

UNIDAD 02 - IDE y JDK

01.- DEFINICIÓN DE IDE

Entorno Integrado de Desarrollo, proporciona herramientas para facilitar el desarrollo de software.

1) Editor de código:

Resalta, colorea y autocompleta el código. Ayuda y lista funciones y métodos. Autocompleta paréntesis, corchetes y indentaciones.

2) Compilador / intérprete:

Detecta errores en tiempo real. Refactorización de código. Separa archivos fuente de los compilados. Permite compilar proyectos complejos fácilmente.

3) Depurador

Permite realizar trazas de variables y establecer breakpoints.

4) Generador automático

Visualización, creación y edición de componentes gráficos. Control de versiones. Generar documentación.

5) Interfaz gráfica

Programar con varios lenguajes. Acceder a bibliotecas y plugins.

02.- DEFINICIÓN DE JDK

Java Development Kit, proporciona herramientas de desarrollo para Java.

- **JAVAPATH:** Ruta del directorio donde está instalado el JDK.
- **CLASSPATH:** Ruta a las bibliotecas y clases.
- **PATH:** Variable del sistema donde se agrega el JAVAPATH.

UNIDAD 03 - UML

01.- DEFINICIÓN DE UML

Lenguaje de Modelado Unificado sirve para expresar modelos basados en diagramas de cada una de las partes del software o hardware.

Utiliza diferentes diagramas que representan los distintos puntos de vista.

02.- DEFINICIÓN DE OBJETO

Conjunto de atributos (datos) y métodos (procedimientos) destinados a resolver tareas comunes.

- Clase: Plantilla para crear objetos.
- Objeto: Instancia de una clase.

03.- PROPIEDADES DE LAS CLASES

1) Abstracción:

Mostrar únicamente las propiedades y acciones necesarias.

2) Encapsulación:

Separar aspectos internos del objeto (private) de aspectos externos (public).

3) Modularidad:

Separar clases en componentes independientes, pero relacionados entre si.

4) Herencia:

Permite al objeto hijo extender la funcionalidad de la clase padre.

5) Polimorfismo:

Agrupar bajo el mismo nombre comportamientos diferentes.

6) Tipificación:

Imposición a un objeto de pertenecer a una clase. Como int, double... pero en clases.

7) Persistencia:

Capacidad para guardarse y recuperarse desde un medio de almacenamiento.

04.- P00

1) Envío de mensajes:

Un objeto indica a otro que tiene que realizar una operación.

2) Asociación:

Los objetos se relacionan entre sí. Puede ser bidireccional. Pueden asociarse en más de una forma

- Multiplicidad → Cantidad de objetos que se relacionan con otros.(1:1, 1:N)

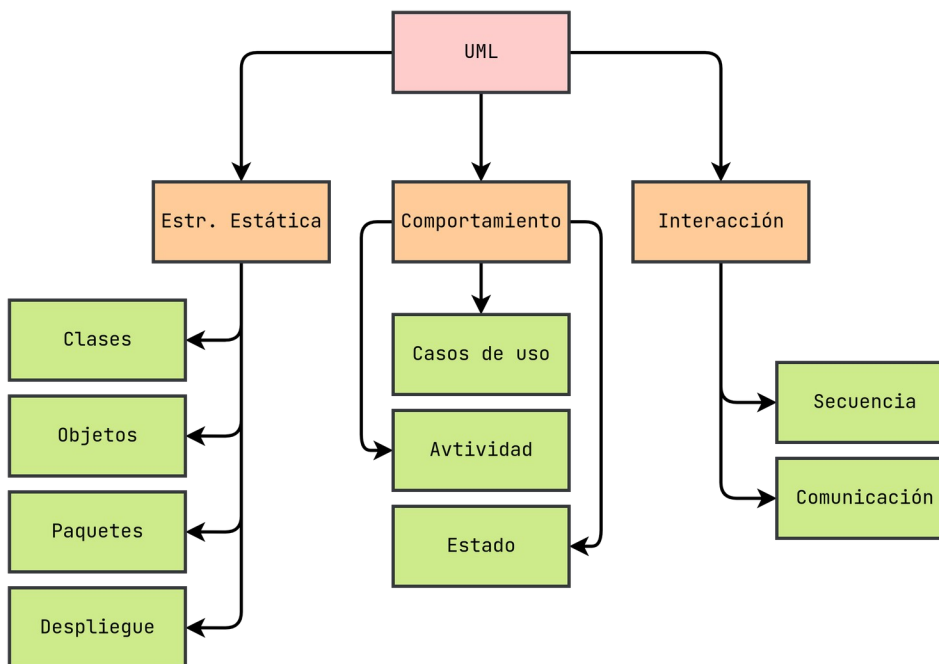
3) Agregación:

Un objeto es compuesto por otros. Tiene sentido que existan por separado.

- Composición → Agregación más fuerte. NO tiene sentido por separado.

	Agregación	Composición
Símbolo		
Varias asociaciones comparten los componentes	SI	NO
Destrucción de los componentes al destruir el compuesto	No	SI
Cardinalidad del compuesto	Cualquiera	0..1 o 1

05.- CATEGORÍAS UML



Estructura estática

- **Diagramas de clases:** Clases que componen el sistema y cómo se relacionan.
- **Diagramas de objeto:** Objetos instanciados y cómo se relacionan en un momento particular.
- **Diagramas de despliegue:** Especifica el hardware y cómo lo utiliza el software.

Comportamiento

- **Diagramas de casos uso:** Como actores usan funciones del sistema y relacionan entre ellos.
- **Diagramas de actividad:** Flujo de decisiones.
- **Diagramas de estado:** Muestra estados, eventos, transiciones y actividades.

Interacción

- **Diagramas de secuencia:** Interacción entre objetos cronológicamente.