

Primera parte

Proyecto: **Agencia.**

Lenguaje: **Python.**

Framework: **Django.**

Editor: VS code.

1 Procedimiento para crear carpeta del Proyecto: **UIII\_Agencia\_0330** procedimiento para abrir vs code sobre la carpeta

### **UIII\_Agencia\_0330**

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\_Agencia** sin duplicar carpeta.

9 procedimiento para ejecutar servidor en el **puerto 8030**

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

11 procedimiento para crear aplicación **app\_Agencia**

12 Aquí el modelo models.py

=====

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

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

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

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

```
# MODELO: CLIENTES
```

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

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

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

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

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

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

```
    fecha_registro = models.DateField(auto_now_add=True)
```

```
    estatus = models.CharField(max_length=50, default="Activo")
```

```
    tipo_cliente = models.CharField(max_length=50, default="Regular")
```

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

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

```
    return self.nombre_completo
```

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

```
# MODELO: CASAS
```

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

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

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

```

ciudad = models.CharField(max_length=100)
precio = models.DecimalField(max_digits=10, decimal_places=2)
tipo = models.CharField(max_length=50)
estado = models.CharField(max_length=50)
fecha_construccion = models.DateField()
cliente = models.ForeignKey(Cliente, on_delete=models.CASCADE,
related_name="casas")
# Relación 1 a muchos: un cliente puede tener varias casas
numero_registro = models.CharField(max_length=50, unique=True)

def __str__(self):
    return f"{self.tipo} en {self.ciudad}"

# =====
# MODELO: PAGOS
# =====
class Pago(models.Model):
    monto = models.DecimalField(max_digits=10, decimal_places=2)
    fecha_pago = models.DateField()
    metodo = models.CharField(max_length=50)
    referencia = models.CharField(max_length=100, unique=True)
    descripcion = models.TextField(blank=True, null=True)
    comprobante = models.CharField(max_length=100, blank=True, null=True)
    casas = models.ManyToManyField(Casa, related_name="pagos")
    # Relación muchos a muchos: una casa puede tener varios pagos y un pago puede
    # aplicarse a varias casas
    estatus_pago = models.CharField(max_length=50, default="Completado")

    def __str__(self):
        return f"Pago {self.referencia} - ${self.monto}"
```

=====

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

13 Primero trabajamos con el **MODELO: CLIENTES**.

14 En **views** de **app\_Agencia** crear las funciones con sus códigos correspondientes (**inicio\_agencia**, **agregar\_cliente**, **actualizar\_cliente**, **realizar\_actualizacion\_cliente**, **borrar\_cliente**).

15 Crear la carpeta “**templates**” dentro de “**app\_Agencia**”.

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 Agencia**”,

“**Inicio**”, “**Ci**entes****”, en submenú de **Ci**entes**** (**Agregar Cliente**, **Ver Cliente**, **Actualizar Cliente**, **Borrar Cliente**), “**Casas**” en submenú de **Casas** (**Agregar Casa**, **Ver Casas**, **Actualizar Casas**, **Borrar Casas**), “**Pagos**” en submenú de **Pagos** (**Agregar Pago**, **Ver Pagos**, **Actualizar Pagos**, **Borrar Pagos**).

Incluir iconos a las opciones principales, no en los submenús.

19 En el archivo **footer.html** incluir derechos de autor, fecha del sistema y “**Creado por Leonardo Rivera 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 agencia.

21 Crear la subcarpeta **carpeta clientes** dentro de **app\_Agencia\templates**.

22 Crear los archivos **html** con su código correspondiente de (**agregar\_cliente.html**, **ver\_clientes.html** mostrando en tabla con los botones **ver**, **editar** y **borrar**, **actualizar\_cliente.html**, **borrar\_cliente.html**) dentro de **app\_Agencia\templates\clientes**.

23 No utilizar **forms.py**.

24 Procedimiento para crear el archivo **urls.py** en **app\_Agencia** con el código correspondiente para acceder a las funciones de **views.py** para operaciones de **CRUD** en **clientes**.

25 Procedimiento para agregar **app\_Agencia** en **settings.py** de **backend\_Agencia**.

26 Realizar las configuraciones correspondientes a **urls.py** de **backend\_Agencia** para enlazar con **app\_Agencia**.

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

27 Por lo pronto solo trabajar con “**clientes**”, dejar pendiente **# MODELO: CASAS y MODELO: PAGOS**.

28 Utilizar colores suaves, atractivos y modernos; el código de las páginas web debe ser sencillo.

28 No validar entrada de datos.

29 Al inicio crear la estructura completa de carpetas y archivos.

30 Proyecto totalmente funcional.

31 Finalmente ejecutar el servidor en el puerto **8030**.