

A simple illustration of a hanging lamp with a purple shade and a green light beam, positioned above the Django logo.

django

Isadora Ferrão

Doutoranda em Ciência da Computação
Universidade de São Paulo (USP)





django

**Criação do
ambiente virtual**



1 Criação do ambiente virtual

```
python3 -m venv nome_ambiente
```

2 Ativação do ambiente

```
nome_ambiente/Scripts/Activate
```

A purple hanging lamp with a green light cone illuminating the word "django".

django

Criação do projeto



1

Instalação django

```
pip install django
```

(dentro da pasta do ambiente virtual)

2

Criação do projeto

```
django-admin startproject nome_projeto
```

Pasta projeto - servidor 127.0.0.1:8000

```
python manage.py runserver
```

An illustration on the left side of the slide shows a hand holding a smartphone with a red screen displaying a white grid pattern. The phone is resting on a blue surface with white circles and starburst patterns. Above the phone, a purple pendant light hangs, casting a greenish glow.

django

Criação do aplicativo

1

Criação do aplicativo

```
python manage.py startapp APP_NAME  
(dentro da pasta do projeto)
```

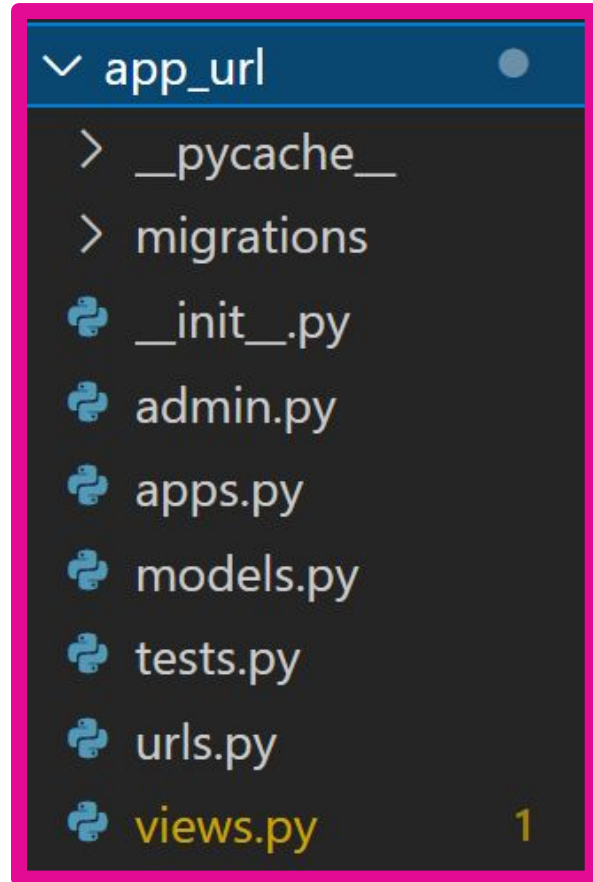



django

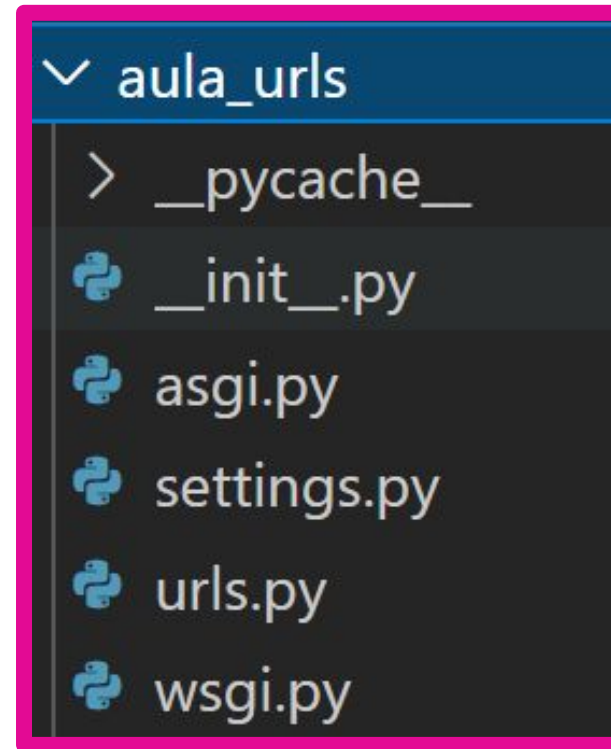
**Como diferenciar
app e projeto?**



Como diferenciar app e projeto?

