

Primera parte

Proyecto: Xbox.

Lenguaje: Python.

Framework: Django.

Editor: VS Code.

1 Procedimiento para crear carpeta del Proyecto:

UIII_Xbox_0538

2 Procedimiento para abrir VS Code sobre la carpeta

UIII_Xbox_0538

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_Xbox sin duplicar carpeta.

9 Procedimiento para ejecutar servidor en el puerto 8036

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

11 Procedimiento para crear aplicación

app_Xbox

12 Aquí el modelo models.py

==--==--==

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

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

```
# MODELO: PROVEEDOR
```

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

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

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

```

    contacto = models.CharField(max_length=100)
    telefono = models.CharField(max_length=20)
    email = models.CharField(max_length=100)
    direccion = models.CharField(max_length=200)
    activo = models.BooleanField(default=True)
    fecha_registro = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.nombre

# =====
# MODELO: VIDEOJUEGO
# =====
class Videojuego(models.Model):
    titulo = models.CharField(max_length=150)
    genero = models.CharField(max_length=50)
    plataforma = models.CharField(max_length=50)
    precio = models.DecimalField(max_digits=10, decimal_places=2)
    fecha_lanzamiento = models.DateField()
    desarrollador = models.CharField(max_length=100)
    proveedor = models.ForeignKey(Proveedor,
on_delete=models.CASCADE, related_name="videojuegos")

    def __str__(self):
        return self.titulo

# =====
# MODELO: VENTA
# =====
class Venta(models.Model):
    videojuego = models.ForeignKey(Videojuego,
on_delete=models.CASCADE, related_name="ventas")
    total_venta = models.DecimalField(max_digits=10,
decimal_places=2)
    fecha_venta = models.DateTimeField(auto_now_add=True)
    cliente_nombre = models.CharField(max_length=100)
    metodo_pago = models.CharField(max_length=50)
    descuento = models.DecimalField(max_digits=5, decimal_places=2,
default=0)
    observaciones = models.TextField(blank=True, null=True)

```

```
def __str__(self):
    return f"Venta #{self.id} - {self.videojuego.titulo}"
```

=====

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

13 Primero trabajamos con el MODELO: PROVEEDOR

14 En view de app_Xbox crear las funciones con sus códigos correspondientes
(inicio_xbox, agregar_proveedor, actualizar_proveedor, realizar_actualizacion_proveedor, borrar_proveedor)

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

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 Xbox”, “Inicio”, “Proveedor”, en submenu de proveedor (Agregar Proveedor, Ver Proveedores, Actualizar Proveedor, Borrar Proveedor), “Videojuego” en submenu de videojuego (Agregar Videojuego, Ver Videojuegos, Actualizar Videojuego, Borrar Videojuego), “Ventas” en submenu de ventas (Agregar Venta, Ver Ventas, Actualizar Venta, Borrar Venta), incluir íconos a las opciones principales, no en los submenu.)

19 En el archivo footer.html incluir derechos de autor, fecha del sistema y “Creado por Ing. Eliseo Nava, 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 videojuegos de Xbox.

21 Crear la subcarpeta proveedor dentro de
app_Xbox\templates.

22 Crear los archivos html con su código correspondiente de

(agregar_proveedor.html, ver_proveedores.html mostrar en tabla con los botones ver, editar y borrar, actualizar_proveedor.html, borrar_proveedor.html)
dentro de app_Xbox\templates\proveedor.

23 No utilizar forms.py.

24 Procedimiento para crear el archivo urls.py en app_Xbox con el código correspondiente para acceder a las funciones de views.py para operaciones de CRUD en proveedor.

25 Procedimiento para agregar app_Xbox en settings.py de backend_Xbox.

26 Realizar las configuraciones correspondientes a urls.py de backend_Xbox para enlazar con app_Xbox.

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

27 Por lo pronto solo trabajar con “PROVEEDOR”, dejar pendiente MODELO: VIDEOJUEGO 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 8036.