

# Tutorial de Django Girls

---

para la versión 0.3.1, 20 Jan 2021

Django Girls

Revisado por David Arroyo Menéndez ([davidam@gmail.com](mailto:davidam@gmail.com))

---

Tutorial de Django Girls (version 0.3.1, 20 Jan 2021).

Copyright © 2021 David Arroyo Menéndez

Este documento está bajo una Licencia Creative Commons Atribución Compartir por Igual 3.0 España (<http://creativecommons.org/licenses/by-sa/3.0/es/deed.es>)

# Índice General

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Bienvenida	1
1.2	Introducción	1
1.3	¿Qué aprenderás con este tutorial?	1
1.4	Seguir el tutorial desde casa	2
1.5	Sobre nosotras y Cómo contribuir	3
1.6	¿Te gustaría ayudarnos a traducir el tutorial a otros idiomas?	3
<b>2</b>	<b>Instalación</b>	<b>4</b>
2.1	Si estás haciendo el tutorial en casa	4
2.2	Si estás asistiendo a un workshop	4
2.3	Instalación	4
2.4	Breve introducción a la línea de comandos	4
2.4.1	Abriendo: Windows	5
2.4.2	Abriendo: OS X	5
2.4.3	Abriendo: Linux	6
2.5	Instalar Python	6
2.5.1	Instalar Python: Windows	6
2.5.2	Instalar Python: Linux	7
2.5.3	Instalar Python: Debian o Ubuntu	7
2.5.4	Instalar Python: Fedora	7
2.5.5	Instalar Python: openSUSE	7
2.5.6	Instala un Editor de Código	8
2.5.7	Visual Studio Code	8
2.5.8	Gedit	8
2.5.9	Sublime Text	8
2.5.10	Atom	8
2.6	¿Por qué estamos instalando un editor de código?	8
2.7	Configura el entorno virtual (virtualenv) e instala Django	9
2.8	Entorno virtual	9
2.8.1	Virtual environment: Windows	9
2.8.2	Virtual environment: Linux and OS X	9
2.9	Trabajar con virtualenv	11
2.9.1	Working with virtualenv: Windows	11
2.9.2	Working with virtualenv: Linux and OS X	11
2.10	Instalar Django	12
2.10.1	Instalar paquetes con un fichero de requisitos (requirements)	12
2.10.2	Installing Django: Windows	12
2.10.3	Installing Django: Windows 8 and Windows 10	13
2.10.4	Installing Django: Linux	13
2.11	Instalar Git	13
2.11.1	Installing Git: Windows	13
2.11.2	Installing Git: OS X	13

2.11.3	Installing Git: Debian or Ubuntu .....	13
2.11.4	Installing Git: Fedora .....	13
2.11.5	Installing Git: openSUSE .....	13
2.12	Crear una cuenta de GitHub .....	14
2.13	Crear una cuenta de PythonAnywhere .....	14
2.13.1	Crear un token para la API de PythonAnywhere .....	14
2.13.2	Comienza a leer .....	15
2.13.3	¡Disfruta el taller! .....	15
<b>Index .....</b>		<b>16</b>

# 1 Introducción

## 1.1 Bienvenida

¡Bienvenido/a al tutorial de las Django Girls! ¡Nos alegra que estés aquí :) En este tutorial, te llevaremos de viaje a las entrañas de la tecnología web, para que veas todas las piezas que se necesitan para que la web funcione.

Como pasa con todas las cosas nuevas, va a ser una aventura - pero no te preocupes; una vez que te has decidido a empezar, te irá de maravilla :)

## 1.2 Introducción

¿Alguna vez has tenido la sensación de que el mundo es cada vez más tecnológico? ¿que cada vez lo entiendes menos? ¿Alguna vez te has planteado crear un sitio web pero no sabías por dónde empezar? ¿Has pensado alguna vez que el mundo del software es demasiado complicado como para intentar hacer algo por tu cuenta?

Bueno, ¡tenemos buenas noticias! Programar no es tan difícil como parece y queremos demostrarte lo divertido que puede llegar a ser.

