



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

Periodo Primavera 2025



PROGRAMACIÓN AVANZADA

Profesor: Jaime Alejandro Romero Sierra

PRÁCTICA UNIDAD 1

Diagramas de Clase UML

NOMBRE:

Valeria Pérez Chávez

MATRICULA:

202452031

11 de febrero

JUSTIFICACIÓN

En la actualidad, la optimización de los procesos administrativos y operativos en los negocios es un factor clave para mejorar la eficiencia, reducir costos y ofrecer un mejor servicio a los clientes. La gestión manual de registros, inventarios y transacciones puede resultar tediosa, propensa a errores y demandante en términos de tiempo y recursos humanos. Ante esta problemática, la implementación de soluciones tecnológicas automatizadas se ha convertido en una necesidad para cualquier empresa que busque mejorar su competitividad en el mercado.

El presente proyecto consiste en el desarrollo de tres sistemas de gestión diseñados para distintos tipos de establecimientos: un cine, una cafetería y una biblioteca. Cada uno de estos programas ha sido desarrollado en **Python**, utilizando principios de programación estructurada y orientada a objetos para garantizar eficiencia, escalabilidad y facilidad de mantenimiento.

1. Sistema de gestión para un cine:

Este software permite la administración integral de un cine, abarcando funciones esenciales como la gestión de clientes, personal, películas, asignación de espacios y control de promociones. La automatización de estos procesos no solo facilita el manejo interno del negocio, sino que también mejora la experiencia del cliente al ofrecer un servicio más ágil y organizado.

2. Sistema de gestión para una cafetería:

La administración de cafeterías requiere un control preciso del

inventario, ventas, pedidos y personal. Este programa permite gestionar el stock de productos, promoción por fidelidad y optimizar la distribución de recursos dentro del establecimiento. Además, la implementación de este software reduce la posibilidad de errores humanos en el cálculo de ingresos y control de insumos, lo que contribuye a una mayor rentabilidad y sostenibilidad del negocio.

3. Sistema de gestión para una biblioteca:

El correcto funcionamiento de una biblioteca depende en gran medida de un sistema de control eficiente para la administración de préstamos, devoluciones y usuarios registrados. Este programa automatiza dichos procesos, permitiendo un seguimiento preciso del material bibliográfico, evitando pérdidas y asegurando que los usuarios tengan acceso oportuno a los recursos disponibles. Además, facilita la generación de reportes y el control de sanciones en caso de retrasos en la devolución de libros.

CLASES A OCUPAR

1. Sistema de Reservas para un Cine

1. Clase Persona:

- Atributos:

- name: Almacena el nombre de la persona
- age: Almacena la edad
- mail: Almacena el correo electrónico
- list: Una lista vacía

- Métodos:

- saludo(): Imprime un mensaje con el nombre y edad de la persona
- actualizar_datos(): Permite actualizar nombre, email y edad

2. Clase Usuario (hereda de Persona):

- Atributos:

- Hereda todos los atributos de Persona
- reservas: Lista para almacenar las reservas del usuario

- Métodos:

- promos(): Muestra las promociones disponibles con sus descuentos y condiciones
- hacer_reserva(): Agrega una nueva reserva con función y asientos

- cancelar_reserva(): Elimina una reserva existente y libera los asientos ocupados

3. Clase Funcion:

- Atributos:

- movie: Almacena la película
- room: Almacena la sala
- horario: Almacena la hora de la función
- all_functions: Lista estática que guarda todas las funciones creadas

- Métodos:

- __str__(): Retorna una representación en texto de la función con película, sala y horario.

