

**Comentarios generales:**

- En cada hoja, escriba apellido, nombre, legajo o dni, y número de hoja / el total de hojas entregadas.
- Tenga en cuenta que los 4 ejercicios son **requeridos para aprobar**. No deje ninguno en blanco.
- Debe entregar al menos una hoja con sus datos, aunque no haya completado todo lo pedido.

## Ejercicio 1

Se desea desarrollar una plataforma dedicada a la compra y venta de videos musicales. Las personas podrán publicar sus propios videos y adquirir aquellos compartidos por otros miembros de la comunidad.

De cada persona se conoce su nombre, una dirección de correo electrónico que la identifica y un saldo de créditos. Además, puede tener videos musicales puestos a la venta. Cada vez que otra persona compra uno de sus videos, recibe un ingreso que se suma automáticamente a su saldo de créditos, el cual luego puede utilizar para realizar sus propias compras dentro de la plataforma.

Los videos disponibles en la plataforma se describen mediante un título, una breve descripción, comentarios y un precio expresado en créditos. Cada comentario incluye un texto, la persona que lo escribió y la fecha en que fue realizado. Cada vez que una persona compra un video, se descuentan créditos de su saldo, estos se transfieren al autor del contenido, y el video adquirido se registra en su historial junto con la fecha de compra.

La plataforma debe permitir las siguientes funcionalidades:

- **Recargar créditos:** una persona puede agregar créditos a su cuenta, actualizando así su saldo disponible.
- **Comprar un video:** una persona puede adquirir acceso a un video siempre que cuente con créditos suficientes. Si la compra se concreta, se descuentan los créditos correspondientes de su saldo, se acreditan al autor del video y el video se incorpora a su historial, junto con la fecha de compra y el precio pagado en créditos. Si la persona no tiene créditos suficientes, la compra no se realiza.
- **Calcular los créditos gastados desde una fecha dada:** se debe retornar la cantidad de créditos gastados en videos que se hayan adquirido a partir de la fecha indicada, sumando lo gastado en cada uno.
- **Cambiar precio de un video:** El autor de un video puede modificar su precio expresado en créditos. Este cambio no debe afectar el valor registrado en ventas ya realizadas.
- **Agregar comentario:** Una persona puede agregar un comentario con texto a cualquier video, registrándose automáticamente quién lo escribió y la fecha en que fue creado.

### Tarea:

Realice un diagrama de clases UML donde identifique las clases del dominio, atributos, y las relaciones con sus roles y cardinalidades según corresponda. No debe incluir comportamiento.

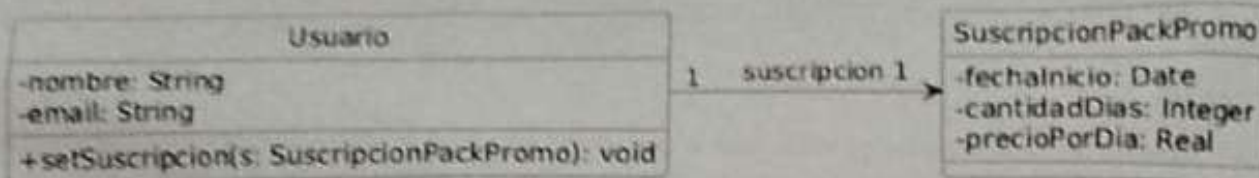
**NOTA:** Asegúrese que su modelo permite resolver todas las funcionalidades pedidas.



## Ejercicio 2

Orientación a Objetos I - 29/11/2025

Se tiene un sistema que permite a los usuarios contratar suscripciones para acceder a servicios digitales. Actualmente, en este sistema existe la **Suscripción Pack Promo**, que cuenta con una fecha de inicio, una duración en días y un precio por día. El sistema permite que el pago se realice incluso después de la fecha de inicio, aplicando en ese caso un **recargo**. El monto a pagar se calcula en una fecha determinada, tomando como precio base el precio por día multiplicado por los días contratados, aplicando un recargo del 30% si el pago se realiza después de la fecha de inicio y sumando el IVA del 21%.



```

public class Usuario {
    private String nombre;
    private String email;
    private SuscripcionPackPromo suscripcion;
  
```

```

    public Usuario(String nombre, String email,
        SuscripcionPackPromo suscripcion) {
        this.nombre = nombre;
        this.email = email;
        this.suscripcion = suscripcion;
    }
  
```

```

    public double montoAPagar(LocalDate
        fechaPago) {
        return this.suscripcion
            .montoAPagar(fechaPago);
    }
  }
  
```

```

public class SuscripcionPackPromo {
    private LocalDate fechaInicio;
    private int cantidadDias;
    private double precioPorDia;
  
```

```

    public SuscripcionPackPromo(LocalDate fechaInicio, int
        cantidadDias, double precioPorDia) { ... }
  
```

```

    private double calcularMontoConIVA(double valor) {
        double iva = valor * 0.21;
        return valor + iva;
    }
  
```

```

    private double calcularRecargo(LocalDate fechaPago,
        double monto) {
        if (fechaPago.isAfter(this.fechaInicio))
            return monto * 0.3;
        return 0.0;
    }
  