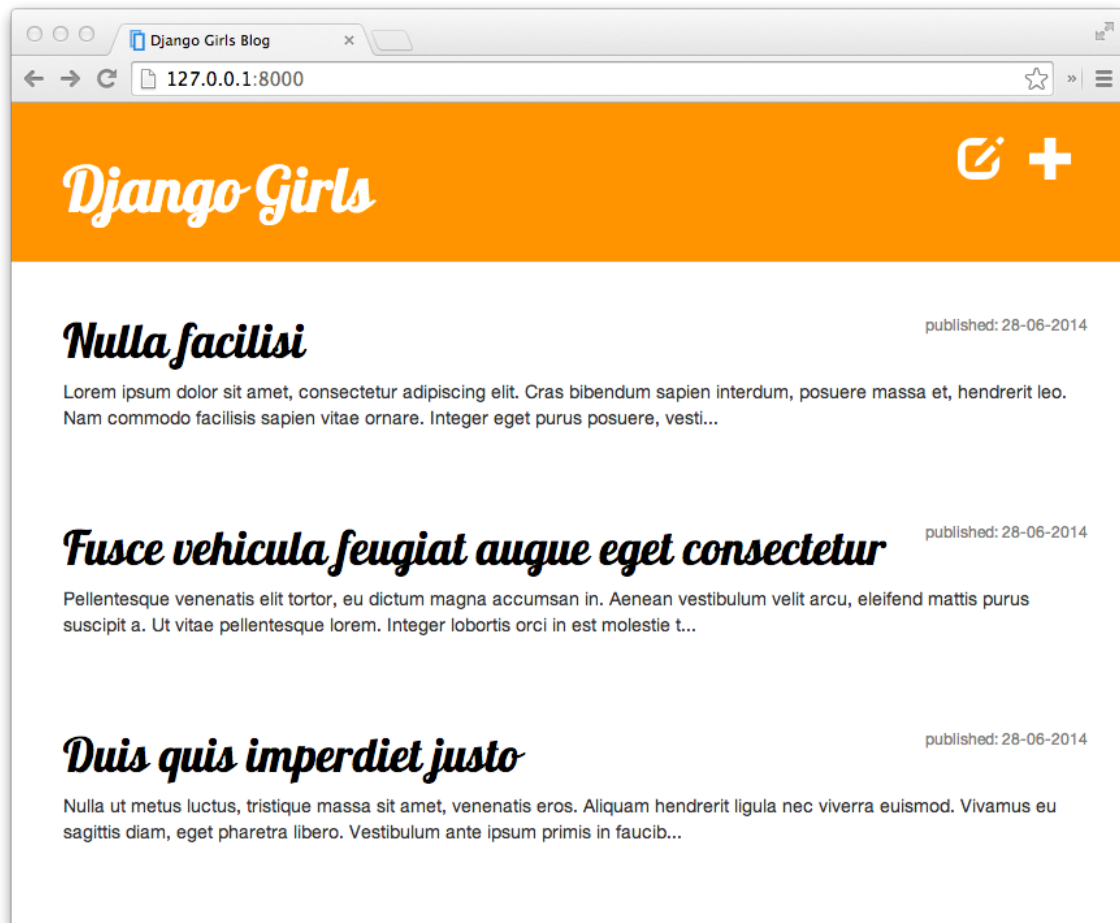
Este tutorial no te convertirá en programadora de la noche a la mañana. Si quieres ser buena en esto, necesitarás meses o incluso años de aprendizaje y práctica. Sin embargo queremos enseñarte que programar o crear sitios web no es tan complicado como parece. Intentaremos explicar las cosas lo mejor que podamos, para perderle el miedo a la tecnología.

¡Esperamos conseguir que te guste la tecnología tanto como a nosotras!

## 1.3 ¿Qué aprenderás con este tutorial?

Cuando termines el tutorial, tendrás una aplicación web sencilla y funcional: tu propio blog. Te mostraremos como ponerla en línea, ¡para que otros puedan ver tu trabajo!

Tendrá (más o menos) esta pinta:



Si estás siguiendo este tutorial por tu cuenta y no tienes un mentor que te ayude en caso de dificultades, tenemos un chat para ti: Gitter. ¡Hemos pedido a mentores y asistentes de ediciones anteriores, que pasen por allí de vez en cuando para echar una mano a otras con el tutorial! ¡No tengas miedo de preguntar ahí!

Si estás siguiendo este tutorial por tu cuenta y no tienes un mentor que te ayude en caso de dificultades, tenemos un chat para ti: Gitter (<https://gitter.im/DjangoGirls/tutorial>). ¡Hemos pedido a mentores y asistentes de ediciones anteriores, que pasen por allí de vez en cuando para echar una mano a otras con el tutorial! ¡No tengas miedo de preguntar ahí!

## 1.4 Seguir el tutorial desde casa

Participar en un taller de Django Girls en vivo es genial, pero somos conscientes de que no siempre es posible hacerlo. Por eso, te recomendamos hacer este tutorial en casa. Para las que estáis en casa, estamos preparando vídeos que facilitarán seguir el tutorial por tu cuenta. Todavía está en progreso, pero cada vez hay más cosas explicadas en el canal de YouTube

Coding is for girls (<https://www.youtube.com/channel/UC0hNd2uW8jTR5K3KBzRuG2A/feed>) (Nota: en inglés).

En cada capítulo hay un enlace que lleva al vídeo correspondiente (si lo hay).

## 1.5 Sobre nosotras y Cómo contribuir

Este tutorial lo mantiene DjangoGirls (<https://djangogirls.org/>). Si encuentras algún error o quieres actualizar el tutorial, por favor sigue la guía de cómo contribuir (<https://github.com/DjangoGirls/tutorial/blob/master/README.md>)

## 1.6 ¿Te gustaría ayudarnos a traducir el tutorial a otros idiomas?

Actualmente, las traducciones se hacen en la plataforma crowdin.com, en el siguiente enlace:

<https://crowdin.com/project/django-girls-tutorial>

Si tu idioma no aparece en la lista de crowdin, por favor abre un nuevo issue con el idioma para que podamos añadirlo.

