**Práctica Calificada 1 - Programación Orientada a Objetos**

**Introducción**

En esta práctica evaluaremos el manejo de relaciones entre clases y la correcta aplicación de los principios de la Programación Orientada a Objetos (POO). Trabajaremos con registros de pedidos realizados por dos clientes, los cuales se encuentran en archivos separados. Se evaluará tanto la capacidad de lectura y modelado de datos como el uso de clases, atributos y métodos para representar una solución coherente y funcional.

**Objetivos**

* Aplicar los conceptos de clases, objetos, atributos y métodos.
* Utilizar relaciones entre clases como agregación y composición.
* Diseñar una solución modular basada en objetos.
* Leer y procesar archivos de entrada que contienen pedidos de clientes.

**Datos de Entrada**

**Se va a trabajar con dos archivos de pedidos para los clientes:**

* **Cliente 1:** Dalton Perez

Revisar el archivo adjunto.

* **Cliente 2:** Ryan Thompson

Revisar el archivo Adjunto.

**Requisitos del Sistema**

1. Crear las clases necesarias para representar:
   * Cliente
   * Producto
   * Pedido
2. Cada cliente tiene una clase pedido.
3. Cada pedido puede contener múltiples productos.
4. Debe implementarse una clase principal que gestione los pedidos cargados desde los archivos.

**Guía de Campos y Clases**

Los archivos de Dalton y Ryan contienen la misma estructura a continuación se le brinda una ayuda para que identifique los campos y cree las respectivas clases.

|  |  |
| --- | --- |
| Campos | Clases |
| SalesOrderID | Producto |
| SalesOrderDetailID | Producto |
| ProductID | Producto |
| NombreProducto | Producto |
| Cantidad | Producto |
| Categoria | Categoria |
| tipo | Producto |
| Precio | Producto |
| FechaVenta | Producto |
| CustomerID | Cliente |
| FirstName | Cliente |
| LastName | Cliente |
| Peso | Producto |
| link\_descarga | Producto |

**Interfaz de Usuario**

La interfaz de ingreso debe contemplar **ingresar los datos de los dos clientes** con su respectivo archivo de carga. En la primera parte se deben ingresar los datos del Cliente Dalton y en la segunda parte los datos del Cliente Ryan.

