

## Recomendaciones para la gestión de archivos de Java

Aunque la especificación del lenguaje Java no obliga a ello, se suele emplear el siguiente procedimiento para construir un sistema jerárquico que facilite la gestión y uso de los archivos fuente (.java) y compilados (.class):

- a) Almacenar el código fuente de una clase o interfaz en un archivo de texto con el mismo identificador que la clase o interfaz y extensión .java. Por ejemplo: Rectangulo.java.
- b) Poner el archivo en un directorio con el mismo nombre que el paquete al que pertenezca la clase o la interfaz. Por ejemplo: geometria/Rectangulo.java. El identificador *cualificado* del componente del paquete (geometria.Rectangulo) y la vía de acceso al archivo son ahora paralelos o equivalentes.
- c) Por convención, una organización emplea su nombre de dominio a la inversa para identificar los *packages* que desarrolla. Por ejemplo, la empresa ficticia Novedades García y Arranz con nombre de dominio garciayarranz.com añadiría com.garciayarranz al principio de los identificadores de sus *packages*. Así, si la empresa García y Arranz tuviera un paquete geometria que contuviera un archivo fuente Rectangulo.java, debería incluirse en una secuencia de directorios de la siguiente forma:

```
package com.garciayarranz.geometria;
```

```
public class Rectangulo {  
    ...  
}
```

- d) Al compilar un archivo fuente, el compilador genera un archivo de *bytecodes* distinto para cada clase e interfaz declarada en el archivo fuente. El nombre del archivo es el identificador de la clase o interfaz y la extensión es .class. Los archivos con extensión .class suelen organizarse en una estructura específica de directorios que refleja el nombre del paquete y diferente de la estructura utilizada para organizar los archivos fuente. Por ejemplo:

Archivo de *bytecodes*: clases/com/garciayarranz/geometria/Rectangulo.class

Archivo fuente: fuentes/com/garciayarranz/geometria/Rectangulo.java

- e) El directorio raíz de la estructura de directorios debe incluirse en la vía del acceso a las clases del ordenador. Esta vía de acceso se define mediante la variable de entorno CLASSPATH.

Cuando el compilador o el intérprete Java necesitan localizar una clase en un paquete, buscan en cada uno de los directorios asignados a CLASSPATH para ver si el árbol de directorios que contiene el paquete se encuentra ahí.

- f) Esta estructura facilita que tanto el compilador como el intérprete de Java busquen y encuentren las clases en el directorio apropiado. Si se desea los archivos pueden encontrarse dentro de archivos comprimidos en formato ZIP.