## 2 Instalación

### 2.1 Si estás haciendo el tutorial en casa

Si estás haciendo el tutorial en casa, y no en uno de los eventos de Django Girls, puedes saltar este capítulo por completo e ir directamente al capítulo ¿cómo funciona Internet?.

Esto es porque cubrimos las instalaciones de cosas a medida que se requieran en el tutorial – esta es solamente una página adicional que reúne toda la información de instalación en un solo lugar (lo que es útil para algunos formatos de taller). Puedes escoger instalar todo lo que está en esta página ya mismo si lo deseas. Pero si quieres empezar a aprender cosas antes de instalar un grupo de materiales en tu computadora, salta este capítulo y te explicaremos las partes de la instalación luego, cuando sean necesarias.

¡Buena suerte!

### 2.2 Si estás asistiendo a un workshop

Si estás asistiendo a uno de los Django Girls events:

Tu workshop puede tener una "fiesta de instalación" antes del workshop principal. Si estás en un equipo de instalación, ¡esta página es para ti! Sigue las instrucciones aquí para obtener todo lo que tu necesitas para el workshop de instalación, con la ayuda de los entrenadores si lo necesitas. Entonces en el workshop principal, tu estarás preparado para saltar las instrucciones de instalación que encontrarás en el tutorial principal cuando llegues a ellos.

Los organizadores del taller pueden pedirte que en casa intentes instalar todo en tu computadora antes de iniciar el taller. Si has estado preguntando cómo hacer esto, ¡esta página es para ti! Sigue las instrucciones aquí, lo mejor puedas. Así, en el taller principal, cuando estés en uno de los pasos de la instalación del tutorial, y si no tenías esa pieza instalada, puedes pedir ayuda a una de tus entrenadoras.

Si tu taller no tiene una sesión de instalación (o no pudiste asistir), y si los organizadores no te piden que intentes instalar todo antes de tu llegada, salta esta página y ve al capítulo Cómo el internet funciona. Instalarás todo lo que necesitas para trabajar a lo largo del tutorial.

### 2.3 Instalación

En este tutorial vas a construir un blog. Según cómo vayas a través del tutorial, serás instruida en cómo instalar varios softwares en tu computadora y configurar algunas cuentas online como sean necesarias. Esta página reúne todas las instalaciones e instrucciones del registro en un lugar (el cual es útil para algunos formatos del taller).

### 2.4 Breve introducción a la línea de comandos

Muchos de los pasos citados abajo hacen referencia a la "consola", "terminal", "ventana de comandos", o "línea de comandos" – todos éstos términos significan la misma cosa: una ventana en tu computadora donde puedes introducir comandos. Cuando estés en el tutorial principal, aprenderás más acerca de la línea de comandos. Por ahora, la parte principal que necesitas es saber cómo abrir una ventana de comandos y cómo luce:



### 2.4.1 Abriendo: Windows

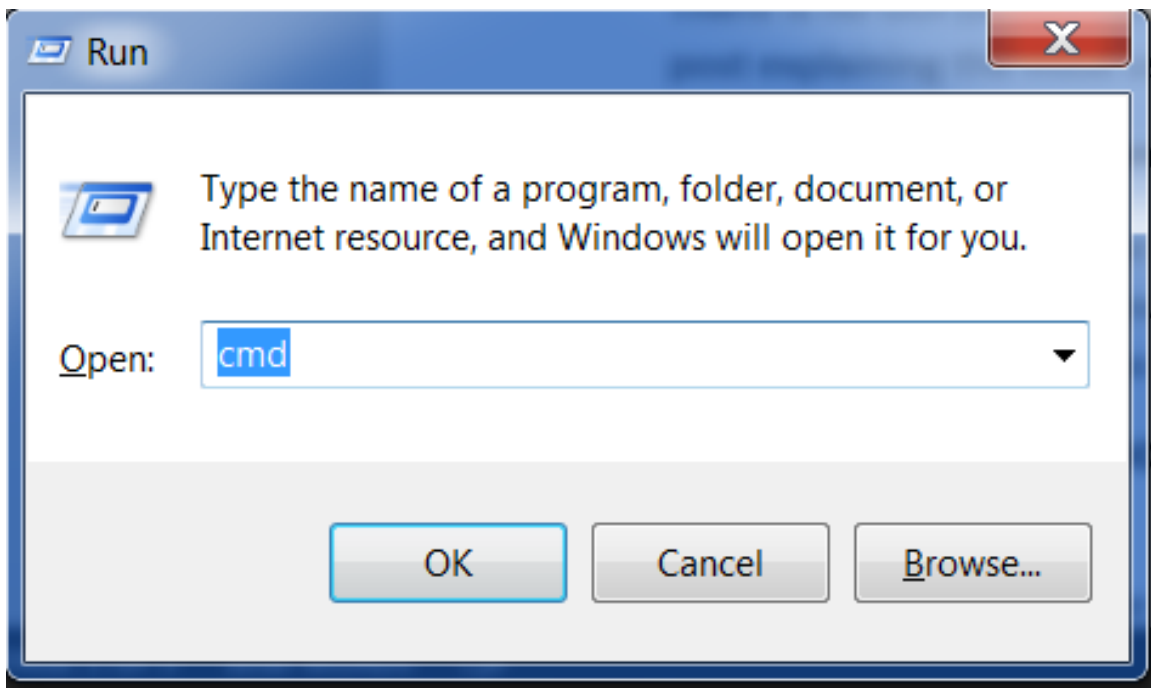
Dependiendo de tu versión de Windows y tu teclado, una de las opciones siguientes debería abrir una ventana de comandos (puede que necesites experimentar un poco, pero no se necesita probar todas estas sugerencias):

Ve al menú o pantalla de Inicio, y escribe "Símbolo del Sistema" en el cuadro de búsqueda. Ve a Menú de inicio → Windows System → Command Prompt. Ve al menú de Inicio → Todos los Programas → Accesorios → Símbolo del Sistema.

