

Primera parte

Proyecto: **Delrio**

Lenguaje: **Python.**

Framework: **Django.**

Editor: VS code.

1 Procedimiento para crear carpeta del Proyecto: UIII_Delrio_0459

2 procedimiento para abrir vs code sobre la carpeta

UIII_Delrio_0459

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_Delrio sin duplicar carpeta.

9 procedimiento para ejecutar servidor en el puerto 8459

10 procedimiento para copiar y pegar el link en el navegador.

11 procedimiento para crear aplicación app_Delrio

12 Aquí el modelo models.py

=====

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

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

```
# MODELO: CLIENTE
```

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

```
class Cliente(models.Model):
```

```
    nom_clie = models.CharField(max_length=100)
```

```
    apellido_clie = models.CharField(max_length=100)
```

```
    C_E_clie = models.CharField(max_length=100) # Correo electrónico
```

```
    tel_clie = models.CharField(max_length=20)
```

```
    direc_clie = models.TextField()
```

```
    fech_compra = models.DateField()
```

```
def __str__(self):
```

```
    return f"{self.nom_clie} {self.apellido_clie}"
```

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

```
# MODELO: PRODUCTO
```

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

```
class Producto(models.Model):
```

```
    nom_prod = models.CharField(max_length=100)
```

```
    desc_prod = models.TextField(blank=True, null=True)
```

```
    precio_unidad = models.DecimalField(max_digits=10, decimal_places=2)
```

```
    stock = models.IntegerField()
```

```
    fech_creacion = models.DateTimeField(auto_now_add=True)
```

```
    id_prov = models.IntegerField() # Si existiera tabla 'proveedor', sería ForeignKey
```

```

def __str__(self):
    return self.nom_prod

# =====
# MODELO: VENTA
# =====
class Venta(models.Model):
    fech_venta = models.DateTimeField(auto_now_add=True)
    total_venta = models.DecimalField(max_digits=10, decimal_places=2)
    estado = models.CharField(max_length=50)

    # Relaciones foráneas
    id_clie = models.ForeignKey(Cliente, on_delete=models.CASCADE,
                                related_name="ventas")
    id_prod = models.ForeignKey(Producto, on_delete=models.CASCADE,
                                related_name="ventas")
    id_empl = models.IntegerField() # Si tuvieras tabla Empleado, usarías ForeignKey

    def __str__(self):
        return f"Venta #{self.id} - Cliente: {self.id_clie.nom_clie} - Total: ${self.total_venta}"

```

12.5 Procedimiento para realizar las migraciones(makemigrations y migrate.

13 primero trabajamos con el MODELO: VENTAS

14 En view de app_Delrio crear las funciones con sus códigos correspondientes (inicio_Delrio, agregar_venta,

actualizar_venta, realizar_actualizacion_venta,
borrar_venta)

15 Crear la carpeta “templates” dentro de “app_Delrio”.

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 Delrio”, “Inicio”, “Venta”,en submenu de Venta(Agregar Venta,ver venta, actualizar venta, borrar venta), “Cliente” en submenu de Cliente(Agregar cliente,ver clientes, actualizar cliente, borrar cliente)

“Producto” en submenu de Producto(Aregar productos,ver productos, actualizar producto, borrar producto), incluir iconos a las opciones principales, no en los submenu.

19 En el archivo footer.html incluir derechos de autor,fecha del sistema y “Creado por Gael carrasco, Cbtis 128” y mantenerla fija al final de la página.

20 En el archivo inicio.html se usa para colocar información del sistema más una imagen tomada desde la red sobre Delrio.

- 21 Crear la subcarpeta carpeta Venta dentro de app_Delrio\templates.
- 22 crear los archivos html con su codigo correspondientes de (agregar_venta.html, ver_venta.html mostrar en tabla con los botones ver, editar y borrar, actualizar_venta.html, borrar_venta.html) dentro de app_Delrio\templates\Ventas.
- 23 No utilizar forms.py.
- 24 procedimiento para crear el archivo urls.py en app_Delrio con el código correspondiente para acceder a las funciones de views.py para operaciones de crud en Ventas.
- 25 procedimiento para agregar app_Delrio en settings.py de backend_Delrio
- 26 realizar las configuraciones correspondiente a urls.py de backend_Delrio para enlazar con app_Delrio

- 27 procedimiento para registrar los modelos en admin.py y volver a realizar las migraciones.
- 27 por lo pronto solo trabajar con “Venta” dejar pendiente # MODELO: Cliente y # MODELO: Producto
- 28 Utilizar colores suaves, atractivos y modernos, el código de las páginas web sencillas.
- 28 No validar entrada de datos.
- 29 Al inicio crear la estructura completa de carpetas y archivos.
- 30 proyecto totalmente funcional.
- 31 finalmente ejecutar servidor en el puerto puerto 8459.