

| Elementos | Formato |
|---|---|
| Control de acceso | [<i>public</i> <i>private</i> <i>protected</i>] nombreMétodo() bloque |
| Referencia la objeto actual | <i>This</i> |
| Referencia a la superclase | <i>super</i> |
| Llamada al método de la superclase | <i>super</i> .nombreMétodo() |
| Llamada al constructor de la clase | <i>this</i> (...) |
| Llamada al constructor de la superclase | <i>super</i> (...) |
| Retorna un valor | <i>return</i> [valor] |

Paquetes, interfaces e importación

Tabla A1.10. Paquetes, interfaces e importación

| Elementos | Formato |
|--|--|
| Importación de clase específica | <i>import</i> nombrePaquete.nombreClase; |
| Importación de todas las clases públicas del paquete | <i>import</i> nombrePaquete.*; |
| Las clases del archivo pertenecen a un paquete | <i>package</i> nombrePaquete; |
| Interfaces | <i>interface</i> nombreInterface [<i>extends</i> nombreInterface] bloque [<i>public</i>] <i>interface</i> nombreInterface bloque [<i>abstract</i>] <i>interface</i> nombreInterface bloque |

Excepciones

Tabla A1.11. Excepciones

| Excepciones | Formato |
|---|--|
| Sentencias ejecutadas si una excepción es lanzada | <i>try</i> bloque <i>catch</i> (excepción) bloque [<i>finally</i> bloque] |

Anexo 2. Lenguaje unificado de modelado (UML)

El lenguaje unificado de modelado (UML) es un lenguaje de modelado visual utilizado para visualizar, especificar, construir y documentar sistemas *software* (Booch, Rumbaugh y Jacobson, 2007). UML está conformado por

una colección de diagramas que se clasifican en diagramas estructurales y de comportamiento (Seidl *et al.*, 2015).

En este libro se ha utilizado la notación de cuatro diagramas UML:

- ▶ Diagramas de clases
- ▶ Diagramas de objetos
- ▶ Diagramas de máquinas de estados
- ▶ Diagramas de actividad

A continuación, se describirá brevemente la notación gráfica de cada uno de estos diagramas.

Diagramas de clases

Una clase representa un conjunto de objetos similares que se encuentran en el sistema. Las clases poseen atributos y métodos. Un atributo permite almacenar información conocida para todas las instancias, pero que tiene valores específicos diferentes para cada instancia. Por su parte, los métodos determinan un comportamiento específico en objetos concretos (Booch, Rumbaugh y Jacobson, 2017).

Las clases se representan por medio de rectángulos con tres compartimientos (figura A2.1). El primer compartimiento es para el nombre de la clase; el segundo, para el listado de atributos de la clase y el tercero, para el listado de métodos de la clase.

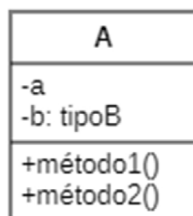
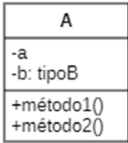



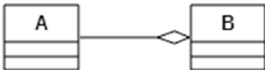
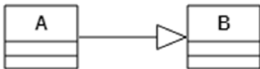

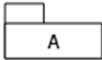



Figura A2.1. Notación de una clase en UML

Los diagramas de clase UML son diagramas estructurales. Los diagramas de clases permiten modelar la estructura estática de un sistema describiendo los elementos de este y sus relaciones entre ellos (Rumpe, 2016). Los diagramas de clase son los diagramas UML más usados.

La notación gráfica de los diagramas de clases UML es extensa. En la tabla A2.1 se presenta un resumen de los conceptos más importantes de los diagramas de clase utilizados en los ejercicios desarrollados en este libro.