Ve a la pantalla de Inicio, pasa el ratón sobre la esquina inferior izquierda de la pantalla, y haz click en la flecha hacia abajo (en una pantalla táctil, desliza hacia arriba desde la parte baja de la pantalla). La página de la Aplicación debería abrirse. Haz click en Símbolo del Sistema en la sección Sistema de Windows.

Mantén la tecla especial de Windows de tu teclado y pulsa "X". Elige "Símbolo del Sistema" del menú emergente.

Mantén pulsada la tecla de Windows y pulsa "R" para abrir una ventana "Ejecutar". Escribe "cmd" en la caja, y haz click en OK.



Más adelante en este tutorial, necesitarás tener dos consolas de comandos abiertas a la misma vez. Sin embargo, en algunas versiones de Windows, si ya tienes abierta una ventana de comandos e intentas abrir otra usando el mismo método, simplemente maximizará la que ya tienes abierta. ¡Inténtalo ahora en tu ordenador y mira qué ocurre! Si solo se abre una ventana de comandos, intenta alguno de los otros métodos explicados anteriormente. Al menos uno de ellos debería abrir una nueva ventana de comandos.

### 2.4.2 Abriendo: OS X

Ve a Aplicaciones → Utilidades → Terminal.

### 2.4.3 Abriendo: Linux

Probablemente se encuentre en Aplicaciones  $\mapsto$  Accesorios  $\mapsto$  Terminal, o Aplicaciones  $\mapsto$  Sistema  $\mapsto$  Terminal, aunque esto dependerá de tu sistema. Si no lo encuentras allí, intenta buscarlo en Google. :)

## 2.5 Instalar Python

Para lectores en casa: este capítulo se cubre en el vídeo Installing Python & Code Editor. (<https://www.youtube.com/watch?v=pVTaqzKZCdA>)

Esta sección está basada en un tutorial de Geek Girls Carrots (<https://github.com/ggcarrots/django-carrots>)

Django está escrito en Python. Necesitamos Python para hacer cualquier cosa en Django. ¡Empecemos con instalarlo! Queremos que instales la última versión de Python 3, así que si tienes una versión anterior, necesitarás actualizarla. Si ya tienes la versión 3.4 o una superior, debería ir bien.

Por favor, instala Python normalmente de la siguiente forma, incluso si tienes Anaconda instalada en el ordenador.

### 2.5.1 Instalar Python: Windows

Primero comprueba si tu ordenador ejecuta la versión 32 bits de Windows o la de 64, en "Tipo de sistema" en la página de "Acerca de". Para llegar a esta página, intenta uno de estos métodos:

- Presiona la tecla de Windows y la tecla Pause/Break al mismo tiempo
- Abre el Panel de Control desde el menú de Windows, después accede a Sistema & Seguridad, luego a Sistema
- Presiona el botón de Windows, luego accede a Configuración > Sistema > Acerca de

Puedes descargar Python para Windows desde la siguiente web <https://www.python.org/downloads/windows/>. Clica en el enlace "Latest Python 3 Release -Python x.x.x". Si tu ordenador ejecuta la versión de 64 bits de Windows, descarga Windows x86-64 executable installer. De lo contrario, descarga Windows x86 executable installer. Después de descargar el instalador, deberías ejecutarlo (dándole doble click) y seguir las instrucciones.

Una cosa para tener en cuenta: Durante la instalación, verás una ventana de "Setup". Asegúrate de marcar las casillas "Add Python 3.6 to PATH" o "Add Python to your environment variables" y hacer click en "Install Now", como se muestra aquí (puede que se vea un poco diferente si estás instalando una versión diferente):

Cuando la instalación se complete, verás un cuadro de diálogo con un enlace que puedes seguir para saber más sobre Python o sobre la versión que has instalado. Cierra o cancela ese dialogo – ¡Aprenderás más en ese tutorial!

Nota: si estás usando una versión anterior de Windows (7, Vista o cualquier versión anterior) y el instalador de la versión 3.6.x de Python falla con un error, intenta también:

- instalar todas las actualizaciones de Windows e intenta instalar Python de nuevo; o
- instalar una versión de Python anterior (<https://www.python.org/downloads/windows/>), por ejemplo, 3.4.6 (<https://www.python.org/downloads/release/python-346/>).

