

# Primera parte

**Proyecto:** Skateshop

**Lenguaje:** Python

**Framework:** Django

**Editor:** VS Code

---

## 1 Procedimiento para crear carpeta del Proyecto:

UIII\_Skateshop\_0402

## 2 Procedimiento para abrir VS Code sobre la carpeta

UIII\_Skateshop\_0402

## 3 Procedimiento para abrir terminal en VS Code

## 4 Procedimiento para crear carpeta entorno virtual “.venv” desde terminal de VS Code

## 5 Procedimiento para activar el entorno virtual

## 6 Procedimiento para activar intérprete de Python

## 7 Procedimiento para instalar Django

## 8 Procedimiento para crear proyecto

backend\_Skateshop (sin duplicar carpeta)

## 9 Procedimiento para ejecutar servidor en el puerto 8036

## 10 Procedimiento para copiar y pegar el link en el navegador

## 11 Procedimiento para crear aplicación

app\_Skateshop

## 12 Aquí el modelo [models.py](#)

```
from django.db import models
```

```
# =====
```

```

# MODELO: CATEGORÍA
# =====
class Categoria(models.Model):
    nombre = models.CharField(max_length=50, unique=True)
    descripcion = models.TextField(blank=True, null=True)
    fecha_creacion = models.DateField(auto_now_add=True)
    activo = models.BooleanField(default=True)
    popularidad = models.PositiveIntegerField(default=0)
    icono = models.CharField(max_length=100, blank=True, null=True)

    def __str__(self):
        return self.nombre

# =====
# MODELO: PRODUCTO
# =====
class Producto(models.Model):
    nombre = models.CharField(max_length=100)
    descripcion = models.TextField(blank=True, null=True)
    precio = models.DecimalField(max_digits=10, decimal_places=2)
    stock = models.PositiveIntegerField(default=0)
    categoria = models.ForeignKey(Categoria,
on_delete=models.CASCADE, related_name="productos")
    marca = models.CharField(max_length=50)
    modelo = models.CharField(max_length=50, blank=True, null=True)
    fecha_agregado = models.DateField(auto_now_add=True)
    disponible = models.BooleanField(default=True)
    imagen = models.URLField(blank=True, null=True)

    def __str__(self):
        return f"{self.nombre} - {self.marca}"

```

## 12.5 Procedimiento para realizar las migraciones (makemigrations y migrate)

### 13 Primero trabajamos con el MODELO: CATEGORÍA

14 En `views.py` de `app_Skateshop` crear las funciones con sus códigos correspondientes:

`inicio_skateshop`, `agregar_categoria`, `actualizar_categoria`,  
`realizar_actualizacion_categoria`, `borrar_categoria`.

15 Crear la carpeta `templates` dentro de `app_Skateshop`.

16 En la carpeta `templates` crear los archivos HTML:

`base.html`, `header.html`, `navbar.html`, `footer.html`, `inicio.html`.

17 En el archivo `base.html` agregar Bootstrap para CSS y JS.

18 En el archivo `navbar.html` incluir las opciones:

"Sistema de Administración Skateshop",  
"Inicio", "Categorías" (con submenús: *Agregar Categoría*, *Ver Categorías*, *Actualizar Categoría*, *Borrar Categoría*),  
"Productos" (con submenús: *Agregar Producto*, *Ver Productos*, *Actualizar Producto*, *Borrar Producto*).  
Incluir iconos en las opciones principales (no en los submenús).

19 En el archivo `footer.html` incluir:

Derechos de autor, fecha del sistema y "Creado por  
Alejandro Jesus Villagrana Bojorquez, CBTis 128",  
manteniéndolo fijo al final de la página.

20 En el archivo `inicio.html` colocar información del sistema más una imagen tomada desde la red sobre Skateshop o Skateboarding.

21 Crear la subcarpeta `categoria` dentro de

`app_Skateshop/templates/`

22 Crear los archivos HTML con su código correspondiente dentro de `app_Skateshop/templates/categoria`:

- `agregar_categoria.html`

- `ver_categorias.html` (mostrar en tabla con botones *ver*, *editar* y *borrar*)
- `actualizar_categoria.html`
- `borrar_categoria.html`

(No utilizar `forms.py`)

---

## 23 Procedimiento para crear el archivo `urls.py` en `app_Skateshop`

Con el código correspondiente para acceder a las funciones de `views.py` para operaciones CRUD en categorías.

## 24 Procedimiento para agregar `app_Skateshop` en `settings.py` de `backend_Skateshop`.

## 25 Realizar las configuraciones correspondientes en `urls.py` de `backend_Skateshop`

para enlazar con `app_Skateshop`.

## 26 Procedimiento para registrar los modelos en `admin.py` y volver a realizar las migraciones.

---

## 27 Por lo pronto solo trabajar con "Categoría"; dejar pendiente Producto si aún no se ha terminado.

## 28 Utilizar colores suaves, atractivos y modernos.

Código de las páginas web sencillo.

## 29 No validar entrada de datos.

## 30 Al inicio crear la estructura completa de carpetas y archivos.

**31 Proyecto totalmente funcional.**

**32 Finalmente ejecutar servidor en el puerto 8036.**