4. Clase Empleado (hereda de Persona):

- Atributos:

- rol: Almacena el rol del empleado
- functions: Lista de funciones
- movies: Lista de películas
- promotions: Lista de promociones

- Métodos:

- new_function(): Agrega una nueva función si el rol es "Limpieza"

- new_movie(): Agrega una nueva película si el rol es "Limpieza"
- new_promotion(): Agrega una nueva promoción si el rol es "Limpieza"

5. Clase Espacio:

- Atributos:
 - size: Almacena el tamaño del espacio

6. Clase Sala (hereda de Espacio):

- Atributos:
 - type: Tipo de sala
 - capacity: Capacidad máxima
 - asientos_ocupados: Lista de asientos ocupados
- Métodos:
 - availability(): Verifica disponibilidad de asientos
 - occupy(): Ocupa asientos si hay capacidad disponible

7. Clase ZonaComida (hereda de Espacio):

- Atributos:
 - products: Diccionario de productos y precios
- Métodos:

- add_product(): Agrega un producto con su precio

8. Clase Película:

- Atributos:

- title: Título de la película
- duration: Duración
- classif: Clasificación
- gender: Género

- Métodos:

- __str__(): Retorna información formateada de la película

9. Clase Promocion:

- Atributos:

- description: Descripción de la promoción
- discount: Porcentaje de descuento
- condition: Condiciones de la promoción

- Métodos:

- __str__(): Retorna información formateada de la promoción

10. Clase Reserva:

- Atributos:

- user: Usuario que hace la reserva
- seats: Asientos reservados
- function: Función reservada
- total: Precio total calculado

- Métodos:

- calcular_total(): Calcula el precio total según el tipo de sala (2D, 3D, IMAX)
- confirm_reserva(): Confirma la reserva verificando disponibilidad

2. Gestión de Pedidos en una Cafetería

1. Clase Persona:

- Atributos:

- name: Nombre de la persona
- age: Edad
- mail: Correo electrónico
- list: Lista vacía
- roles: Lista de roles asignados

- Métodos:

- actualizar_datos(): Actualiza nombre, edad y email

2. Clase Cliente (hereda de Persona):

- Atributos:

- orders: Lista de pedidos
- puntos_fidelidad: Contador de puntos

- Métodos:

- hacer_pedido(): Verifica stock, consume ingredientes y agrega el pedido
- consultar_historial(): Retorna lista de pedidos
- aplicar_descuento_fidelidad(): Aplica 10% de descuento si tiene 2+ puntos

3. Clase Empleado (hereda de Persona):

- Atributos:

- rol: Rol específico del empleado

4. Clase ProductoBase:

- Atributos:

- name: Nombre del producto
- price: Precio
- receta: Diccionario de ingredientes

- Métodos:

- set_receta(): Establece los ingredientes necesarios

5. Clase Bebida (hereda de ProductoBase):

- Atributos:

- size: Tamaño de la bebida

- type: Tipo de bebida

- custom: Lista de personalizaciones

- Métodos:

- agregar_B(): Agrega una bebida

6. Clase Postre (hereda de ProductoBase):

- Atributos:

- vegano: Indica si es vegano

- sin_gluten: Indica si es sin gluten

- Métodos:

- agregar_P(): Agrega un postre

7. Clase Inventario:

- Atributos:

- stock: Diccionario de productos y cantidades
- Métodos:
 - actualizar_inventario(): Actualiza cantidad de un producto existente
 - add_product(): Agrega nuevo producto al inventario
 - consumir(): Reduce el stock al usar ingredientes

8. Clase Pedido:

- Atributos:
 - client: Cliente que realiza el pedido
 - product: Lista de productos del pedido
 - status: Estado del pedido (inicia como "Pendiente")
 - total: Monto total del pedido (inicia en 0)
- Métodos:
 - calcular_total(): Calcula la suma total de los precios de los productos
 - new_status(): Cambia el estado del pedido si el empleado tiene un rol autorizado (Mesero, Barista o Gerente)
 - __repr__(): Genera una representación en texto del pedido incluyendo:
 - Nombre del cliente
 - Lista de productos
 - Total calculado

- Estado actual del pedido

9. Clase Promocion:

- Atributos:

- discount: Porcentaje de descuento

- Métodos:

- apply_discount(): Aplica descuento específicamente a brownies

3. Biblioteca Digital

1. Clase Material:

- Atributos:

- titulo: Título del material
- editorial: Editorial que lo publicó
- anio: Año de publicación
- disponible: Estado de disponibilidad

- Métodos:

- `__str__()`: Muestra información formateada del material

2. Clase Libro (hereda de Material):

- Atributos adicionales:

- autor: Autor del libro
- genero: Género literario

3. Clase Revista (hereda de Material):

- Atributos adicionales:

