



Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Computación

Profesor: Jaime Alejandro Romero Sierra

Programacion Avanzada

# Practica #1

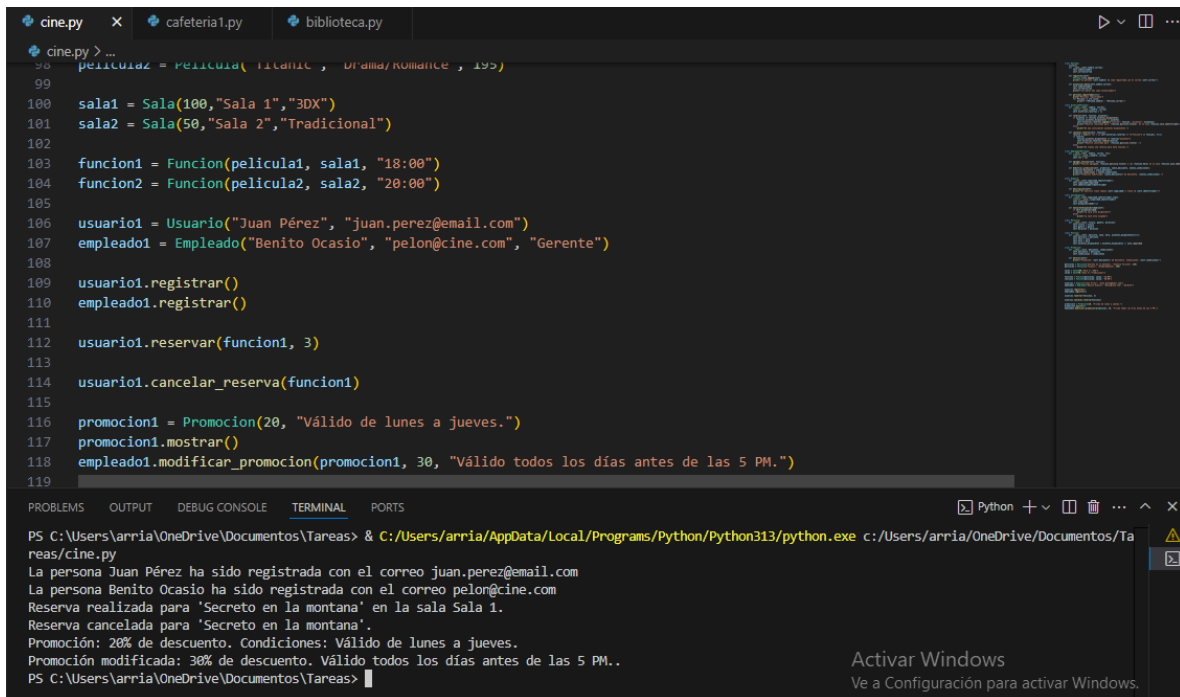
Nombre: Axel Aldair Arriaga Blanco

Matricula: 202440060

## Código 1: Cine

Se necesita crear un programa para hacer reservas en un cine, aplicar promociones, los tipos de salas, horarios, asientos y gestionar los productos.

