



PROGRAMACION ORIENTADA A OBJETOS (CC201)

Ciclo 2024 - 02

Taller Semana 9

POO-Encapsulamiento-Agregacion

Todos los programas se deben resolver usando POO y atributos privados

Caso 1 : Encomienda

Diseñe la clase Encomienda con los atributos privados:

- Código
- Remitente
- Destinatario
- Dirección de Entrega
- Peso en Kg
- Volumen en m³
- Costo por Kg
- Costo por m³

Además, debe haber **un atributo de clase** tipo diccionario para almacenar todas las encomiendas creadas.

Implemente, además:

- Un Constructor que inicialice todos los atributos en base a los parámetros recibidos.
- Propiedades para el acceso set/get para todos los atributos privados.
- Un método que retorne el valor del costo por peso (peso Kg * costo Kg).
- Un método que retorne el valor del costo por volumen (volumen * costo m³).
- Un método que retorne el valor del envío (costo por peso + costo por volumen).

En el programa principal:

- Declare, cree e inicialice varios objeto de tipo Encomienda haciendo uso del constructor.
- Imprima en consola los datos de las encomiendas registradas en el diccionario y costo por peso, costo por volumen y costo de envío.

Caso 2: Terreno

Diseñe la clase Terreno con los atributos de instancia privados:

- Código
- Ubicación
- Área
- Valor m² Comercial
- Valor m² Predial.

Además, debe haber **un atributo de clase** tipo diccionario para almacenar todos los terrenos creados.

Implemente, además:

- Un Constructor que inicialice todos los atributos.
- Propiedades con métodos de acceso set/get para todos los atributos privados.
- Un método que retorne el valor predial del terreno (Área * valor m² Predial).
- Un método que retorne el valor comercial del terreno (Área * valor m² Comercial).
- Un método que retorne el margen de ganancia por el terreno (valor comercial – valor predial).

En el programa principal:

- Declare, cree e inicialice varios objeto de tipo Terreno haciendo uso del constructor.
- Imprima en consola los datos de los terrenos registrados en el diccionario y sus valores predial y comercial.

Caso 3: Pasajeros de un vuelo

Se tendrá una clase vuelo que tendrá como datos un número de vuelo, hora de salida y hora de llegada, ciudad de partida y ciudad de llegada. Hay dos clases de asientos, primera clase (8 asientos) y turista (92 asientos). Asimismo se tendrá la clase pasajeros, cuyos atributos serán número de pasaporte, nombre, edad, tipo de asiento. Se debe crear un programa en Python que permita agregar pasajeros al vuelo

Se debe tener atributos privados y se deben crear los métodos necesarios para agregar y listar los pasajeros del vuelo