

## Trabajo Práctico 5: Relaciones UML 1 a 1

Materia: Programación 2

Estudiante: Etchegoyen Gabriel

Enlace a repositorio en GitHub: <https://github.com/Gabriel071185/UTN-TUPaD-P2>

### OBJETIVO GENERAL

Modelar clases con relaciones 1 a 1 utilizando diagramas UML. Identificar correctamente el tipo de relación (asociación, agregación, composición, dependencia) y su dirección, y llevarlas a implementación en Java.

### MARCO TEÓRICO

Concepto	Aplicación en el proyecto
Asociación	Relación entre clases con referencia mutua o directa, puede ser uni o bidireccional
Agregación	Relación de "tiene un" donde los objetos pueden vivir independientemente
Composición	Relación fuerte de contención, el ciclo de vida del objeto contenido depende del otro
Dependencia de uso	Una clase usa otra como parámetro en un método, sin almacenarla como atributo
Dependencia de creación	Una clase crea otra en tiempo de ejecución, sin mantenerla como atributo
Asociación	Relación entre clases con referencia mutua o directa, puede ser uni o bidireccional
Agregación	Relación de "tiene un" donde los objetos pueden vivir independientemente

## Caso Práctico

Desarrollar los siguientes ejercicios en Java. Cada uno deberá incluir:

- Diagrama UML ( en cada proyecto se adjunta en un folder cada diagrama)
- Tipo de relación (asociación, agregación, composición, dependencia)
- Dirección (unidireccional o bidireccional)
- Implementación de las clases con atributos y relaciones definidas

## Conclusiones esperadas

- Diferenciar claramente los tipos de relaciones entre clases (asociación, agregación, composición).
- Representar las relaciones con la dirección adecuada en diagramas UML.
- Comprender e implementar dependencias de uso y de creación.
- Aplicar relaciones 1 a 1 en el diseño e implementación de clases en Java.
- Reforzar el análisis de modelos orientados a objetos y la capacidad de abstracción.

## Aclaraciones

Se desarrollaron 13 ejercicios independientes organizados en paquetes separados (ejercicio1 a ejercicio13), cada uno conteniendo sus respectivas clases Java con implementación completa de atributos, métodos y relaciones UML.

Cada ejercicio incluye una clase principal (main) que demuestra el funcionamiento del sistema mediante instanciación de objetos e invocación de métodos.

Se generaron diagramas UML para cada ejercicio y fueron exportados en formato PNG y organizados en una carpeta dedicada dentro de cada paquete(Diagrama UML) dentro de la carpeta src (source).

## Respuesta:

Solución en repositorio de GitHub: <https://github.com/Gabriel071185/UTN-TUPaD-P2>