Clases	Subclases	Atributos	Métodos
Persona	Usuario	<ul style="list-style-type: none"><li>• Nombre</li><li>• Correo</li></ul>	<ul style="list-style-type: none"><li>• Reservar</li><li>• Cancelar reserva</li></ul>
	Empleado	<ul style="list-style-type: none"><li>• Nombre</li><li>• Correo</li><li>• Rol</li></ul>	<ul style="list-style-type: none"><li>• Agregar función</li><li>• Modificar promoción</li></ul>
Espacio	Sala	<ul style="list-style-type: none"><li>• Capacidad</li><li>• Identificador</li><li>• Tipo</li></ul>	<ul style="list-style-type: none"><li>• Descripción (Espacio)</li><li>• Consultar disponibilidad</li></ul>
Película		<ul style="list-style-type: none"><li>• Título</li><li>• Género</li><li>• Duración</li></ul>	
Función		<ul style="list-style-type: none"><li>• Película</li><li>• Sala</li><li>• Hora</li><li>• Asientos disponibles</li></ul>	
Promoción		<ul style="list-style-type: none"><li>• Descuento</li><li>• Condiciones</li></ul>	<ul style="list-style-type: none"><li>• Mostrar el descuento y las condiciones</li></ul>



```

cine.py x cafeteria1.py biblioteca.py
cine.py > ...
98 pelicula2 = Pelicula( 'Titanic', 'Drama/Romance', 195)
99
100 sala1 = Sala(100,"Sala 1","3DX")
101 sala2 = Sala(50,"Sala 2","Tradicional")
102
103 funcion1 = Funcion(pelicula1, sala1, "18:00")
104 funcion2 = Funcion(pelicula2, sala2, "20:00")
105
106 usuario1 = Usuario("Juan Pérez", "juan.perez@email.com")
107 empleado1 = Empleado("Benito Ocasio", "pelon@cene.com", "Gerente")
108
109 usuario1.registrar()
110 empleado1.registrar()
111
112 usuario1.reservar(funcion1, 3)
113
114 usuario1.cancelar_reserva(funcion1)
115
116 promocion1 = Promocion(20, "Válido de lunes a jueves.")
117 promocion1.mostrar()
118 empleado1.modificar_promocion(promocion1, 30, "Válido todos los días antes de las 5 PM.")
119

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Python + v ...

PS C:\Users\arria\OneDrive\Documentos\Tareas> & C:/Users/arria/AppData/Local/Programs/Python/Python313/python.exe c:/Users/arria/OneDrive/Documentos/Tareas/cine.py

La persona Juan Pérez ha sido registrada con el correo juan.perez@email.com

La persona Benito Ocasio ha sido registrada con el correo pelon@cene.com

Reserva realizada para 'Secreto en la montana' en la sala Sala 1.

Reserva cancelada para 'Secreto en la montana'.

Promoción: 20% de descuento. Condiciones: Válido de lunes a jueves.

Promoción modificada: 30% de descuento. Válido todos los días antes de las 5 PM..

PS C:\Users\arria\OneDrive\Documentos\Tareas>

Activar Windows  
Ve a Configuración para activar Windows.

[https://github.com/axelab08/Practica\\_1/blob/main/cine.py](https://github.com/axelab08/Practica_1/blob/main/cine.py)

## Código 2: Cafetería

Se necesita un programa para gestionar una cafetería, se añadirán pedidos personalizados, se analizará el inventario verificando la cantidad y agregando ingredientes y se incluirán los roles de empleado.

Clases	Subclases	Atributos	Métodos
<b>Persona</b>		<ul style="list-style-type: none"> <li>Nombre</li> <li>Correo</li> </ul>	
	Cliente	<ul style="list-style-type: none"> <li>Nombre</li> <li>Correo</li> </ul>	
	Empleado	<ul style="list-style-type: none"> <li>Nombre</li> <li>Correo</li> <li>Rol</li> </ul>	<ul style="list-style-type: none"> <li>Reabastecer ingredientes</li> </ul>
<b>ProductoBase</b>		<ul style="list-style-type: none"> <li>Nombre</li> <li>Precio</li> </ul>	
	Bebida	<ul style="list-style-type: none"> <li>Tipo</li> <li>Tamaño</li> <li>Opciones</li> </ul>	<ul style="list-style-type: none"> <li>Personalizar</li> </ul>
	Postre	<ul style="list-style-type: none"> <li>Vegano</li> <li>Sin gluten</li> </ul>	
<b>Inventario</b>		<ul style="list-style-type: none"> <li>Ingredientes</li> </ul>	<ul style="list-style-type: none"> <li>Verificar disponibilidad</li> <li>Actualizar inventario</li> <li>Reabastecer</li> </ul>
<b>Promoción</b>		<ul style="list-style-type: none"> <li>Descripción</li> <li>Descuento</li> <li>Mínimo de productos</li> <li>Cliente frecuente</li> </ul>	<ul style="list-style-type: none"> <li>Validar si consume una cantidad de productos o es cliente frecuente</li> </ul>
<b>Pedido</b>		<ul style="list-style-type: none"> <li>Cliente</li> <li>Inventario</li> <li>Productos</li> <li>Estado del pedido</li> <li>Total</li> </ul>	<ul style="list-style-type: none"> <li>Agregar productos</li> <li>Calcular total</li> <li>Cambiar estado</li> <li>Aplicar la promocion</li> </ul>

```
cine.py  cafeteria1.py X  biblioteca.py
cafeteria1.py > ...
120 cafe = Pedido(cafe, 5, mediano, cliente,
121 cafe.personalizar("leche", "almendra")
122 cafe.personalizar("azúcar", "sin azúcar")
123
124 postre = Postre("Brownie", 8, sin_gluten=True)
125
126 pedido1 = Pedido(cliente1, inventario)
127
128 pedido1.agregar_producto(cafe, {"café": 2, "leche": 1})
129 pedido1.agregar_producto(postre, {"harina": 2, "chocolate": 1})
130
131 print(f"Total antes de promoción: ${pedido1.total}")
132
133 promo = Promocion("Descuento cliente frecuente", 3, min_productos=2, cliente_frecuente=True)
134 pedido1.aplicar_promocion(promo)
135
136 print(f"Total después de promoción: ${pedido1.total}")
137
138 pedido1.cambiar_estado("en preparación")
139 print(f"Estado del pedido: {pedido1.estado}")
140
141 empleado1.reabastecer_ingredientes("café", 10)

