


Open · Opened 4 days ago by  **Rubén Montero**

# Django

## Resumen

- Veremos qué es Django
- Lanzaremos un comando para inicializar un proyecto nuevo
- Visitaremos la página de ejemplo desde el navegador

## Descripción

Vamos a comenzar a trabajar con Python como se suele hacer habitualmente: *Aprovechando la inmensa variedad de utilidades y librerías que están a nuestra disposición.*

Uno de los puntos fuertes de Python es que, para *algo* que queramos hacer, seguramente *alguien* ya lo haya hecho antes.

### No hay que reinventar la rueda

Nosotros tenemos por **objetivo final** implementar un API REST con Python. ¡Ya hay mucha gente y proyectos destinados a servicios web con Python!

Vamos a comenzar a trabajar con Django.

## ¿Django?

Este [Django](#) está igual de *desencadenado* que el de la [película](#).

Se trata de un *framework* de desarrollo web con Python.

### ¡Es relevante!

Las siguientes páginas web [están creadas con Django](#):

- Youtube
- Instagram
- Pinterest
- Dropbox
- Spotify
- National Geographic
- Bitbucket
- Mozilla Firefox (sitio web)
- NASA (sitio web)

### Asombroso

En efecto.

### Quiero aprender Django

Muy bien.

## Primer paso

**Abramos** nuestro *terminal* de Windows. Podemos pulsar **Tecla de windows > cmd** y darle a ENTER.

**pip**

Las librerías en Python se instalan, generalmente, usando [pip](#). Es un comando que ya obtuvimos al instalar Python.

Para instalar Django, ejecutamos:

```
pip install django
```

(Y ya que estamos, instalemos una dependencia que facilitará la ejecución de los tests)

```
pip install requests
```

¡Instalado!

Estas *librerías Python* se ubicarán entre las librerías *que guarda nuestro sistema operativo*

## Cambiar al repositorio local

Usamos `cd` para cambiar el directorio activo a la carpeta de nuestro proyecto `python-introduction/`, cambiamos a la carpeta `api/` en su interior.

## Iniciar un proyecto Django

Desde `python-introduction/api/` ejecutamos este comando:

```
django-admin startproject RestAPI
```

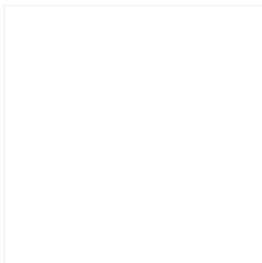
Si tienes problemas porque el comando `django-admin` parece no estar disponible, quizá hubo algún error al registrarlo en el PATH durante la instalación de Django.

Alternativamente, puedes probar este comando que es **equivalente**:

```
python -m django startproject RestAPI
```

## ¿Qué ha pasado?

Si **no** ha salido ningún mensaje de error, habrás creado una nueva carpeta en `api/` con el siguiente contenido:



Ahora, vamos a [lanzar el proyecto](#) que contiene una página de ejemplo.

**Cambia** al directorio recién creado:

```
cd RestAPI
```

...y **ejecuta** el fichero `manage.py`, pasando el *argumento* `runserver`:

```
python manage.py runserver
```

Esta será la salida esperable:

```
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, autht
Run 'python manage.py migrate' to apply them.
July 08, 2022 - 09:42:43
Django version 4.0.5, using settings 'RestAPI.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

## ¡El servidor está funcionando!

Si te fijas, una de las líneas reza:

```
Starting development server at http://127.0.0.1:8000/
```

`127.0.0.1` es sinónimo de `localhost` en IPv4.

Es decir, podemos abrir nuestro navegador web favorito y escribir `http://localhost:8000/` en la barra de direcciones...

**Voilà!**

¿Qué ves?

¿Y para detenerlo?

Basta con pulsar `CTRL+C` o bien cerrar el terminal (cmd.exe)

¿Se puede ejecutar Django desde PyCharm?

¡Sí!

Es tan fácil como añadir una nueva **configuración**.

Para ello, seleccionamos el *desplegable* en la barra superior, al lado del botón verde de **'Play'** y elegimos **Edit Configurations**

En la ventana que sale, *arriba a la izquierda* hay un botón *más* ( `+` ). Le damos **Add new configuration > Python**.

Ahora, en el panel principal de la ventana editamos 3 cosas:

1. **Name:** (arriba). En vez de *Unnamed*, podemos dar algo más significativo como *Django* ó *RestAPI*
2. **Script path:** Hacemos click en el icono de *carpeta* (a la derecha) y buscamos *python-introduction > api > RestAPI > manage.py* .
3. **Parameters:** (justo debajo). Escribimos *runserver*

Dándole a **OK** se guarda la configuración, y el **botón verde 'Play'** la ejecuta. ¡Ya podemos probar el servidor desde PyCharm!

**Por último**

Verifica que tu código pasa el *test* asociado a la tarea.

Haz `commit` y `push` para subir los cambios al repositorio.



[Rubén Montero @ruben.montero](#) changed milestone to [%Sprint 3](#) 4 days ago