

UIII Act4 Segunda Parte - Tienda de Abarrotes

Proyecto: Tienda de Abarrotes

Lenguaje: Python

Framework: Django

Editor: VS Code

1. Aquí el modelo models.py que ya está creado

```
# =====  
# MODELO: CATEGORÍA  
# =====  
class Categoria(models.Model):  
    nombre = models.CharField(max_length=100, unique=True)  
    descripcion = models.TextField(blank=True, null=True)  
    pasillo = models.IntegerField(blank=True, null=True)  
    responsable_area = models.CharField(max_length=100, blank=True, null=True)  
    fecha_creacion = models.DateField(auto_now_add=True)  
    activa = models.BooleanField(default=True)  
    def __str__(self):  
        return self.nombre  
# =====
```

2. Procedimiento para realizar las migraciones (makemigrations y migrate)

3. Trabajar con el **MODELO: CATEGORÍA**.
4. En **views.py** de **app_Tienda** crear las funciones correspondientes (*agregar_categoria, actualizar_categoria, realizar_actualizacion_categoria, borrar_categoria*).
5. Modificar el archivo **navbar.html** dentro de **templates** para actualizar la opción “**Categorías**” con submenús (*Agregar, Ver, Actualizar, Borrar*).
6. Crear la subcarpeta **categoria** dentro de **app_Tienda/templates**.
7. Crear los archivos HTML correspondientes:
 - `agregar_categoria.html`
 - `ver_categorias.html`
 - `actualizar_categoria.html`
 - `borrar_categoria.html`
8. No utilizar **forms.py**.
9. Agregar las rutas en **urls.py** de **app_Tienda** para las funciones CRUD.
10. Registrar los modelos en **admin.py** y volver a realizar las migraciones.
11. Por lo pronto solo trabajar con “**Categoría**”, dejar pendiente los modelos **Producto** y **Proveedor**.
12. Utilizar **colores suaves, atractivos y modernos**.
13. **No validar la entrada de datos**.
14. Al inicio crear la estructura completa de carpetas y archivos actualizados.
15. **Proyecto totalmente funcional**.
16. Finalmente, **ejecutar el servidor en el puerto 8033**.