Nota Antes de instalar Python en OS X, debes asegurarte de que la configuración del Mac permita instalar paquetes que no estén en la App Store. ve a preferencias del sistema (System Preferences, está en la carpeta Aplicaciones), da click en "Seguridad y privacidad" (Security & Privacy) y luego la pestaña "General". Si tu "Permitir aplicaciones descargadas desde:" (Allow apps downloaded from:) está establecida a "Mac App Store," cambia a "Mac App Store y desarrolladores identificados." (Mac App Store and identified developers)

Necesitas ir a la página web <https://www.python.org/downloads/release/python-361/> y descargar el instalador de Python:

- Descarga el archivo Mac OS X 64-bit/32-bit installer,
- Doble click en python-3.6.1-macosx10.6.pkg para ejecutar el instalador.

### 2.5.2 Instalar Python: Linux

Es muy posible que ya tengas instalado Python de serie. Para verificar que ya lo tienes instalado (y qué versión es), abre una consola y escribe el siguiente comando:

```
$ python3 --version
Python 3.6.1
```

Si tienes instalada una versión diferente de Python, al menos 3.4.0 (por ejemplo 3.6.0), entonces no tienes que actualizar. Si tu no has instalado Python, o si tu quieres una versión diferente, primero verifica que distribución de Linux estás usando con el siguiente comando:

```
$ grep ^NOMBRE= /etc/os-release
```

Después, dependiendo de el resultado, sigue una de las siguientes guías de instalación bajo ésta sección.

### 2.5.3 Instalar Python: Debian o Ubuntu

Escribe este comando en tu consola:

```
$ sudo apt install python3
```

### 2.5.4 Instalar Python: Fedora

Usa este comando en tu consola:

```
$ sudo dnf install python3
```

Si estás en versiones antiguas de Fedora, puedes obtener un error que el comando dnf no se encuentra. En ese caso, necesitas usar yum en su lugar.

### 2.5.5 Instalar Python: openSUSE

Verifica si la instalación fue exitosa abriendo una terminal o consola, y corriendo el comando python3:

```
$ python3 --version
Python 3.6.1
```

La versión mostrada puede ser diferente desde 3.6.1 – debería marcar la versión que instalaste.

NOTA: Si está en Windows y recibe un mensaje de error que indica que python3 no se encontró, intente usar python (sin el 3) y compruebe si todavía podría ser una versión de Python que sea 3.4.0 o superior. Si eso tampoco funciona, puede abrir un nuevo símbolo

del sistema e intentar nuevamente; Esto sucede si usa un símbolo del sistema abierto antes de la instalación de Python.

Si tienes alguna duda, o si ocurrió algún error y no tienes idea sobre qué hacer, ¡por favor pregunta a tu entrenador! Algunas veces las cosas no van bien y es mejor pedir ayuda a alguien con más experiencia.

### 2.5.6 Instala un Editor de Código

Hay muchos editores diferentes y la elección es principalmente una cuestión de preferencia personal. La mayoría de programadores de Python usan IDEs (Entornos de Desarrollo Integrados) complejos pero muy potentes, como PyCharm. Sin embargo, como principiante, probablemente no es lo más aconsejable; nuestras recomendaciones son igualmente potentes pero mucho más simples.

Abajo presentamos nuestras sugerencias pero también puedes preguntarle a tu mentora cuáles son las tuyas -será más fácil que te ayude.

### 2.5.7 Visual Studio Code

Visual Studio Code es un recurso de edición de código desarrollado por Microsoft para Windows, Linux y macOS. Esto incluye soporte para depuración, control de Git incrustado, sintaxis destacada, completación de código inteligente, fragmentos y refactorización de código.

### 2.5.8 Gedit

Gedit es un editor gratuito de código abierto, disponible para todos los sistemas operativos.

### 2.5.9 Sublime Text

Sublime Text es un editor muy popular con un periodo de prueba gratis, y está disponible para todos los sistemas operativos.

### 2.5.10 Atom

Atom es otro editor popular. Es gratis, de código abierto y disponible para Windows, OS X and Linux. Atom está desarrollado por Github.

## 2.6 ¿Por qué estamos instalando un editor de código?

Tú podrías estar preguntándote por qué estamos instalando este especializado software editor de código en vez de usar algo como Word o Notepad.

La primera razón es que el código necesita ser texto plano, y el problema con programas como Word y Textedit es que no producen texto plano, sino texto enriquecido (con fuentes y formatos), usando formatos personalizados como RTF ( Formato del Texto Enriquecido, del inglés Rich Text Format).

La segunda razón es que los editores de código están especializados para esta función, así ellos pueden proveer ayuda a características como destacar código con color acorde a su significado, o automáticamente cerrando etiquetas para ti.

Veremos todo esto en acción más adelante. Pronto pensarás en convertir el editor de código en una de tus herramientas favoritas. :)

## 2.7 Configura el entorno virtual (virtualenv) e instala Django

Parte de esta sección está basada en tutoriales por Geek Girls Carrots (<https://github.com/ggcarrots/django-carrots>)

Parte de este capítulo está basada en el django-marcador tutorial bajo la licencia Creative Commons Attribution-ShareAlike 4.0 internacional. El tutorial de django-marcador tiene derechos de autor de Markus Zapke-Gündemann et al.