- edicion: Número de edición
- period: Periodicidad

4. Clase MaterialDigital (hereda de Material):

- Atributos adicionales:
 - formato: Formato del material
 - url: URL de acceso

5. Clase Usuario:

- Atributos:
 - nombre: Nombre del usuario
 - edad: Edad
 - email: Correo electrónico
 - multas: Monto de multas acumuladas
- Métodos:
 - acumular_multa(): Suma una cantidad a las multas
 - mostrar_catalogo(): Muestra materiales disponibles por sucursal

6. Clase Bibliotecario (hereda de Usuario):

- Métodos:
 - prestar_material(): Gestiona el préstamo de materiales
 - recibir_material(): Registra la devolución de materiales
 - transferir_material(): Mueve materiales entre sucursales
 - agregar_material(): Añade material al catálogo

7. Clase Sucursal:

- Atributos:

- nombre: Nombre de la sucursal
- catalogo: Lista de materiales

- Métodos:

- buscar_material_disponible(): Busca material en la sucursal
- buscar_material_global(): Busca material en todas las sucursales

8. Clase Prestamo:

- Atributos:

- usuario: Usuario que realiza el préstamo
- material: Material prestado
- fecha_prestamo: Fecha de inicio
- fecha_devolucion: Fecha límite de devolución

9. Clase Penalizacion:

- Atributos:

- usuario: Usuario penalizado
- dias_retrazo: Días de retraso
- multa: Monto calculado (días * 5)

10. Clase Catalogo:

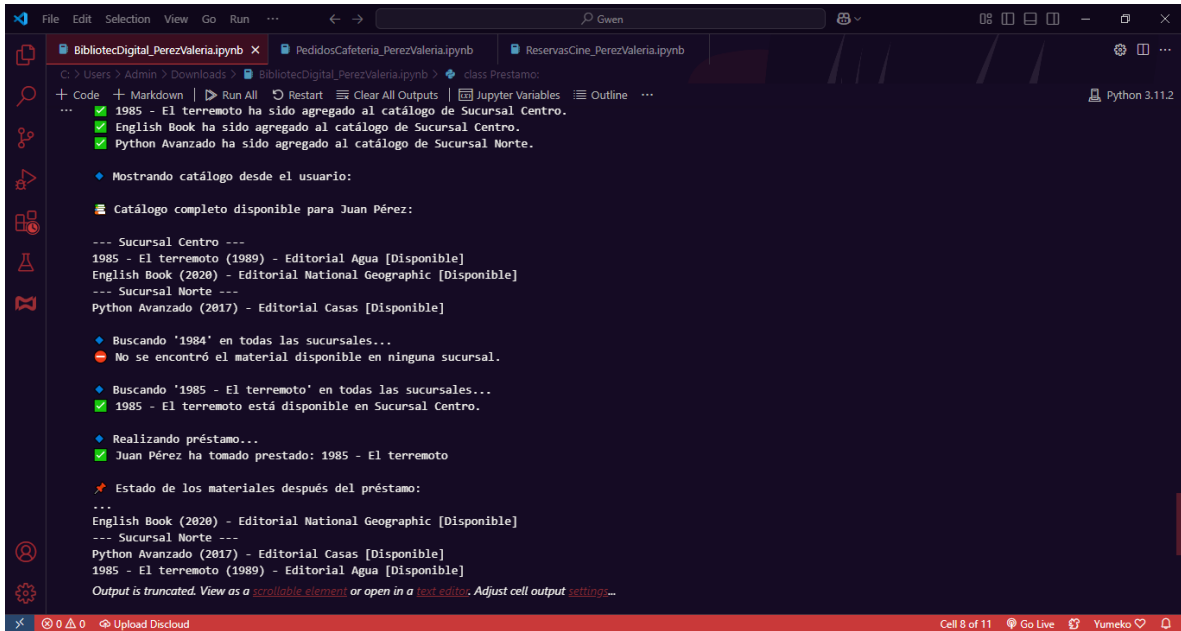
- Atributos:

- materiales: Lista de materiales

- Métodos:

- agregar_material(): Añade material al catálogo
- listar_materiales(): Retorna lista de títulos
- buscar_por_criterio(): Busca materiales según criterio específico

