

Síntesis conceptual

Grado: Desarrollo de aplicaciones multiplataforma
Asignatura: Entornos de desarrollo
Unidad: 6. Elaboración de diagramas de clases

Resumen

Con el fin de conocer cómo elaborar notaciones de diagramas de clase es necesario conocer el empleo de diversos elementos regidos por el estándar de la UML. Entre estos elementos debemos conocer:

- Clases: Es el elemento básico, ya que de él dependen el resto, sirve como el molde para la introducción de los distintos elementos. No hace referencia a objetos reales, sino a tipos de objetos.
- Instancia: Es la representación de un objeto específico de la realidad a partir del empleo de la clase como modelo para su elaboración.
- Atributos: Elementos que se encuentran en la clase y que designan, a través de un valor, alguna de sus características.
- Notas adjuntas: Permiten el empleo de cuadros de textos dentro del código a modo de recordatorio sin que afecte al programa en sí.
- Métodos: Permiten el empleo de las funciones u operaciones que cada elemento debe realizar.
- Asociaciones: Conjunto de elementos formados por la unión de distintas clases en una asociación dirigida, esta asociación puede ser tanto bipartita como multipartita, generalmente con varias clases unidas a una tercera. Existen diversas variaciones dentro de estas asociaciones como pueden ser:
 - Asociaciones en múltiples niveles.
 - Asociaciones con restricciones.
 - Asociaciones con restricciones de cantidad.
 - Asociaciones reflexivas.
- Herencia: Un tipo particular de relaciones, que no son asociaciones, son las herencias, las cuales nos permite seguir los antecedentes. De modo que podemos realizar agrupaciones a partir de "un árbol genealógico".
- Visibilidad: Con el fin de restringir o permitir la visión de los atributos y de los métodos de una clase por parte de la otra, debemos incluir junto a estos los símbolos que designarán su estatus, existen tres que se emplean principalmente:
 - Nivel público, marcado por un "+".
 - Nivel protegido, marcado por un "#".
 - Nivel privado, marcado por un "-".
- Composición y agregación: Permite la relación de un conjunto de clases que permiten, al unirse, la creación de otra clase, en función de si esta unión es libre de deshacerse o si es una unión duradera residirá la diferencia entre composición y agregación.

Junto con los tiempos las herramientas utilizadas evolucionan, por lo que no debemos ver como algo extraño la aparición de elementos informáticos que nos permitan el desarrollo con facilidad de, entre otras cosas, este tipo de diagrama.

El auge de las herramientas CASE, ingeniería de Software Asistida por Computadora, permite la creación de código a partir de un diagrama y viceversa, facilitando en gran medida el trabajo de los programadores.

Conceptos fundamentales

- **Booleano:** es uno de los tipos de valor, en este caso crea una selección bipartita entre verdadero y falso.
- **UML:** Unified Modeling Language o lenguaje unificado de modelado, es el lenguaje de modelado de Software más empleado.
- **camelCase:** modelo de escritura para palabras compuestas donde la primera palabra empieza en minúscula y la segunda palabra comienza por mayúscula y sin espacio. Ejemplo: nombreUsuario.
- **String:** es uno de los tipos de valor, en este caso crea una línea de caracteres, números y letras.
- **Float:** es uno de los tipos de valor, en este caso crea una lista de números con comas flotantes, es decir, un número con decimales.