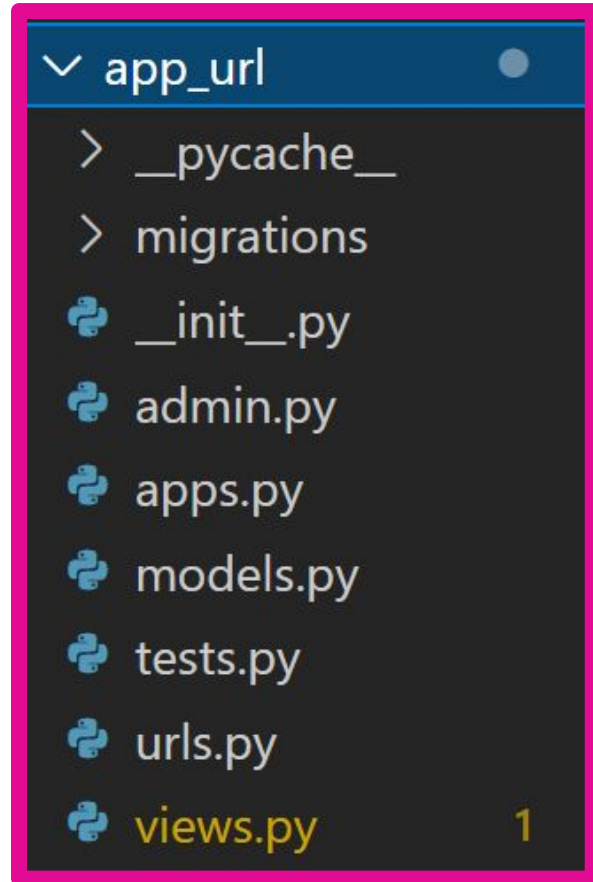


Aplicativo

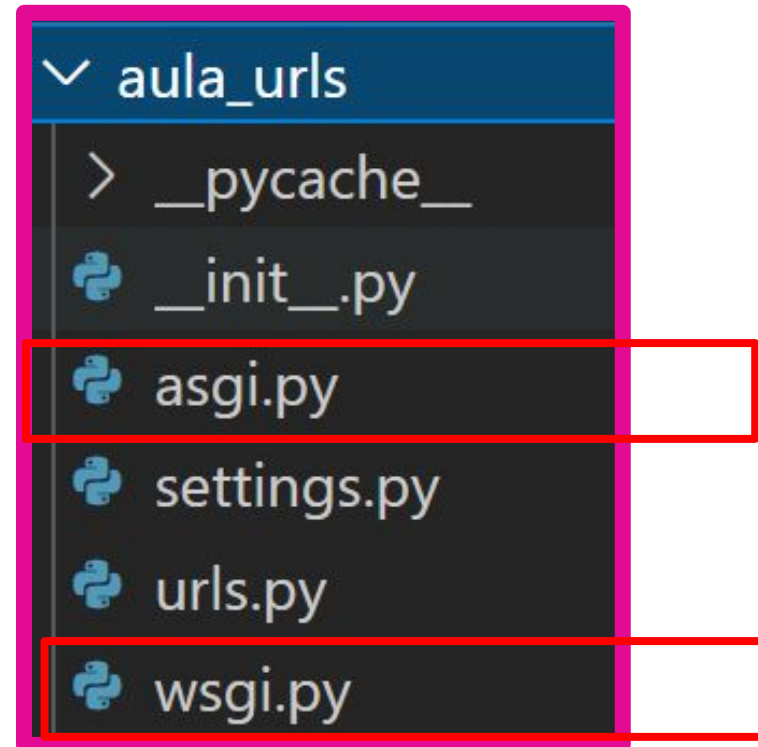


Projeto

Como diferenciar app e projeto?



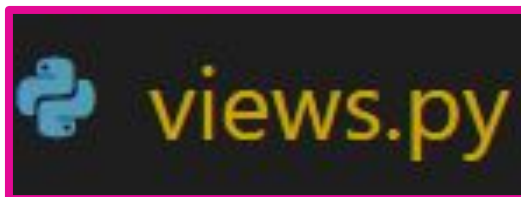
Aplicativo



Projeto



Classes, urls e bibliotecas



```
def nome_classe(request):
```

**As Views são compostas por
classes que agrupam funcionalidades**

**Essas classes descrevem o comportamento padrão para as
funcionalidades (listagem, exclusão, busca simples, atualização).**



```
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path("exemplo/", include("app_url.urls")),  
]
```

Cada página na Internet precisa de sua própria URL.

