arquivo em Lotes ...	1 KB
PostgreSQL-Restart.bat	06/04/2012 09:42	Arquivo em Lotes ...	1 KB
PostgreSQL-Start.bat	28/04/2017 21:46	Arquivo em Lotes ...	1 KB
PostgreSQL-Stop.bat	06/04/2012 09:43	Arquivo em Lotes ...	1 KB
Service - Delete.bat	29/08/2020 23:25	Arquivo em Lotes ...	1 KB
Service - Maker.bat	29/08/2020 23:26	Arquivo em Lotes ...	1 KB

PROGRAMAÇÃO BACK-END

Será aberta uma janela em Texto informando que o servidor iniciou (**server started**).
Não feche esta janela

A screenshot of a Windows Command Prompt window. The title bar shows the path 'C:\WINDOWS\system32\cmd.' and standard window controls. The command prompt shows the execution of 'pg_ctl.exe' to start the PostgreSQL 12.4 server. The output includes log messages indicating the server is starting, listening on IPv6 and IPv4 ports, and that the database system is ready to accept connections. The final output is 'server started' followed by a cursor.

```
C:\WINDOWS\system32\cmd. x + v

E:\Portables\pgsql>"E:\Portables\pgsql\bin\pg_ctl.exe" start -D "E:\Portables\pgsql\data"
waiting for server to start...2024-02-18 14:26:40.716 -05 [18968] LOG:  iniciando PostgreSQL 12.4, compiled by Visual C
++ build 1914, 64-bit
2024-02-18 14:26:40.721 -05 [18968] LOG:  escuchando en la direcci3n IPv6 «::1», port 5432
2024-02-18 14:26:40.721 -05 [18968] LOG:  escuchando en la direcci3n IPv4 «127.0.0.1», port 5432
2024-02-18 14:26:40.770 -05 [1972] LOG:  el sistema de bases de datos fue apagado en 2024-02-05 18:23:29 -05
2024-02-18 14:26:40.791 -05 [18968] LOG:  el sistema de bases de datos est1 listo para aceptar conexiones
done
server started
|
```

PROGRAMAÇÃO BACK-END

Arquivos settings.py e models.py

PROGRAMAÇÃO BACK-END

Na pasta **dvdrentalapp** adicione o seguinte conteúdo no arquivo **models.py**

```
dvdrentalapp > 📁 models.py > 📄 Post
1  from django.db import models
2  from django.contrib.auth.models import AbstractUser
3
4  class CustomUser(AbstractUser):
5      endereco = models.CharField(max_length=255)
6      cidade = models.CharField(max_length=100)
7      estado = models.CharField(max_length=100)
8      cep = models.CharField(max_length=20)
9
```

PROGRAMAÇÃO BACK-END

Na pasta **my_ads** adicione o seguinte conteúdo no arquivo **settings.py**

```
my_ads > settings.py > ...  
101  
102     AUTH_PASSWORD_VALIDATORS = [  
103  
104     ]  
105  
106     AUTH_USER_MODEL = 'dvdrentalapp.CustomUser'  
107
```

No banco será criado a tabela **CustomUser**

PROGRAMAÇÃO BACK-END

python manage.py makemigrations

```
Prompt de Comando
(meuMundo) E:\django\my_ads>py manage.py makemigrations
Migrations for 'dvdrentalapp':
  dvdrentalapp\migrations\0001_initial.py
    - Create model Actor
    - Create model Address
    - Create model BasketA
    - Create model BasketB
    - Create model Category
    - Create model City
    - Create model Country
    - Create model Customer
    - Create model Departments
    - Create model DistinctDemo
    - Create model Employee
    - Create model Employees
    - Create model Film
    - Create model Inventory
    - Create model Language
    - Create model Payment
    - Create model Post
    - Create model Rental
    - Create model Staff
    - Create model Store
    - Create model T1
    - Create model T2
    - Create model FilmActor
    - Create model FilmCategory
    - Create model CustomUser
(meuMundo) E:\django\my_ads>
```

PROGRAMAÇÃO BACK-END

python manage.py migrate

```
(meuMundo) E:\django\my_ads>py manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, dvdrentalapp, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0001_initial... 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 auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying dvdrentalapp.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying sessions.0001_initial... OK

(meuMundo) E:\django\my_ads>
```

PROGRAMAÇÃO BACK-END

Tabelas Criadas

Tables (34)

- > actor
- > address
- > auth_group
- > auth_group_permissions
- > auth_permission
- > basket_a
- > basket_b
- > category
- > city
- > country
- > customer
- > departments
- > distinct_demo
- > django_admin_log
- > django_content_type
- > django_migrations
- > django_session
- > dvdrentalapp_customuser
- > dvdrentalapp_customuser_groups
- > dvdrentalapp_customuser_user_permissions
- > employee
- > employees
- > film

PROGRAMAÇÃO BACK-END

Campos inseridos

```
5  CREATE TABLE IF NOT EXISTS public.dvdrentalapp_customuser
6  (
7      id bigint NOT NULL GENERATED BY DEFAULT AS IDENTITY ( INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 922
8      password character varying(128) COLLATE pg_catalog."default" NOT NULL,
9      last_login timestamp with time zone,
10     is_superuser boolean NOT NULL,
11     username character varying(150) COLLATE pg_catalog."default" NOT NULL,
12     first_name character varying(150) COLLATE pg_catalog."default" NOT NULL,
13     last_name character varying(150) COLLATE pg_catalog."default" NOT NULL,
14     email character varying(254) COLLATE pg_catalog."default" NOT NULL,
15     is_staff boolean NOT NULL,
16     is_active boolean NOT NULL,
17     date_joined timestamp with time zone NOT NULL,
18     endereco character varying(255) COLLATE pg_catalog."default" NOT NULL,
19     cidade character varying(100) COLLATE pg_catalog."default" NOT NULL,
20     estado character varying(100) COLLATE pg_catalog."default" NOT NULL,
21     cep character varying(20) COLLATE pg_catalog."default" NOT NULL,
22     CONSTRAINT dvdrentalapp_customuser_pkey PRIMARY KEY (id),
23     CONSTRAINT dvdrentalapp_customuser_username_key UNIQUE (username)
24 )
```