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\arria\OneDrive\Documentos\Tareas> & C:/Users/arria/AppData/Local/Programs/Python/Python313/python.exe
reas/cafeteria1.py
Café agregado al pedido.
Brownie agregado al pedido.
Total antes de promoción: $13
Total después de promoción: $10
Estado del pedido: en preparación
Ana ha reabastecido 10 de café.
PS C:\Users\arria\OneDrive\Documentos\Tareas> |
```

[https://github.com/axelab08/Practica\\_1/blob/main/cafeteria.py](https://github.com/axelab08/Practica_1/blob/main/cafeteria.py)

### Código 3: Biblioteca digital

El propósito de este código es crear un programa que sea capaz de gestionar el préstamo de libros, incluyendo libros y revistas. Utilizando a bibliotecario para agregar material nuevo y gestionando los prestamos y al usuario que puede consultar el material disponible y sus libros prestados. Este programa incluirá una penalización por devolución tardía y la capacidad de devolver en varias sucursales.

Clases	Subclases	Atributos	Métodos
<b>Material</b>		<ul style="list-style-type: none"> <li>• Título</li> <li>• Estado</li> </ul>	
	Libro	<ul style="list-style-type: none"> <li>• Autor</li> <li>• Genero</li> </ul>	
	Revista	<ul style="list-style-type: none"> <li>• Edición</li> <li>• Periodicidad</li> </ul>	
	Material Digital	<ul style="list-style-type: none"> <li>• Tipo de archivo</li> <li>• Enlace</li> </ul>	
<b>Persona</b>		<ul style="list-style-type: none"> <li>• Nombre</li> </ul>	
	Usuario	<ul style="list-style-type: none"> <li>• Prestamos</li> <li>• Multas</li> </ul>	<ul style="list-style-type: none"> <li>• Consulta el catalogo</li> <li>• Devolver libros prestados</li> </ul>
	Bibliotecario	<ul style="list-style-type: none"> <li>• Sucursal donde se encuentra</li> <li>• Material</li> </ul>	<ul style="list-style-type: none"> <li>• Agregar material</li> <li>• Gestionar prestamos</li> <li>• Transferir material</li> </ul>
<b>Sucursal</b>		<ul style="list-style-type: none"> <li>• Nombre</li> <li>• Catalogo</li> </ul>	<ul style="list-style-type: none"> <li>• Listar materiales</li> </ul>
<b>Préstamo</b>		<ul style="list-style-type: none"> <li>• Usuario</li> <li>• Material</li> <li>• Fecha de préstamo</li> <li>• Fecha de devolución</li> </ul>	
<b>Catalogo</b>		<ul style="list-style-type: none"> <li>• Sucursales</li> </ul>	<ul style="list-style-type: none"> <li>• Listar materiales</li> </ul>

```
cine.py  cafeteria1.py  biblioteca.py X
biblioteca.py > ...
38     return materiales
39
40
41 sucursal_central = Sucursal("Biblioteca Central")
42 sucursal_norte = Sucursal("Biblioteca Norte")
43
44 catalogo = Catalogo([sucursal_central, sucursal_norte])
45 bibliotecario = Bibliotecario("Ana Gómez")
46 libro1 = Libro("Cien años de soledad", "Gabriel García Márquez", "Novela")
47 revista1 = Revista("National Geographic", "Enero 2024", "Mensual")
48 bibliotecario.agregar_material(sucursal_central, libro1)
49 bibliotecario.agregar_material(sucursal_norte, revista1)
50
51 usuario = Usuario("Carlos Ramírez")
52 print("Materiales en todas las sucursales:", usuario.consultar_catalogo(catalogo))
53 bibliotecario.gestionar_prestamo(usuario, libro1)
54
55 bibliotecario.gestionar_prestamo(usuario, libro1)
56 usuario.devolver_material(usuario.prestamos[0])
57 print(f"Multa acumulada por {usuario.nombre}: {usuario.multas} créditos")
58 bibliotecario.transferir_material(revista1, sucursal_norte, sucursal_central)
59 print(f"Materiales en todas las sucursales después de la transferencia: {catalogo.lista_materiales}")
60

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
S C:\Users\arria\OneDrive\Documentos\Tareas> & C:/Users/arria/AppData/Local/Programs/Python/Python313/python.exe
eas/biblioteca.py
Materiales en todas las sucursales: ['Cien años de soledad', 'National Geographic']
Cien años de soledad ha sido prestado a Carlos Ramírez
Material no disponible
Carlos Ramírez ha devuelto 'Cien años de soledad'
Multa acumulada por Carlos Ramírez: 0 créditos
National Geographic transferido a Biblioteca Central
Materiales en todas las sucursales después de la transferencia: ['Cien años de soledad', 'National Geographic']
S C:\Users\arria\OneDrive\Documentos\Tareas>

Ln 104, Col 1
```

[https://github.com/axelab08/Practica\\_1/blob/main/biblioteca.py](https://github.com/axelab08/Practica_1/blob/main/biblioteca.py)