

Primera parte

Proyecto: **Panadería.**

lenguaje: **Python.**

Framework: **Django.**

Editor: **VS code.**

1 Procedimiento para crear carpeta del Proyecto: **UIII\_Panadería\_0501**

2 procedimiento para abrir vs code sobre la carpeta

**UIII\_Panadería\_0501**

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\_Panaderia** sin duplicar carpeta.

9 procedimiento para ejecutar servidor en el **puerto 8014**

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

11 procedimiento para crear aplicacion **app\_Panaderia**

12 Aqui el modelo models.py

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

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

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

```
# MODELO: Ingrediente
```

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

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

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

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

```
    unidad_medida = models.CharField(max_length=50)
```

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

```
    proveedor = models.CharField(max_length=100, blank=True, null=True)
```

```
    fecha_compra = models.DateField()
```

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

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

```
        return self.nombre
```

```
    class Meta:
```

```
        verbose_name = "Ingrediente"
```

```
        verbose_name_plural = "Ingredientes"
```

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

```
# MODELO: Receta
```

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

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

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

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

```
    tiempo_preparacion = models.IntegerField(help_text="Tiempo en minutos")
```

```
    fecha_creacion = models.DateField()
```

```

autor = models.CharField(max_length=100)
ingredientes = models.ManyToManyField(Ingrediente, related_name='recetas')

def __str__(self):
    return self.nombre

class Meta:
    verbose_name = "Receta"
    verbose_name_plural = "Recetas"
# =====
# MODELO: Producto
# =====
class Producto(models.Model):
    nombre = models.CharField(max_length=100)
    descripcion = models.TextField(blank=True, null=True)
    precio_venta = models.DecimalField(max_digits=10, decimal_places=2)
    fecha_elaboracion = models.DateField()
    fecha_vencimiento = models.DateField()
    receta = models.ForeignKey(Receta, on_delete=models.CASCADE,
related_name='productos')
    def __str__(self):
        return f"{self.nombre} (${self.precio_venta})"
    class Meta:
        verbose_name = "Producto"
        verbose_name_plural = "Productos"
==--==

```

12.5 Procedimiento para realizar las migraciones(makemigrations y migrate.

13 primero trabajamos con el MODELO: Ingrediente

14 En view de app\_Panaderia crear las funciones con sus códigos correspondientes (inicio\_panaderia, agregar\_ingredient,

actualizar\_ingredient, realizar\_actualizacion\_ingredient, borrar\_ingredient)

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

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 Panaderia”, “Inicio”, “categoria”,en submenu de categorias(Agregar Ingrediente, ver Ingrediente, actualizar Ingrediente, borrar Ingrediente), “Receta” en submenu de Recetas(Agregar receta,ver recetas, actualizar receta, borrar receta)

“Producto” en submenu de Producto(Agregar producto,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 Téc. Eliseo García, 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 panadería.

21 Crear la subcarpeta carpeta ingrediente dentro de app\_Panaderia\templates.

22 crear los archivos html con su código correspondientes de (agregar\_categoria.html, ver\_categorias.html mostrar en tabla con los botones ver, editar y borrar, actualizar\_categoria.html, borrar\_categoria.html) dentro de app\_Panaderia\templates\ingredientes.

23 No utilizar forms.py.

24 procedimiento para crear el archivo urls.py en app\_Panaderia con el código correspondiente para acceder a las funciones de views.py para operaciones de crud en ingredientes.

25 procedimiento para agregar app\_Panaderia en settings.py de backend\_Panaderia

26 realizar las configuraciones correspondiente a urls.py de backend\_Panaderia para enlazar con app\_Panaderia

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

27 por lo pronto solo trabajar con “ingrediente” dejar pendiente # MODELO: receta 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 8014.