## 2.8 Entorno virtual

Antes de instalar Django, instalaremos una herramienta extremadamente útil que ayudará a mantener tu entorno de desarrollo ordenado en tu computadora. Es posible saltarse este paso, pero es altamente recomendable. ¡Empezar con la mejor configuración posible te ahorrará muchos problemas en el futuro!

Así que, vamos a crear un entorno virtual (también llamado un virtualenv). Virtualenv aísla tu configuración de Python/Django para cada proyecto. Esto quiere decir que cualquier cambio que hagas en un sitio web no afectará a ningún otro que estés desarrollando. Genial, ¿no?

Todo lo que necesitas hacer es encontrar un directorio en el que quieras crear el virtualenv; tu directorio home, por ejemplo. En Windows, puede verse como C:\Users\Name (donde Name es el nombre de tu usuario).

NOTA: En Windows, asegúrate de que este directorio no contiene caracteres especiales o acentuados; si tu nombre de usuario contiene caracteres acentuados, usa un directorio distinto, por ejemplo C:\djangugirls.

Para este tutorial usaremos un nuevo directorio djangugirls en tu directorio home:

```
$ mkdir djangugirls
$ cd djangugirls
```

Haremos un virtualenv llamado myenv. El comando general estará en el formato:

```
$ python3 -m venv myenv
```

### 2.8.1 Virtual environment: Windows

Para crear un nuevo virtualenv, necesitas abrir una terminal "command prompt" y ejecutar

```
python -m venv myenv
```

Se verá así:

```
C:\Users\Name\djangugirls> python -m venv myenv
```

Donde myenv es el nombre de tu virtualenv. Puedes utilizar cualquier otro nombre, pero asegúrate de usar minúsculas y no usar espacios, acentos o caracteres especiales. También es una buena idea mantener el nombre corto. ¡Vas utilizarlo muchas veces!

### 2.8.2 Virtual environment: Linux and OS X

Podemos crear un virtualenv en Linux y OS X, es tan sencillo como ejecutar

```
python3 -m venv myenv
```

Se verá así:

myvenv es el nombre de tu virtualenv. Puedes usar cualquier otro nombre, pero sólo utiliza minúsculas y no incluyas espacios. También es una buena idea mantener el nombre corto. ¡Vas a referirte muchas veces a él!

NOTA: En algunas versiones de Debian/Ubuntu, puede que obtengas el siguiente error:

```
The virtual environment was not created successfully because
ensurepip is not available.
```

En sistemas Debian/Ubuntu, tendrás que instalar el paquete python3-venv usando el siguiente comando.

```
apt-get install python3-venv
```

Puede que tengas que usar sudo con este comando. Después de instalar el paquete python3-venv, vuelve a crear tu entorno virtual.

En este caso, sigue las instrucciones anteriores e instala el paquete python3-venv:

```
$ sudo apt install python3-venv
```

NOTA: En algunas versiones de Debian/Ubuntu inicializar el entorno virtual de esta manera da el siguiente error:

```
Error: Command '['/home/eddie/Slask/tmp/venv/bin/python3', '-Im',
'ensurepip', '--upgrade', '--default-pip']' returned non-zero exit
status 1
```

Para evitar esto, utiliza directamente el comando virtualenv.

```
$ sudo apt-get install python-virtualenv
$ virtualenv --python=python3.6 myvenv
```

NOTA: Si obtienes un error como

```
E: Unable to locate package python3-venv
```

entonces ejecuta:

```
sudo apt install python3.6-venv
```

## 2.9 Trabajar con virtualenv

El comando anterior creará un directorio llamado myvenv (o cualquier nombre que hayas elegido) que contiene nuestro entorno virtual (básicamente un montón de archivos y carpetas).

### 2.9.1 Working with virtualenv: Windows

Inicia el entorno virtual ejecutando:

```
C:\Users\Name\djangogirls> myvenv\Scripts\activate
```

Nota: en 10 de Windows puedes obtener un error en Windows PowerShell que dice `execution of scripts is disabled on this system`. En este caso, abre otro Windows PowerShell con la opción "Ejecutar como administrador". Luego intenta escribir el siguiente comando antes de inicializar tu entorno virtual:

```
C:\WINDOWS\system32 > Set-ExecutionPolicy -ExecutionPolicy
RemoteSigned Execution Policy Change The execution policy helps
protect you from scripts that you do not trust. Changing the
execution policy might expose you to the security risks
described in the about_Execution_Policies help topic at
http://go.microsoft.com/fwlink/?LinkID=135170.
```

```
Do you want to change the execution policy?
[Y] Yes [A] Yes to All [N] No [L]
No to All [S] Suspend [?] Help (default is "N"): A
```

NOTA: Para usuarios del popular editor VS Code, el cual viene con una terminal integrada basada en el powershell de windows; si deseas añadir la terminal integrada, puedes arrancar el siguiente comando para activar tu entorno virtual:

```
$ . myvenv\Scripts\activate.ps1
```

La ventaja es que no tienes que cambiar las ventanas entre el editor de código y la línea de comandos.

### 2.9.2 Working with virtualenv: Linux and OS X

Inicia el entorno virtual ejecutando:

```
$ source myvenv/bin/activate
```

¡Recuerda reemplazar myvenv con tu nombre de virtualenv que hayas elegido!

