

Prompt primera parte

Proyecto: Similares.

lenguaje: Python.

Framework: Django.

Editor: VS code.

1 Procedimiento para crear carpeta del Proyecto: UIII_Similares_0663

2 procedimiento para abrir vs code sobre la carpeta

UIII_Similares_0663

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_Similares sin duplicar carpeta.

9 procedimiento para ejecutar servidor en el puerto 8663

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

11 procedimiento para crear aplicacion app_Similares

12 Aqui el modelo models.py

==--==--==--

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

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

```
# MODELO: Cliente
```

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

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

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

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

```
    email = models.EmailField(unique=True)
```

```
    telefono = models.CharField(max_length=15)
```

```
    direccion = models.CharField(max_length=200)
```

```
    fecha_nacimiento = models.DateField(null=True, blank=True)
```

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

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

```
        return f'{self.nombre} {self.apellido}'
```

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

```
# MODELO: Medicamento
```

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

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

```
    nombre = models.CharField(max_length=150)
```

```
    descripcion = models.TextField()
```

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

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

```
    stock = models.PositiveIntegerField()
```

```
    fecha_vencimiento = models.DateField()
```

```
    codigo_barras = models.CharField(max_length=50, unique=True)
```

```
def __str__(self):
    return self.nombre
```

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

```
# MODELO: Venta
```

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

```
class Venta(models.Model):
    cliente = models.ForeignKey(Cliente, on_delete=models.CASCADE, related_name='ventas') # 1 a muchos
    medicamentos = models.ManyToManyField(Medicamento, related_name='ventas') # muchos a muchos
    fecha_venta = models.DateTimeField(auto_now_add=True)
    total = models.DecimalField(max_digits=10, decimal_places=2)
    metodo_pago = models.CharField(max_length=50, choices=[
        ('EFFECTIVO', 'Efectivo'),
        ('TARJETA', 'Tarjeta'),
        ('TRANSFERENCIA', 'Transferencia'),
    ])
    numero_factura = models.CharField(max_length=20, unique=True)
    observaciones = models.TextField(blank=True, null=True)
    estado = models.CharField(max_length=20, default='COMPLETADA', choices=[
        ('COMPLETADA', 'Completada'),
        ('PENDIENTE', 'Pendiente'),
        ('CANCELADA', 'Cancelada'),
    ])

    def __str__(self):
        return f"Venta {self.numero_factura} - {self.cliente.nombre}"
```

```
==--==--==
```

12.5 Procedimiento para realizar las migraciones(makemigrations y migrate.

13 primero trabajamos con el MODELO: Cliente

14 En view de app_Similares crear las funciones con sus códigos correspondientes (inicio_similares, agregar_cliente,

actualizar_cliente, realizar_actualizacion_cliente, borrar_cliente)

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

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 Similares”, “Inicio”, “cliente”,en submenu de categorias(Agregar Cliente,ver cliente, actualizar cliente, borrar cliente), “Medicamentos” en submenu de Medicamentos(Agregar medicamentos,ver medicamentos, actualizar medicamentos, borrar medicamentos) “Ventas” en submenu de Ventas(Agregar ventas,ver ventas, actualizar ventas, borrar ventas), 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 Abraham Ochoa, 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 Similares.

21 Crear la subcarpeta carpeta cliente dentro de app_Similares\templates.

22 crear los archivos html con su código correspondientes de (agregar_clientes.html, ver_clientes.html mostrar en tabla con los botones ver, editar y borrar, actualizar_clientes.html, borrar_cliente.html)

dentro de app_Similares templates\clientes

23 No utilizar forms.py.

24 procedimiento para crear el archivo urls.py en app_Similares con el código correspondiente para acceder a las funciones de views.py para operaciones de crud en clientes.

25 procedimiento para agregar app_Similares en settings.py de backend_Similares

26 realizar las configuraciones correspondiente a urls.py de backend_Similares para enlazar con app_Similares

27 procedimiento para registrar los modelos en admin.py y volver a realizar las migraciones.

27 por lo pronto solo trabajar con “cliente” dejar pendiente # MODELO: MEDICAMENTO y # MODELO: VENTA

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 8663.