



códigofacilito



Introducción a Django.

Camila - Backend developer





>_ Temas

- ¿Qué es Django?
- Instalación.
- Mi primer proyecto.
- Mi primera aplicación.
- Patrones de Diseño (MVC vs MTV).





>_ ¿Qué es Django?.

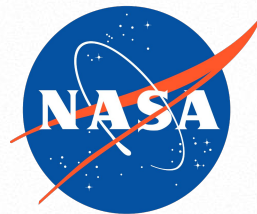




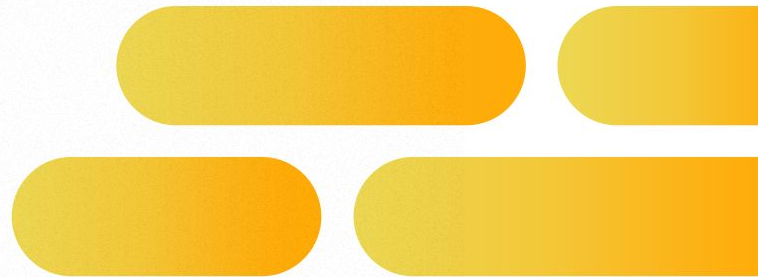
>_ Django es un Framework web de alto nivel desarrollado en Python que fomenta un desarrollo rápido y un diseño limpio y pragmático, gratis y de código abierto.



☰ ¿Quién usa Django?



The New York Times



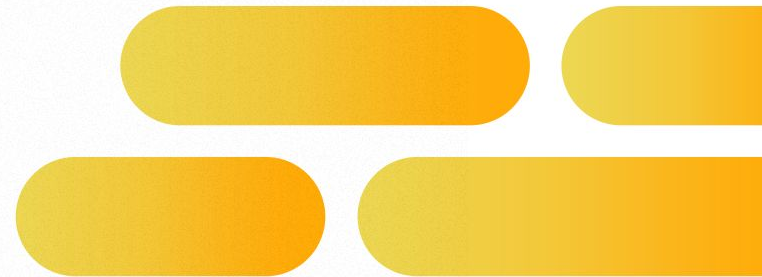


>_ Instalación.



☰ ¿Cómo Instalar Django?

```
>_ python -m venv env  
>_ source venv/bin/activate  
>_ pip install django  
>_ python -m django --version
```

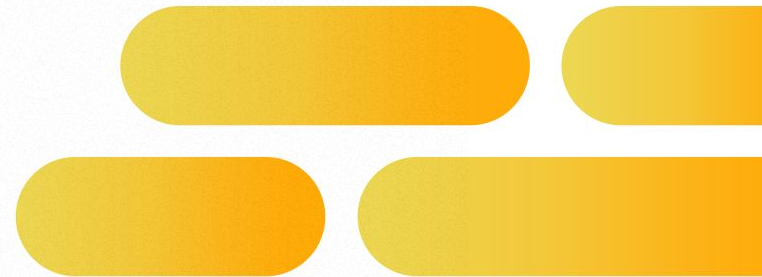


Django-cli

>_ django-admin es la utilidad de línea de comandos de Django para tareas administrativas.

```
>_ django-admin startproject myproject .
```

```
>_ python manage.py runserver
```



Migraciones

> Las migraciones son una manera incremental que tenemos para actualizar el esquema o como esta formada la base de datos.

Django ya viene con una serie de migraciones listas para aplicar pero tu puedes crear las tuyas a medida que vas construyendo tu proyecto.

> `python manage.py makemigrations`

> `python manage.py migrate`

> `python manage.py showmigrations`



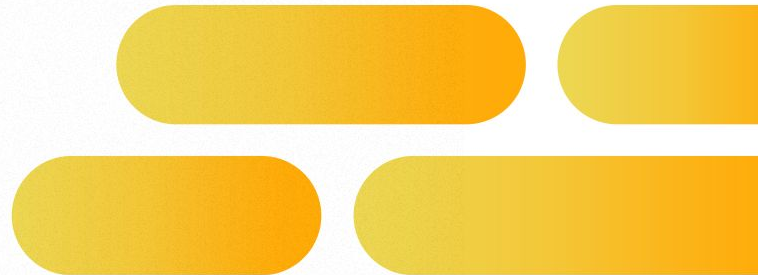


>_ Términos importantes

Django Project: Es una aplicación web impulsada por Django Framework, también considerado como el punto de entrada de nuestro sitio web.

Django Apps: Son pequeñas librerías diseñadas para representar un aspecto único del proyecto. Un proyecto de Django está construido por múltiples Django Apps, algunas de ellas son construidas dentro del proyecto y otras pueden venir de terceros.

Manage.py: El script de gestión del proyecto.





>_ Mi primer proyecto.





Repository





Tienda De Barrio

TiendaDeBarrio

settings

products

inventory

shopping
Cart

manage.py

Requeriments.txt

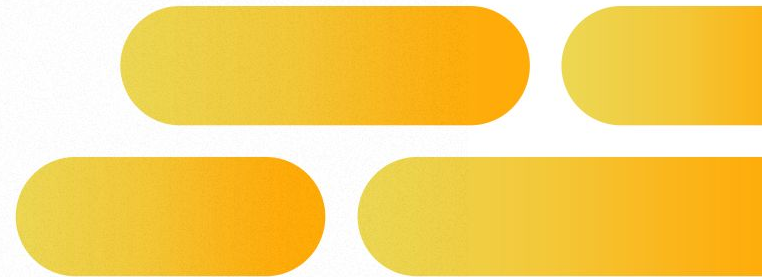
.gitignore

README.md



Crear una App

```
>_ python manage.py startapp tiendadebarrio
```





>_ Patrones de diseño

Los patrones de diseño o *design patterns*, son una solución general, reutilizable y aplicable a diferentes problemas de diseño de software. Se trata de **plantillas que identifican problemas en el sistema y proporcionan soluciones apropiadas a problemas generales** a los que se han enfrentado los desarrolladores durante un largo periodo de tiempo, a través de prueba y error.





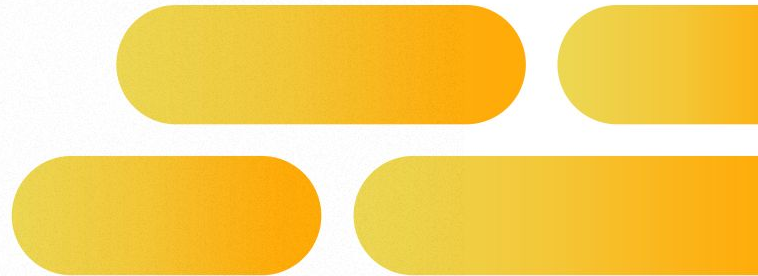
>_ Model View Template (MVT)

En Django, el controlador sigue estando presente, nada más que de una manera intrínseca, ya que todo el framework Django es el controlador.

Model: Maneja todo lo relacionado con la información, esto incluye cómo acceder a esta, la validación, relación entre los datos y su comportamiento.

Template: Decide cómo será mostrada la información.

View: Es un enlace entre el modelo y el template. Decide qué información será mostrada y por cual template.



Model Template Vista (MTV)