Desta forma seu aplicativo sabe o que deve mostrar a um usuário que abre uma URL. Em Django, nós usamos o `urls.py` para que Django apresente a URL pro usuário



 `urls.py`

```
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path("exemplo/", include("app_url.urls")),  
]
```

Nome url

**Nome no aplicativo + . + nome do
arquivo que você quer chamar**



Importações

Importação do arquivo views que está presente no aplicativo app_url

```
from app_url import views
```

Nome do app ou projeto
que pertence o arquivo

Nome do arquivo que você
deseja importar

Você precisa pegar o nome do aplicativo ou do projeto
que você quer importar + import + nome do arquivo

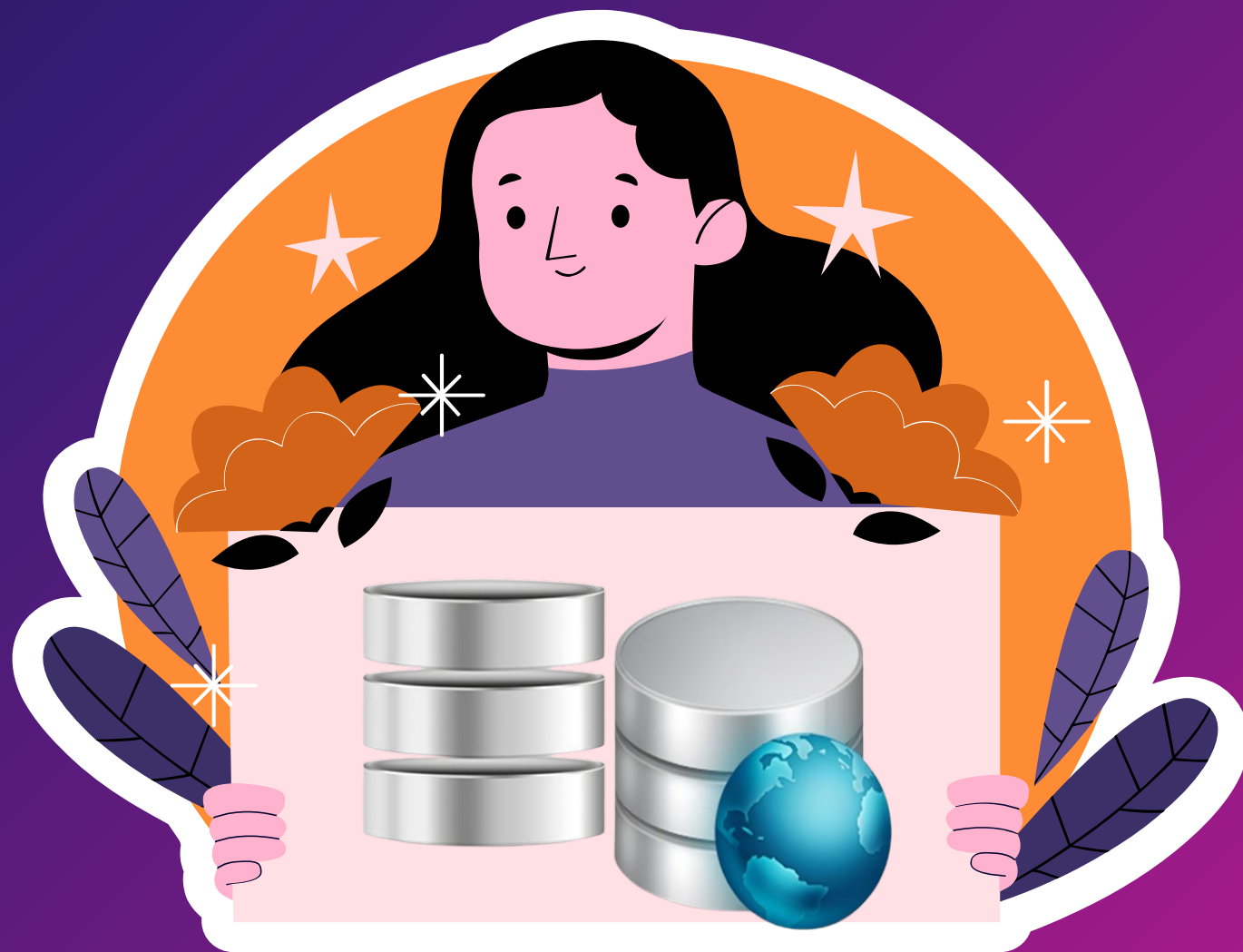


Como executar o projeto

Pasta projeto - servidor 127.0.0.1:8000

```
python manage.py runserver
```

Model e migrations



Model e migrations

Model ou model.py

Vai conter informações sobre alguma coisa, ex: uma tarefa onde temos os atributos título, descrição, data de criação, etc. Então todas as simulações vão estar no arquivo model e depois disso o django vai conseguir compreender que cada linha desse arquivo vai ser um campo da nossa tabela lá no campo do banco de dado

Model e migrations



Após criada uma model com todas as instruções de dados, quando rodamos o comando

python manage.py makemigration

transferimos esse código python para uma instrução sql,

Depois disso, com o comando:

python manage.py migrate

a gente consegue migrar esse arquivo sql para as tabelas do banco.