Tabla A2.1. Conceptos de diagramas de clases UML

| Nombre | Notación | Descripción |
|-----------------|---|---|
| Clase |  | Descripción de la estructura y comportamiento de un conjunto de objetos. |
| Clase abstracta |  | Clases que no pueden ser instanciadas. El nombre de la clase está en cursiva. |
| Asociación |  | Relaciones entre clases. |
| Composición |  | Relaciones partes-todo dependientes (A es parte de B; si B es eliminado, las instancias relacionadas de A son eliminadas). |
| Agregación |  | Relaciones partes-todo (A es parte de B; si B es eliminado, las instancias relacionadas de A no son eliminadas). |
| Generalización |  | Relación de herencia (A hereda de B). |
| Interface |  | Elemento que declara un conjunto de métodos sin su implementación. Especifica un servicio proporcionado por una clase (A implementa B). |
| Paquetes |  | Elemento organizativo de diagramas, puede agrupar elementos de cualquier tipo si están relacionados semánticamente. |
| Clase genérica |  | Plantilla de clase que se puede parametrizar con uno o más tipos de datos. |

Fuente: adaptado de Seidl *et al.* (2015).

Diagramas de objetos

En UML se describen los objetos concretos de un sistema y sus relaciones (enlaces) utilizando diagramas de objetos.

Los objetos representan formas concretas de las clases y son referidos como instancias. Un objeto tiene una identidad única y una serie de características que lo describen con más detalle. Este generalmente interactúa y se comunica con otros objetos. Las relaciones entre objetos se denominan enlaces. Las características de un objeto incluyen sus características estructurales (atributos) y su comportamiento (métodos). Valores concretos se asignan a los atributos en el diagrama de objetos; las operaciones usualmente no se indican (Schildt, 2018).

Los diagramas de objetos UML son diagramas estructurales. En un diagrama de objetos, un objeto se muestra como un rectángulo que se puede subdividir en dos compartimentos. El primer compartimento contiene información en la forma nombreObjeto: Clase (figura A2.2[a]). La segunda notación válida es el nombre del objeto sin colocar su tipo (figura A2.2[b]). La última notación es un objeto anónimo que especifica su tipo, pero no su nombre (figura A2.2[c]) (Booch, Rumbaugh y Jacobson, 2017).



Figura A2.2. Notación de objetos



Si el rectángulo tiene un segundo compartimiento, este contiene los atributos del objeto y los valores actuales de sus atributos. Un enlace se representa como una línea continua que conecta los objetos que están relacionados entre sí (figura A2.3).



Figura A2.3. Notación de objetos relacionados y con valores de sus atributos

En la tabla A2.2 se presenta un resumen de los conceptos más importantes de los diagramas de objetos.

Tabla A2.2. Conceptos principales de los diagramas de objetos UML

| Nombre | Notación | Descripción |
|--------|---|--------------------------|
| Objeto |  | Instancia de una clase |
| Enlace |  | Relaciones entre objetos |

Fuente: adaptado de Seidl *et al.* (2015).

Diagramas de máquinas de estado

Los objetos pasan por diferentes estados a lo largo de su ciclo de vida. Estos cambios (de un estado a otro) se representan en UML utilizando un diagrama de máquinas de estado que describen el comportamiento permitido a un objeto en forma de posibles estados y transiciones de estado iniciados por varios eventos (Booch, Rumbaugh y Jacobson, 2017).

El diagrama de máquinas de estado UML es un diagrama de comportamiento. Este es un gráfico con estados como nodos y transiciones de estado como flechas (Seidl *et al.*, 2015). Un estado se muestra como un rectángulo con esquinas redondeadas y se etiqueta con el nombre del estado (figura A2.4).



Figura A2.4. Notación gráfica de un estado en UML

El cambio de un estado a otro se denomina transición. Esta se representa con una flecha que indica la dirección de la transición. De acuerdo con Booch, Rumbaugh y Jacobson (2017), se pueden especificar varias propiedades para una transición (figura A2.5).

- **El evento** que desencadena la transición de estado (e).
- **Condición de guarda** que permite la ejecución de la transición (g).
- **Actividades** ejecutadas durante el cambio al estado objetivo (a).

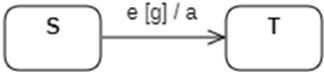


Figura A2.5. Notación gráfica para una transición