NOTA: a veces source podría no estar disponible. En ese caso trata hacerlo de esta forma

```
command-line
```

```
$ . myvenv/bin/activate
```

Sabrás que tienes virtualenv iniciado cuando veas que la línea de comando en tu consola tiene el prefijo (myvenv).

Cuando trabajes en un entorno virtual, python automáticamente se referirá a la versión correcta, de modo que puedes utilizar python en vez de python3.

Ok, tenemos todas las dependencias importantes en su lugar. ¡Finalmente podemos instalar Django!

## 2.10 Instalar Django

Ahora que tienes tu virtualenv iniciado, puedes instalar Django.

Antes de hacer eso, debemos asegurarnos que tenemos la última versión de pip, el software que utilizamos para instalar Django:

```
(myvenv) ~$ python -m pip install --upgrade pip
```

### 2.10.1 Instalar paquetes con un fichero de requisitos (requirements)

Un fichero de requisitos (requirements) tiene una lista de dependencias que se deben instalar mediante pip.

Primero crea un archivo requirements.txt dentro de tu directorio.djangogirls, usando el editor de código que instalaste previamente. Lo puedes hacer mediante abriendo un nuevo archivo en el editor de código y guardándolo como requirements.txt en el directorio.djangogirls. Tu directorio se verá así:

```
djangogirls
requirements.txt
```

Dentro del fichero.djangogirls/requirements.txt deberías tener el siguiente texto:

```
Django~=2.2.4
```

Ahora, ejecuta pip install -r requirements.txt para instalar Django.

```
(myvenv) ~$ pip install -r requirements.txt
Collecting Django~=2.2.4 (from -r requirements.txt (line 1))
  Downloading Django-2.2.4-py3-none-any.whl (7.1MB)
Installing collected packages: Django
Successfully installed Django-2.2.4
```

### 2.10.2 Installing Django: Windows

Si obtienes un error cuando llamas al pip en la plataforma de Windows, por favor verifica si el nombre de ruta de tu proyecto contiene espacios, acentos o caracteres especiales (por ejemplo, C:\Usuarios\NombreUsuario\djangogirls). Si los tiene, utiliza otro en su lugar sin espacios, acentos o caracteres especiales (sugerencia: C:\djangogirls). Crea un nuevo virtualenv en el nuevo directorio, luego borra el viejo y trata de escribir el comando anterior otra vez. (Moviendo el directorio virtualenv no funcionará debido a que usa rutas absolutas.)



### 2.10.3 Installing Django: Windows 8 and Windows 10

Tu línea de comando puede congelarse luego de intentar instalar Django. Si esto sucede, usa

```
C:\Users\Name\djangoirls> python -m pip install -r requirements.txt
```

### 2.10.4 Installing Django: Linux

Si obtienes un error cuando llamas pip en Ubuntu 12.04, por favor corre `python -m pip install -U --force-reinstall pip` para reparar la instalación de pip en el virtualenv.

¡Eso es todo! Ahora estás lista (por fin) para crear una aplicación Django!

## 2.11 Instalar Git

Git es un "sistema de control de versiones" que utilizan muchos programadores. Este software puede seguir los cambios realizados en archivos a lo largo del tiempo de forma que más tarde puedas volver a cualquier versión anterior. Algo similar a la herramienta de "Control de Cambios" en los programas de tipo Word (por ejemplo, Microsoft Word o LibreOffice Writer), pero mucho más potente.

### 2.11.1 Installing Git: Windows

Puedes descargar Git desde <https://git-scm.com/>. Puedes hacer click en "Next" en todos los pasos excepto en dos: cuando se te pregunte que selecciones tu editor, selecciona Nano, y en el paso "adjusting your PATH environment", selecciona "Use Git and optional Unix tools from the Windows Command Prompt" (la última opción). Aparte de eso, los valores por defecto son correctos. "Checkout Windows-style, commit Unix-style line endings" tampoco necesita corrección.

No olvides reiniciar el Símbolo del Sistema o el PowerShell una vez que la instalación se complete con éxito.

### 2.11.2 Installing Git: OS X

Descarga Git de <https://git-scm.com> y sigue las instrucciones.

Nota Si estas usando OS X 10.6, 10.7 o 10.8, tendrás que instalar git desde aquí: Git installer for OS X Snow Leopard (<https://sourceforge.net/projects/git-osx-installer/files/git-2.3.5-intel-universal-snow-leopard.dmg/download>)

### 2.11.3 Installing Git: Debian or Ubuntu

```
$ sudo apt install git
```

### 2.11.4 Installing Git: Fedora

```
$ sudo dnf install git
```

### 2.11.5 Installing Git: openSUSE

```
$sudo zypper install git
```

## 2.12 Crear una cuenta de GitHub

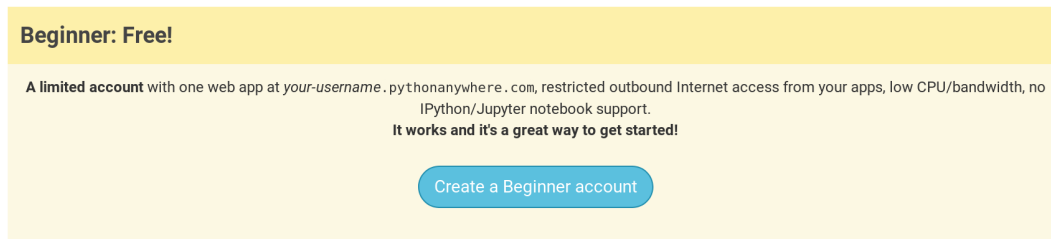
Ve a GitHub.com y regístrate con una nueva y gratuita cuenta de usuario. Asegúrate de recordar tu contraseña (añádela a tu gestor de contraseñas, si usas uno).

