

Primera parte

Proyecto: **Cinépolis.**

lenguaje: **Python.**

Framework: **Django.**

Editor: **VS code.**

1 Procedimiento para crear carpeta del Proyecto: UIII\_Agencia\_Viajes\_0472

2 procedimiento para abrir vs code sobre la carpeta

UIII\_Agencia\_Viajes\_0472

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 8472

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

11 procedimiento para crear aplicacion app\_clientes

12 Aqui el modelo models.py

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

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

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

```
# MODELO: clientes
```

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

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

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

```
    correo = models.EmailField(max_length=254, blank=True, null=True)
```

```
    telefono = models.CharField(max_length=30, blank=True, null=True)
```

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

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

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

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

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

```
# MODELO: viajes
```

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

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

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

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

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

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

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

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

```

transporte = models.ForeignKey(Transporte, on_delete=models.SET_NULL, blank=True,
null=True, related_name='viajes')
cliente = models.ForeignKey(Cliente, on_delete=models.SET_NULL, blank=True,
null=True, related_name='viajes')

```

```

def __str__(self):
    return f"{self.nombre_viaje} - {self.destino}"

```

```

# =====

```

```

# MODELO: transportes

```

```

# =====

```

```

class Transporte(models.Model):
    nombre_transporte = models.CharField(max_length=150)
    descripcion = models.TextField(blank=True, null=True)
    tipo = models.CharField(max_length=50, blank=True, null=True)
    capacidad = models.PositiveIntegerField(blank=True, null=True)
    precio = models.DecimalField(max_digits=10, decimal_places=2, blank=True, null=True)
    horario = models.TimeField(blank=True, null=True)
    empresa = models.CharField(max_length=150, blank=True, null=True)

```

```

def __str__(self):
    return self.nombre_transporte

```

```

==--==--==

```

12.5 Procedimiento para realizar las migraciones(makemigrations y migrate.

13 primero trabajamos con el MODELO: clientes

14 En view de app\_clientes crear las funciones con sus códigos correspondientes (inicio\_agencia, agregar\_clientes,

actualizar\_clientes, realizar\_actualizacion\_clientes, borrar\_clientes)

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

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”, “clientes”,en submenu de clientes(Agregar clientes,ver clientes, actualizar clientes, borrar clientes), “viajes” en submenu de viajes(Aregar viajes,ver viajes, actualizar viajes, borrar viajes)

“transporte” en submenu de transporte(Aregar transporte,ver transporte, actualizar transporte, borrar transporte), 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 Yared Coss, 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 de viajes.

- 21 Crear la subcarpeta carpeta clientes dentro de app\_clientes\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\_clientes.html) dentro de app\_clientes\templates\clientes
- 23 No utilizar forms.py.
- 24 procedimiento para crear el archivo urls.py en app\_clientes con el código correspondiente para acceder a las funciones de views.py para operaciones de crud en clientes
- 25 procedimiento para agregar app\_clientes en settings.py de backend\_agencia
- 26 realizar las configuraciones correspondiente a urls.py de backend\_agencia para enlazar con app\_clientes
- 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: viajes y # MODELO: transporte
- 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 8472