
Primera parte

Proyecto: Netflix.

Lenguaje: Python.

Framework: Django.

Editor: VS code.

1 Procedimiento para crear carpeta del Proyecto: UIII_Netflix_0624

2 procedimiento para abrir vs code sobre la carpeta

UIII_Netflix_0624

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_Netflix sin duplicar carpeta.

9 procedimiento para ejecutar servidor en el puerto 8024

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

11 procedimiento para crear aplicación app_Netflix

12 Aquí el modelo models.py

=====

from django.db import models

MODELO: USUARIO

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

```
    id_usuario = models.AutoField(primary_key=True)
```

```
    nombre = models.CharField(max_length=255)
```

```
    apellido = models.CharField(max_length=255)
```

```
    email = models.CharField(max_length=255, unique=True)
```

```
fecha_registro = models.DateField(auto_now_add=True)
tipo_membresia = models.CharField(max_length=100)
ultimo_acceso = models.DateTimeField(auto_now=True)
pais = models.CharField(max_length=100)
```

```
code Code
downloadcontent_copy
expand_less
def __str__(self):
    return f'{self.nombre} {self.apellido} ({self.email})'
```



```
=====
```



```
=====
```

MODELO: PELICULA


```
=====
```



```
=====
```

```
class Pelicula(models.Model):
    id_pelicula = models.AutoField(primary_key=True)
    titulo = models.CharField(max_length=255)
    ano_lanzamiento = models.IntegerField()
    genero = models.CharField(max_length=100)
    clasificacion_edad = models.CharField(max_length=50)
    duracion_minutos = models.IntegerField()
    director = models.CharField(max_length=255)
    idioma_original = models.CharField(max_length=50)
```

```
code Code
downloadcontent_copy
expand_less
def __str__(self):
    return f'{self.titulo} ({self.ano_lanzamiento})'
```

=====

=====

=====

MODELO: LISTA_FAVORITOS

=====

=====

```
class ListaFavoritos(models.Model):
    categoria_lista = models.IntegerField()
    usuario = models.ForeignKey(Usuario, on_delete=models.CASCADE,
                                related_name='listas_favoritos')
    pelicula = models.ForeignKey(Pelicula, on_delete=models.CASCADE,
                                related_name='en_listas_favoritos')
    fecha_adicion = models.DateField(auto_now_add=True)
    orden_en_lista = models.IntegerField()
    visto_en_lista = models.BooleanField(default=False)
    notas_personales = models.TextField(blank=True, null=True)

    class Meta:
        unique_together = ('usuario', 'pelicula')

    def __str__(self):
        return f"Lista de {self.usuario.nombre} - {self.pelicula.titulo}"
```

=====

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

13 primero trabajamos con el MODELO: USUARIO

14 En view de app_Netflix crear las funciones con sus códigos correspondientes (inicio.netflix, agregar_usuario, actualizar_usuario, realizar_actualizacion_usuario, borrar_usuario)

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

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 Netflix”, “Inicio”, “Usuarios”,en submenu de Usuarios(Agregar Usuario,Ver Usuarios, Actualizar Usuario, Borrar Usuario), “Películas” en submenu de Películas(Agregar Película,Ver Películas, Actualizar Película, Borrar Película), “Listas de Favoritos” en submenu de Listas de Favoritos(Agregar a Lista,Ver Listas, actualizar Lista, Borrar de Lista), 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 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 Netflix.
- 21 Crear la subcarpeta carpeta usuario dentro de app_Netflix\templates.
- 22 crear los archivos html con su codigo correspondientes de (agregar_usuario.html, ver_usuarios.html mostrar en tabla con los botones ver, editar y borrar, actualizar_usuario.html, borrar_usuario.html) dentro de app_Netflix\templates\usuario.
- 23 No utilizar forms.py.
- 24 procedimiento para crear el archivo urls.py en app_Netflix con el código correspondiente para acceder a las funciones de views.py para operaciones de crud en usuarios.
- 25 procedimiento para agregar app_Netflix en settings.py de backend_Netflix
- 26 realizar las configuraciones correspondiente a urls.py de backend_Netflix para enlazar con app_Netflix
- 27 procedimiento para registrar los modelos en admin.py y volver a realizar las migraciones.
- 27 por lo pronto solo trabajar con “usuario” dejar pendiente # MODELO: PELÍCULA y # MODELO: LISTA_FAVORITOS
- 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 8024.