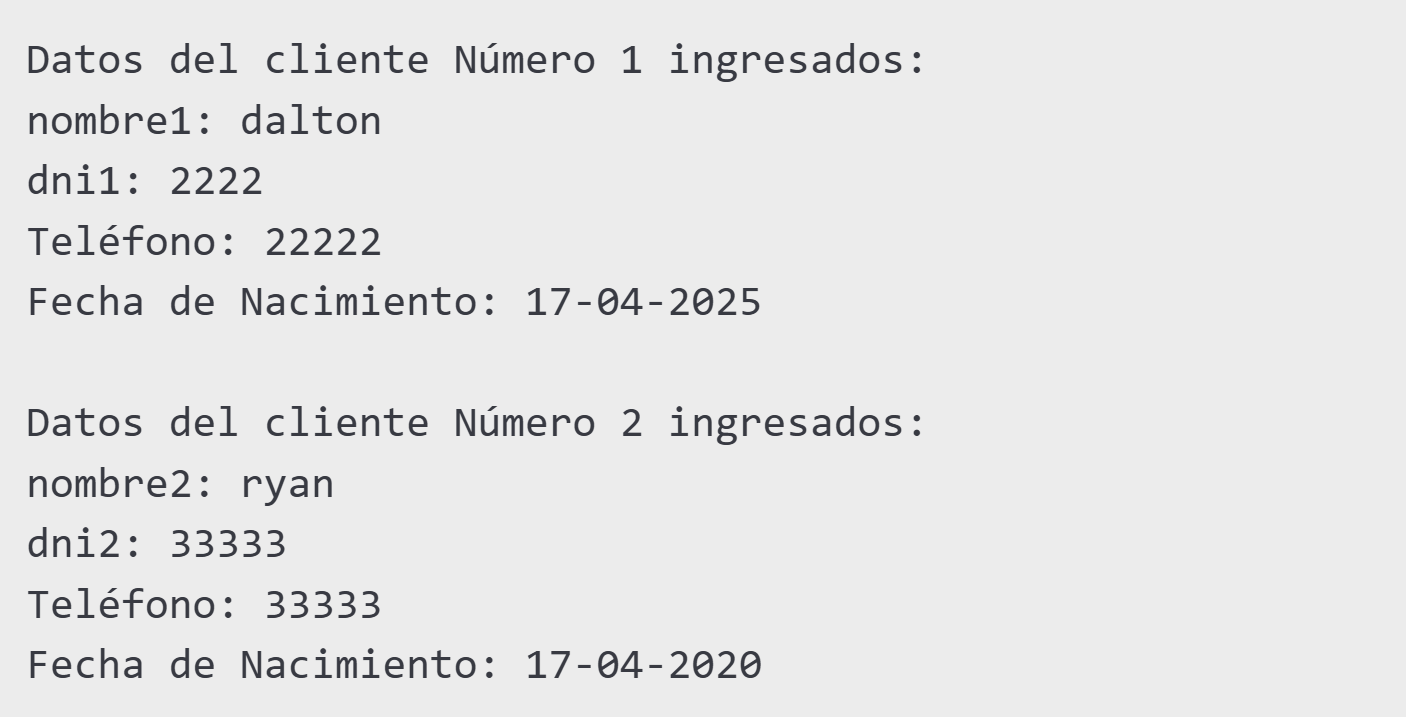
**Estructura de la Interfaz**

|  |  |
| --- | --- |
| Campo | Tipo de dato |
| Nombre completo 1 | **Texto** |
| DNI 1 | **Texto** |
| Teléfono 1 | **Texto** |
| Fecha de nacimiento 1 | **Fecha** |
| Ruta Excel 1 | **Texto** |
| Nombre completo 2 | **Texto** |
| DNI 2 | **Texto** |
| Teléfono 2 | **Texto** |
| Fecha de nacimiento 2 | **Fecha** |
| Ruta Excel 2 | **Texto** |

La interfaz debe tener un botón **"Guardar y continuar"** que valide los datos y permita avanzar al procesamiento de los pedidos.

**Actividades a Realizar**

1. Cargar los datos desde los archivos proporcionados , en conjunto con los datos que se indican en la interfaz.
2. Crear instancias de las clases según los datos cargados.
3. En la Salida se debe ver la siguiente estructura:



1. Mostrar un resumen de los pedidos realizados por cada cliente, incluyendo:

|  |  |
| --- | --- |
| Datos del cliente |  |
| Detalle de los productos |  |
| Resumen del total del pedido |  |

|  |  |
| --- | --- |
| Datos del cliente |  |
| Detalle de los productos |  |
| Resumen del total del pedido |  |

1. Finalmente, ambos objetos de cliente **deben agregarse a una lista**. Se debe implementar una función que determine qué cliente realizó la compra mayor, mostrando un resumen como este:

Texto

El contenido generado por IA puede ser incorrecto.

**Criterios de Evaluación**

La práctica será evaluada en base a los siguientes criterios:

* Correcta definición y relación de clases.
* Lectura y manipulación de archivos.
* Lógica del procesamiento de datos.
* Claridad y orden del código.
* Presentación de resultados.
* **Importante, etiquete, Describa donde esta aplicando Herencia, Encapsulamiento, Agregación etc.**

**Entrega**

Suba su carpeta Zipeada la cual debe tener su Codigo\_Nombre\_Apellido.zip.

Dentro del Archivo tienen que estar:

* **El archivo de Definición del Modelo (Modelo\_[Codigo\_Alumno].py)** agréguele su código en código\_alumno.
* **El archivo de ejecución (main\_[Codigo\_Alumno].py)agregue su código de alumno.**
* **Archivos Excel**
* **Opcional, si considera necesario alguna nota agréguelo en txt.**