```

```

    public double montoAPagar(LocalDate fechaPago) {
        double costo = this.precioPorDia * this.cantidadDias;
        double recargo = this.calcularRecargo(fechaPago, costo);
        return this.calcularMontoConIVA(costo+recargo);
    }
  }
  
```

Se desea agregar un nuevo tipo de **suscripción por uso**, que consiste en un pack con un límite fijo de datos en GB, por ejemplo 8.4GB. Esta suscripción cuenta con una fecha de inicio, una cantidad de GB como límite y un precio por cada GB del paquete. El **monto a pagar se puede calcular en una fecha determinada**, tomando como precio base la cantidad de GB contratada multiplicada por el precio por GB, aplicando un recargo del 30% si el pago se realiza después de la fecha de inicio y sumando el IVA del 21%.

Además, se requiere que un usuario pueda tener **múltiples suscripciones** activas simultáneamente. El monto a pagar de un usuario es la sumatoria de todas las suscripciones activas que tenga.

### Tareas:

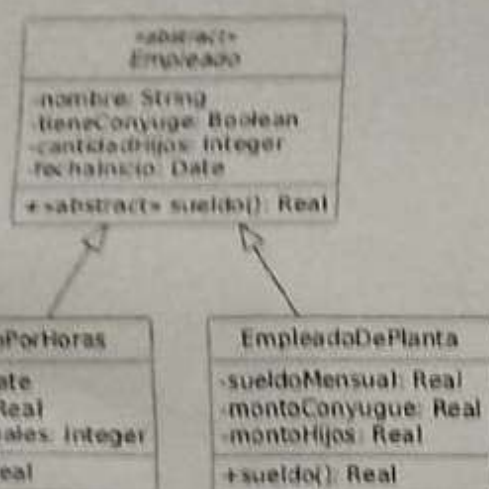
1. Rediseñe lo necesario para permitir que la clase Usuario permita tener una colección de suscripciones, que pueden ser por uso o PackPromo, evitando duplicación de código y calculando los costos polimórficamente. Aplique los conceptos vistos en la materia. Realice un diagrama de clases UML completo de su diseño (incluyendo los métodos y constructores).
2. Reimplemente en Java respetando el nuevo diseño.

### Ejercicio 3

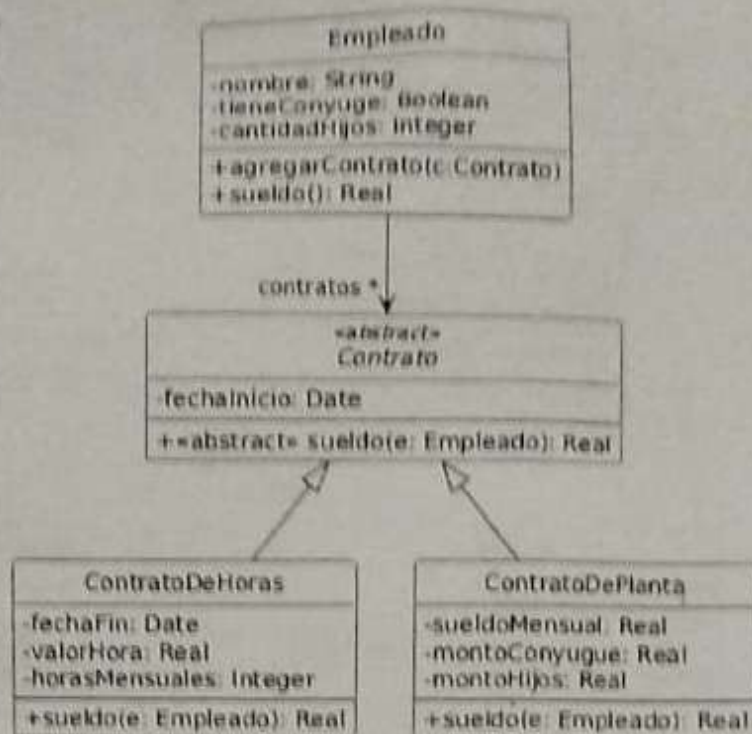
Orientación a Objetos I - 29/11/2025

A continuación se presentan dos posibles modelos para un sistema de administración de personal