## 2.13 Crear una cuenta de PythonAnywhere

PythonAnywhere es un servicio para ejecutar código Python en servidores "en la nube". Lo vamos a usar para alojar nuestro sitio para que esté disponible en Internet.

Almacenaremos del blog que estamos construyendo sobre Python Anywhere. Crea una cuenta como "Principiante/Beginner" en Python Anywhere (el modo gratuito está bien, no necesitas una tarjeta de crédito).

### Plans and pricing



**Beginner: Free!**

A limited account with one web app at `your-username.pythonanywhere.com`, restricted outbound Internet access from your apps, low CPU/bandwidth, no IPython/Jupyter notebook support.

It works and it's a great way to get started!

Create a Beginner account

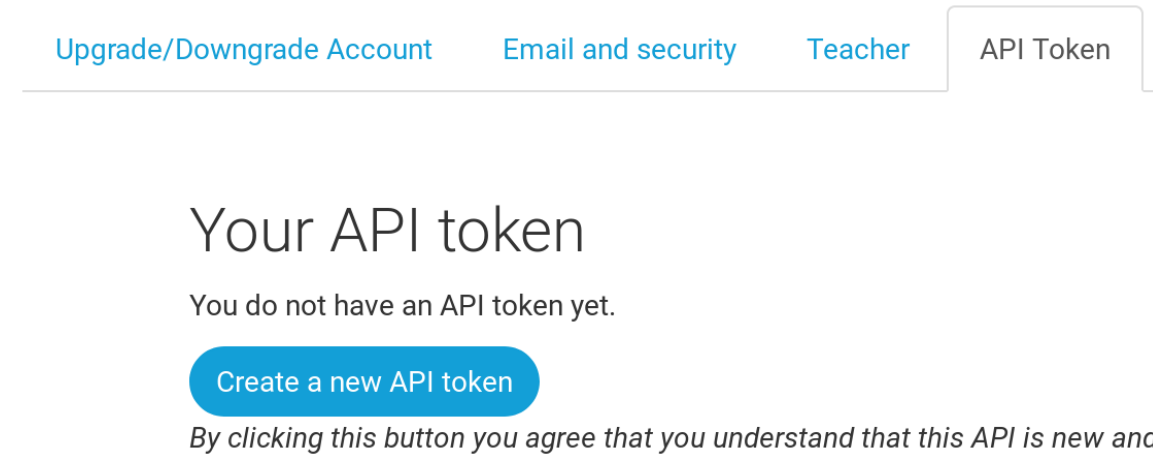
Nota Cuando elijas un nombre de usuario, recuerda que la URL de tu blog tendrá la forma `tunombredeusuario.pythonanywhere.com`, así que lo mejor será usar tu apodo o elegir un nombre que indique de qué trata tu blog. Asegúrate también de recordar tu contraseña (añádela a tu gestor de contraseñas, si usas uno).

### 2.13.1 Crear un token para la API de PythonAnywhere

Este paso solo necesita ser completado una vez. Una vez que te hayas registrado en PythonAnywhere, serás llevado a tu tablón principal. Encontrarás el enlace a la página de tu "Cuenta" en la esquina superior derecha:

Enlace de cuenta en la parte superior derecha de la página

Después selecciona la lengüeta llamada "API token", y haz click en el botón que dice "Crear nueva API token"



### 2.13.2 Comienza a leer

Felicitaciones, ¡tú tienes todo configurado y listo para avanzar! si aún tienes tiempo antes del taller, sería útil empezar a leer un poco de los capítulos iniciales:

- ¿Cómo funciona Internet? ([https://tutorial.djangogirls.org/es/how\\_the\\_internet\\_works/](https://tutorial.djangogirls.org/es/how_the_internet_works/))
- Introducción a la línea de comandos ([https://tutorial.djangogirls.org/es/intro\\_to\\_command\\_line/](https://tutorial.djangogirls.org/es/intro_to_command_line/))
- Introducción a Python ([https://tutorial.djangogirls.org/es/python\\_introduction/](https://tutorial.djangogirls.org/es/python_introduction/))
- ¿Qué es Django? (<https://tutorial.djangogirls.org/es/django/>)

### 2.13.3 ¡Disfruta el taller!

Cuando comiences el taller, estarás habilitada para ir directamente a ¡Tu primer proyecto en Django! porque ya cubriste el material en los capítulos anteriores.

## Index

Instalación .....	4
Introducción .....	1