CAPTURAS DEL CODIGO



```
File Edit Selection View Go Run ... ← → Gwen
BibliotecaDigital_PerezValeria.ipynb PedidosCafeteria_PerezValeria.ipynb ReservasCine_PerezValeria.ipynb
C: > Users > Admin > Downloads > BibliotecaDigital_PerezValeria.ipynb > class Prestamo:
...
+ Code + Markdown | ▶ Run All | ⌂ Restart | ⌂ Clear All Outputs | ⌂ Jupyter Variables | ⌂ Outline ... Python 3.11.2

1985 - El terremoto ha sido agregado al catálogo de Sucursal Centro.
English Book ha sido agregado al catálogo de Sucursal Centro.
Python Avanzado ha sido agregado al catálogo de Sucursal Norte.

Mostrando catálogo desde el usuario:

Catálogo completo disponible para Juan Pérez:

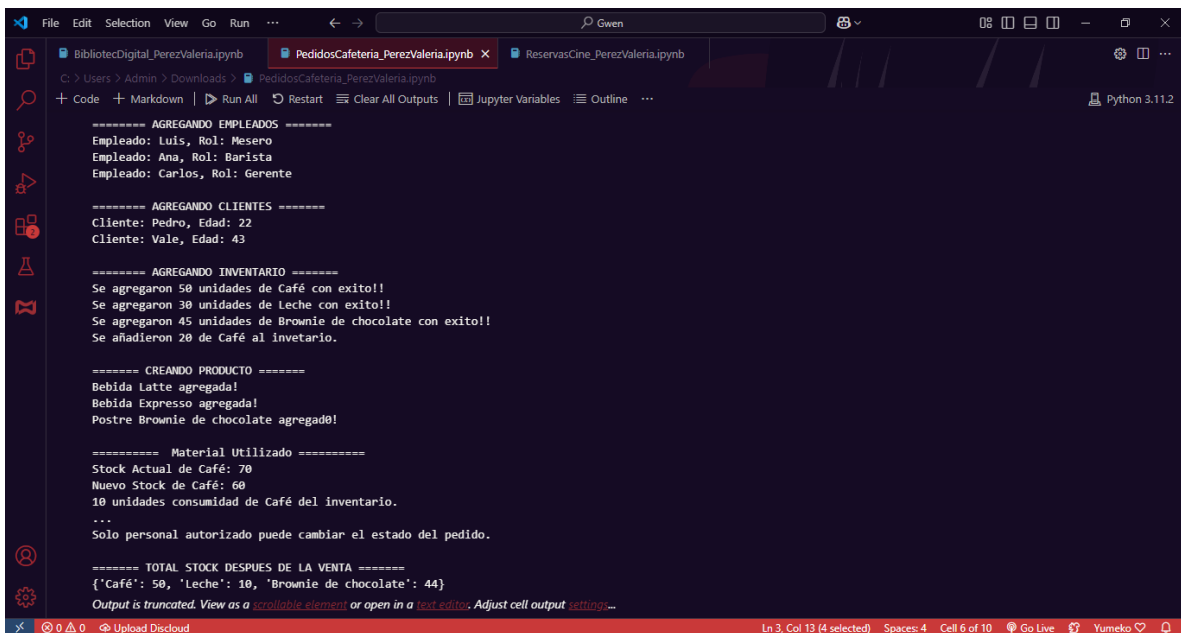
--- Sucursal Centro ---
1985 - El terremoto (1989) - Editorial Agua [Disponible]
English Book (2020) - Editorial National Geographic [Disponible]
--- Sucursal Norte ---
Python Avanzado (2017) - Editorial Casas [Disponible]

Buscando '1984' en todas las sucursales...
No se encontró el material disponible en ninguna sucursal.

Buscando '1985 - El terremoto' en todas las sucursales...
1985 - El terremoto está disponible en Sucursal Centro.

Realizando préstamo...
Juan Pérez ha tomado prestado: 1985 - El terremoto

Estado de los materiales después del préstamo:
...
English Book (2020) - Editorial National Geographic [Disponible]
--- Sucursal Norte ---
Python Avanzado (2017) - Editorial Casas [Disponible]
1985 - El terremoto (1989) - Editorial Agua [Disponible]
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
0 0 0 Upload Disccloud Cell 8 of 11 Go Live Yumeko
```



```
File Edit Selection View Go Run ... ← → Gwen
BibliotecaDigital_PerezValeria.ipynb PedidosCafeteria_PerezValeria.ipynb ReservasCine_PerezValeria.ipynb
C: > Users > Admin > Downloads > PedidosCafeteria_PerezValeria.ipynb
+ Code + Markdown | ▶ Run All | ⌂ Restart | ⌂ Clear All Outputs | ⌂ Jupyter Variables | ⌂ Outline ... Python 3.11.2

===== AGREGANDO EMPLEADOS =====
Empleado: Luis, Rol: Mesero
Empleado: Ana, Rol: Barista
Empleado: Carlos, Rol: Gerente

===== AGREGANDO CLIENTES =====
Cliente: Pedro, Edad: 22
Cliente: Vale, Edad: 43

===== AGREGANDO INVENTARIO =====
Se agregaron 50 unidades de Café con éxito!!
Se agregaron 30 unidades de Leche con éxito!!
Se agregaron 45 unidades de Brownie de chocolate con éxito!!