**Solución (A)**



**Solución (B)**



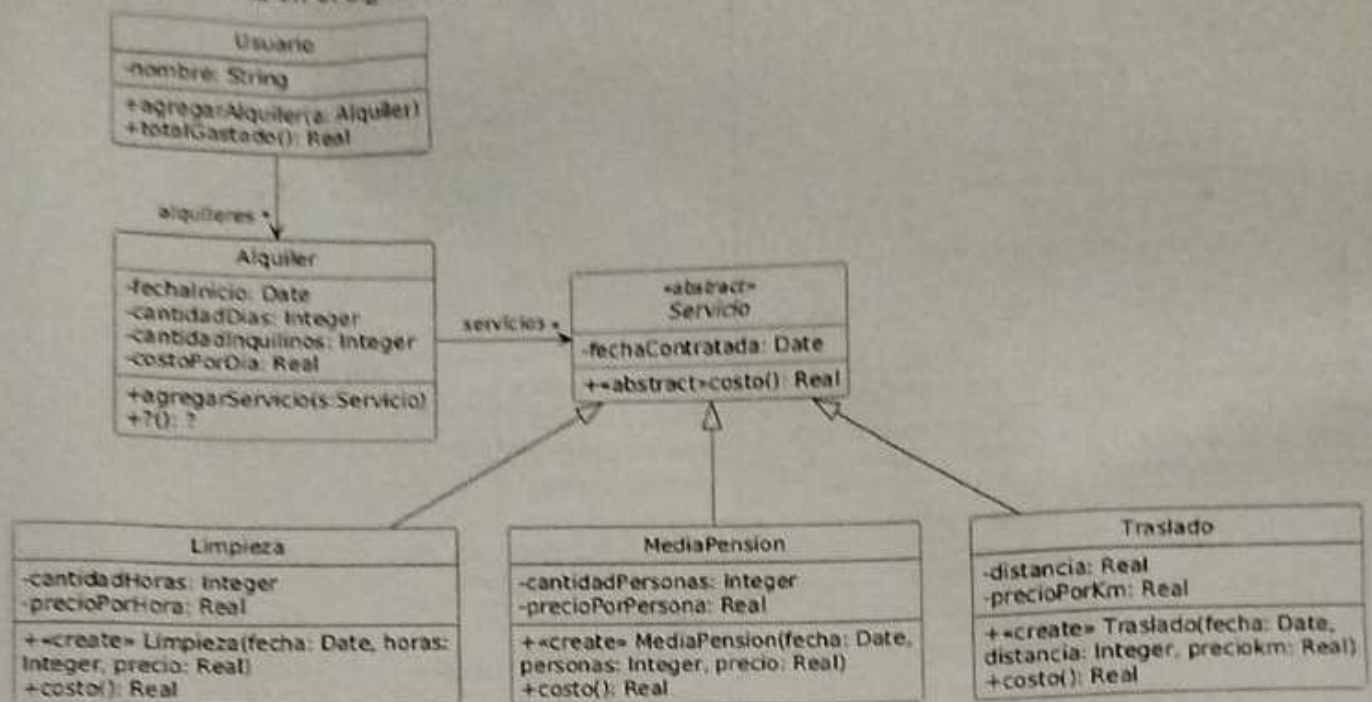
Tarea: Marque con una sola X la opción que considere correcta para cada afirmación.

	Solo (A)	Solo (B)	Ambos	Ninguno
1- Permite al empleado cambiar de contrato dinámicamente.				
2- Permite que un empleado tenga múltiples contratos simultáneamente.				
3- Es posible extender el diseño para tener nuevos empleados con un nuevo tipo de contrato.				
4- Permite reutilizar la implementación de "contrato por horas" en otro sistema sin llevarme todo el código de Empleado.				
5- Si agrego un atributo nuevo a Empleado, automáticamente lo tienen todos los tipos de contrato.				
6- Permite tener un empleado sin contrato.				
7- Permite testear que el cálculo de sueldo funciona bien sin necesidad de crear empleados.				
8- El empleado delega a otro objeto el cálculo del sueldo.				
9- Si agrego la implementación de un nuevo tipo de contrato, la clase Empleado existente NO cambia.				
10- Para cambiar un contrato se debe instanciar un nuevo empleado.				



## Ejercicio 4

Una empresa administra un sistema de alquileres que han sido contratados por distintos usuarios, cuyo diseño se representa en el siguiente diagrama de clases UML.



Cada usuario puede tener múltiples alquileres, y cada alquiler puede incluir diversos servicios adicionales. El costo de un alquiler se calcula como la cantidad de días multiplicado por el costo por día, y a ese resultado se le suma el costo de los servicios asociados.

## Tareas

1. Implemente el constructor de las clases Usuario, Alquiler, Servicio y Traslado.
2. Implemente el método totalGastado() de la clase Usuario, junto con los métodos auxiliares que dicho cálculo requiera. El total gastado se define como la suma de los costos de alquileres que realizó. Utilice Streams cuando resulte apropiado. Considere que los métodos costo de Limpieza, MediaPension y Traslado ya están implementados.
3. Instancie el siguiente usuario con sus alquileres y servicios asociados, utilizando la siguiente información:
  - Usuario: Lautaro Gonzalez, con 2 alquileres:
    - Alquiler 1: fecha de inicio 01/01/2025, 5 días, 2 inquilinos, costo por día 5000.
      - Servicios: Limpieza para el 02/01/2025, 3 horas, 2000 la hora.
      - Media Pensión para 01/01/2025, 2 personas, 2500 por persona.
    - Alquiler 2: fecha de inicio 15/02/2025, 3 días, 1 inquilino, costo por día 6000.
      - Servicio: Traslado para el 15/02/2025, 10 km, precio por km 500.

**Ayuda:** Para instanciar un objeto LocalDate, puede usar `LocalDate.of(year, month, day)`