



Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ingeniería en Sistemas

Manual
"Técnico"

Gerardo Leonel Ortiz Tobar – 202200192

Guatemala, 03 de marzo, 2024

Menú:

Se implemento un método con un WHILE para que el usuario pueda usar nuevamente el menú para poder usar todas sus opciones, cuando el usuario no quisiera usar el programa puede escoger la opción de salir y si no escoge alguna de las opciones en el menú le manda un mensaje.

Este es un ejemplo:

```
def mostrar_informacion_desarrollador():
    print("="*50)
    print("Bienvenido al Sistema      ")
    print("Desarrollado por:           ")
    print("Lenguajes Formales y de Programación")
    print("Seccion: B+")
    print("Carné: 202200196")
    print("Nombre: Gerardo Leonel Ortiz Tobar")
    print("="*50)
    input("Presiona cualquier tecla para continuar...")
    mostrar_menu_principal()
```

```
def mostrar_menu_principal():
    global archivo, contenido
    print("=====")
    print("      Menú Principal      ")
    print("=====")
    print("1. Cargar archivo de entrada")
    print("2. Gestionar películas")
    print("3. Filtrado")
    print("4. Gráficas")
    print("5. Salir")
    print("")
    opcion = input("Selecciona una opción: ")
    if opcion == '1':
        limpiar_variables()
        archivo = input("Ingrese el nombre del archivo de entrada: ")
        contenido = cargar_archivo_entrada(archivo + ".lfp")
        agregarPelículas()
        t.sleep(0.1)

    elif opcion == '2':
        print(archivo)
        if validar_archivo(archivo):
            opcion_gestion_peliculas()
        else:
            cambioColor(2,"No se ha cargado un archivo de entrada")

    elif opcion == '3':
        if validar_archivo(archivo):
            opcion_filtrado()
        else:
            cambioColor(2,"No se ha cargado un archivo de entrada")

    elif opcion == '4':
        if validar_archivo(archivo):
            generar_grafico_peliculas()
```

```
    else:
        cambioColor(2,"No se ha cargado un archivo de entrada")

    elif opcion == '5':
        cambioColor(1,"Gracias por utilizar el sistema :)")
        exit()
    else:
        cambioColor(2,"Opción inválida")
    mostrar_menu_principal()
```

Leer Archivos:

Se creo la función de cargar_archivo_entrada, al momento de colocar la ruta, se va leyendo fila por fila y las retorna para su uso.

Ejemplo:

```
def cargar_archivo_entrada(nombre_archivo):
    try:
        with open(nombre_archivo, 'r') as archivo:
            contenido = archivo.readlines()

            return contenido

    except FileNotFoundError:
        cambioColor(2,f"El archivo '{nombre_archivo}' no se encontró.")
        return None
    except Exception as e:
        cambioColor(2,f"Ocurrió un error al leer el archivo '{nombre_archivo}': {e}")
        return None
```

Agregar Películas:

Esta función lo que realiza es al momento que se utiliza el método de leer archivos y retorna las filas, se hace un ciclo for para leer cada una, se separa y se almacena en un arreglo y se hace una condicional para poder realizar el método correspondiente y se hace la validación de cada uno de los datos guardados y si todo está correcto, se almacena

Ejemplo:

```
def agregarPeliculas():
    global contenido
    if contenido == None:
        cambioColor(2,"No se ha cargado un archivo de entrada.")
        return
    else:
        for i in contenido:
            cadena = i.split(';')
            nombre = cadena[0]
            actores = cadena[1].split(',')
            year = cadena[2].strip()
            genero = cadena[3].strip()
            pelicula = peli.Pelicula(nombre, year, genero)
            eliminar_pelicula(nombre)
            lista_peliculas.append(pelicula)
            for j in actores:
                actor = j.strip()
                pelicula.agregar_actor(actor)
                if actor not in lista_actores:
                    lista_actores.append(actor)
            cambioColor(1,"Peliculas agregadas correctamente")
```

Opciones de Filtrado:

Se implementó un método similar al del menú principal, solo que en este caso es para el uso del filtrado, cada opción tiene una función que retorna los datos solicitados. Ejemplo:

```
def opcion_filtrado():
    while True:
        print("=====")
        print("      Menu de Filtrado      ")
        print("=====")
        print("1. Filtrar por actor")
        print("2. Filtrar por año")
        print("3. Filtrar por género")
        print("4. Regresar al menú principal")

        opcion = input("Selecciona una opción: ")

        if opcion == '1':
            mostrarActores_filtro()

        elif opcion == '2':
            opcion_year = input("Ingrese el año: ")
            opcion_filtrado_por_año(opcion_year)
            t.sleep(2)

        elif opcion == '3':
            opcion_genero = input("Ingrese el género: ")
            opcion_filtrado_por_genero(opcion_genero)
            t.sleep(2)

        elif opcion == '4':
            return
```

Generar Grafica:

Lo que realiza este método es que genera una grafica de todas las películas enlazadas con sus actores, en dado caso de que el mismo actor este en dos películas se enlaza la película con el actor.

Ejemplo:

```
def generar_grafico_peliculas():
    global lista_peliculas, lista_actores
    grafo = gv.Digraph(format='png')

    for pelicula in lista_peliculas:
        etiqueta = f'<<table border="0" cellpadding="1" cellspacing="0">' \
            f'<tr><td colspan="1" bgcolor="lightgrey"><b>{pelicula.get_nombre()}</b></td></tr>' \
            f'<tr><td>{pelicula.get_year()}</td></tr>' \
            f'<tr><td>{pelicula.get_genero()}</td></tr>' \
            f'</table>>'
        grafo.node(pelicula.get_nombre(), shape='plaintext', label=etiqueta)

    for actor in lista_actores:
        grafo.node(actor, shape='circle', color='black')

    for pelicula in lista_peliculas:
        actores_pelicula = pelicula.get_actores()
        for actor in actores_pelicula:
            grafo.edge(actor, pelicula.get_nombre())

    grafo.render('peliculas', view=True)
    print("Gráfico de películas generado con éxito.")
```