Se añadieron 20 de Café al inventario.

===== CREANDO PRODUCTO =====
Bebida Latte agregada!
Bebida Espresso agregada!
Postre Brownie de chocolate agregado!

===== Material Utilizado =====
Stock Actual de Café: 70
Nuevo Stock de Café: 60
10 unidades consumidad de Café del inventario.
...
Solo personal autorizado puede cambiar el estado del pedido.

===== TOTAL STOCK DESPUES DE LA VENTA =====
{'Café': 50, 'Leche': 10, 'Brownie de chocolate': 44}
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
0 0 0 Upload Disccloud Ln 3, Col 13 (4 selected) Spaces: 4 Cell 6 of 10 Go Live Yumeko
```

```
File Edit Selection View Go Run ... Gwen
BibliotecDigital_PerezValeria.ipynb PedidosCafeteria_PerezValeria.ipynb ReservasCine_PerezValeria.ipynb
C:\Users\Admin\Downloads> ReservasCine_PerezValeria.ipynb
+ Code + Markdown | Run All Restart Clear All Outputs | Jupyter Variables Outline ... Python 3.11.2

===== CREANDO PELICULAS =====
Interstellar: Duracion: 169, Clasificacion: PG-13, Genero: Ciencia Ficción
El Padrino: Duracion: 175, Clasificacion: R, Genero: Drama

===== CREANDO SALAS =====
Sala creada: 2D, Capacidad: 100
Sala creada: 3D, Capacidad: 80
Sala creada: IMAX, Capacidad: 20

===== CREANDO FUNCIONES =====
Película: Interstellar en sala 2D con horario de 18:00
Película: El Padrino en sala IMAX con horario de 20:30

===== CREANDO EMPLEADOS =====
Empleado: Carlos, Rol: Taquillero
Empleado: Luis, Rol: Limpiero

===== EMPLEADOS CREANDO FUNCIONES =====
Función 'Interstellar' agregada con éxito!!!
Función 'El Padrino' agregada con éxito!!!

===== CREANDO PROMOCIONES =====
Descuento de Martes, Descuento del 20%. Condiciones: Válido solo los martes
Promo de Cumpleaños, Descuento del 15%. Condiciones: Válido solo el día de tu cumpleaños
...
Asientos ocupados en El Padrino: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

===== PROBANDO ZONA DE COMIDA =====
Productos en zona de comida: {'Pizza 🍕': 50, 'Refresco 🥤': 30}
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
x 0 0 0 Upload Discloud Cell 7 of 11 Go Live Yumeko
```

LINK DE LOS CODIGOS

https://github.com/AravidEz/Portafolio/blob/main/ReservasCine_PerezValeria.ipynb

https://github.com/AravidEz/Portafolio/blob/main/PedidosCafeteria_PerezValeria.ipynb

https://github.com/AravidEz/Portafolio/blob/main/BibliotecDigital_PerezValeria.ipynb