**Declaración de una clase con un método e instanciamiento de un objeto de una clase**

Cada declaración de clase que comienza con la palabra clave ***public*** debe almacenarse en un archivo que tenga el mismo nombre que la clase y que termine con la extensión del archivo .java

**Error común de programación 3.1**

Declarar más de una clase ***public*** en el mismo archivo es un error de compilación. Recuerde que una clase es como un plano de construcción.

La palabra clave ***public*** es un modificador de acceso.

Por ahora, simplemente declararemos cada clase como ***public***. Toda declaración de clase contiene la palabra clave ***class***, seguida inmediatamente por el nombre de la clase. El cuerpo de toda clase se encierra entre una llave izquierda y una derecha ({ y }).

Recuerde que main es un método especial, que siempre es llamado, automáticamente, por la Máquina Virtual de Java (JVM) a la hora de ejecutar una aplicación.

La mayoría de los métodos no se llaman en forma automática.

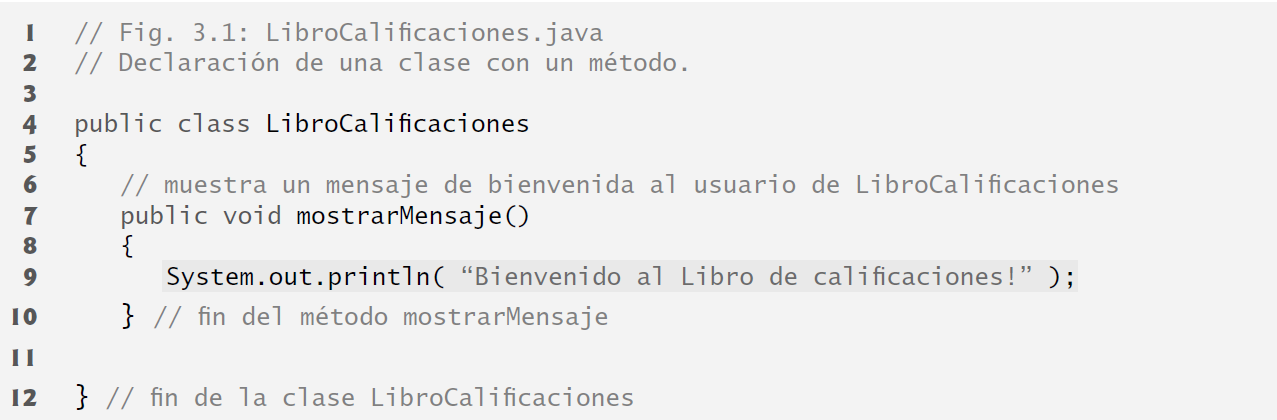
La declaración del método comienza con la palabra clave ***public*** para indicar que el método está “disponible al público”; es decir, los métodos de otras clases pueden llamarlo desde el exterior del cuerpo de la declaración de la clase.

La palabra clave ***void*** indica que este método realizará una tarea, pero no devolverá (es decir, regresará) información al método que hizo la llamada cuando complete su tarea.

El nombre del método, mostrarMensaje, va después del tipo de valor de retorno. Por convención, los nombres de los métodos comienzan con una letra minúscula, y el resto de las palabras en el nombre empiezan con letra mayúscula.

Los paréntesis después del nombre del método indican que éste es un método. Un conjunto vacío de paréntesis, como se muestra en la línea 7, indica que este método no requiere información adicional para realizar su tarea.

La línea 7 se conoce comúnmente como el encabezado del método. El cuerpo de cada método se delimita mediante una llave izquierda y una llave derecha ({ y }), como en las líneas 8 y 10.



*Imagen referencial. Declaración de una clase con un método*

**Declaración de una clase con un método e instanciamiento de un objeto de una clase**

El cuerpo de un método contiene una o varias instrucciones que realizan su trabajo, el método ***main*** empieza la ejecución de todas las aplicaciones. Una clase que contiene el método ***main***

es una aplicación de Java. Dicha clase es especial, ya que la JVM puede utilizar a main como un punto de entrada para empezar la ejecución. La clase ***LibroCalificaciones*** no es una aplicación, ya que no contiene a ***main***.

Para corregir este problema con la clase LibroCalificaciones, debemos declarar una clase separada que contenga un método ***main***, o colocar un método ***main*** en la clase LibroCalificaciones.

**La clase PruebaLibroCalificaciones**

Cualquier clase que contiene el método ***main***, declarado como se muestra en la línea 7, puede utilizarse para ejecutar una aplicación. La declaración de la clase PruebaLibroCalificaciones empieza en la línea 4 y termina en la línea 16. La clase sólo contiene un método ***main***, algo común en muchas clases que empiezan la ejecución de una aplicación.

Una parte clave para permitir que la JVM localice y llame al método ***main*** para empezar la ejecución de la aplicación es la palabra clave ***static*** (línea 7), la cual indica que main es un método ***static***. Un método ***static*** es especial, ya que puede llamarse sin tener que crear primero un objeto de la clase en la cual se declara ese método.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

En esta aplicación nos gustaría llamar al método mostrarMensaje de la clase LibroCalificaciones para mostrar el mensaje de bienvenida en la ventana de comandos. Por lo general, no podemos llamar a un método que pertenece a otra clase, sino hasta crear un objeto de esa clase, como se muestra en la línea 10. Empezaremos por declarar la variable miLibroCalificaciones. Observe que el tipo de la variable es LibroCalificaciones.

Cada nueva clase que creamos se convierte en un nuevo tipo, que puede usarse para declarar variables y crear objetos. Los programadores pueden declarar nuevos tipos de clases según lo necesiten; ésta es una razón por la cual Java se conoce como un lenguaje extensible.

La variable miLibroCalificaciones se inicializa con el resultado de la expresión de creación de instancia de clase ***new*** LibroCalificaciones(). La palabra clave ***new*** crea un nuevo objeto de la clase especifi cada a la derecha de la palabra clave (es decir, LibroCalificaciones). Los paréntesis a la derecha de LibroCalificaciones son obligatorios.

Esos paréntesis en combinación con el nombre de una clase representan una llamada a un constructor, que es similar a un método, pero se utiliza sólo cuando se crea un objeto, para inicializar los datos de éste. En esa sección verá que los datos pueden colocarse entre paréntesis para especificar los valores iniciales para los datos del objeto.

Así como podemos usar el objeto System.out para llamar a los métodos print, printf y println, también podemos usar el objeto miLibroCalificaciones para llamar al método mostrarMensaje, usando miLibroCalificaciones seguida de un separador punto (.), el nombre del método mostrarMensaje y un conjunto vacío de paréntesis. Esta llamada hace que el método mostrarMensaje realice su tarea.

Al inicio de la línea 13, “miLibroCalificaciones”. Indica que main debe utilizar el objeto miLibroCalificaciones que se creó en la línea 10. La línea 7 de la figura 3.1 indica que el método mostrarMensaje tiene una lista de parámetros vacía; es decir, mostrarMensaje no requiere información adicional para realizar su tarea.

Por esta razón, la llamada al método (línea 13 de la fi gura 3.2) especifica un conjunto vacío de paréntesis después del nombre del método, para indicar que no se van a pasar argumentos al método mostrarMensaje. Cuando el método mostrarMensaje completa su tarea, el método main continúa su ejecución en la línea 14. Éste es el fi nal del método ***main***, por lo que el programa termina.

Diagrama de clases de UML para la clase LibroCalificaciones

UML es un lenguaje gráfico, utilizado por los programadores para representar sistemas orientados a objetos en forma estandarizada. En UML, cada clase se modela en un diagrama de clases en forma de un rectángulo con tres componentes. El compartimiento superior contiene el nombre de la clase, centrado en forma horizontal y en negrita. El compartimiento de en medio contiene los atributos de la clase, que en Java corresponden a las variables de instancia.

El compartimiento de en medio está vacío, ya que la versión de la clase LibroCalificaciones en la fi gura 3.1 no tiene atributos. El compartimiento inferior contiene las operaciones de la clase, que en Java corresponden a los métodos.

UML lista el nombre de la operación precedido por un modifi cador de acceso y seguido de un conjunto de paréntesis. La clase LibroCalificaciones tiene un solo método llamado mostrarMensaje, por lo que el compartimiento inferior de la fi gura 3.3 lista una operación con este nombre.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

El signo más (+) que va antes del nombre de la operación indica que mostrarMensaje es una operación public en UML (es decir, un método public en Java). Utilizaremos los diagramas de clases de UML a menudo para sintetizar los atributos y las operaciones de una clase.

**Declaración de un método con un parámetro**