## [Bài đọc] Lớp File (tệp)

*(Nguồn: Introduction to Java Programming, Comprehensive Version (10th Edition)  - Y. Daniel Liang – Tr 473)*

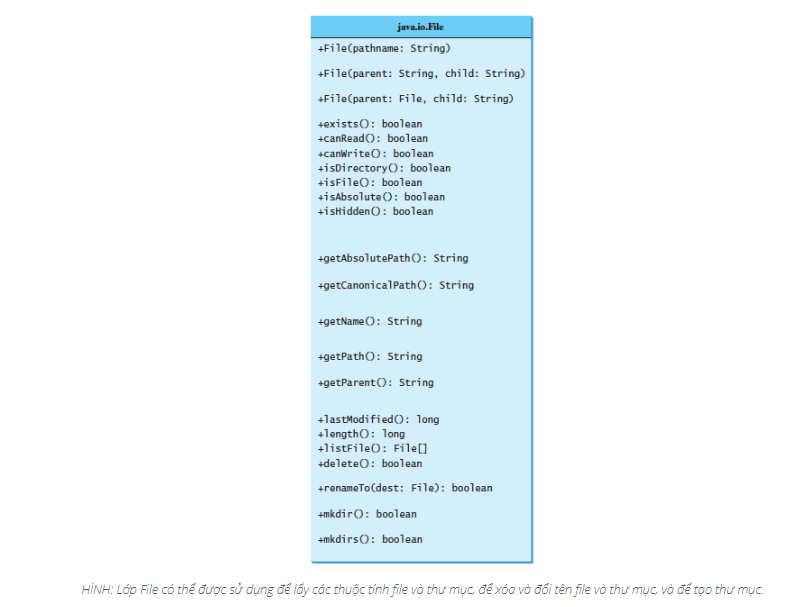
Lớp File chứa các phương thức để thao tác với tệp / thư mục.

Dữ liệu được lưu trữ trong chương trình là tạm thời; chúng bị mất khi chương trình kết thúc. Để lưu trữ vĩnh viễn dữ liệu được tạo ra trong một chương trình, bạn cần lưu chúng vào một tệp trên đĩa hoặc thiết bị lưu trữ. Sau đó, tệp có thể được đọc bằng các chương trình khác.

Mọi file (tệp) được đặt trong một thư mục trong hệ thống file. Tên file tuyệt đối (tên đầy đủ) chứa tên file với đường dẫn đầy đủ. Ví dụ: “c:\book\Welcome.java” là tên file tuyệt đối cho file Welcome.java trên hệ điều hành Windows. Ở đây “c:\book” được gọi là đường dẫn thư mục cho file. Tên file tuyệt đối phụ thuộc vào máy. Trên nền tảng UNIX, tên file tuyệt đối có thể là “/home/liang/book/Welcome.java”, trong đó “/home/liang/book” là đường dẫn thư mục cho file Welcome.java.

Tên file tương đối có liên quan đến thư mục làm việc hiện tại. Đường dẫn thư mục hoàn chỉnh cho tên file tương đối bị bỏ qua. Ví dụ: Welcome.java là tên file tương đối. Nếu thư mục làm việc hiện tại là “c:\book”, thì tên file tuyệt đối sẽ là “c:\book\Welcome.java”.

Lớp File chứa các phương thức để lấy thuộc tính file và thư mục cũng như để đổi tên và xóa file và thư mục. Tuy nhiên, lớp file không chứa các phương thức đọc và ghi nội dung file. Tên file là một chuỗi.  Ví dụ, câu lệnh: new File("c:\\book") tạo một đối tượng File cho thư mục “c:\book” và câu lệnh: new File("c:\\book\\test.dat") tạo một đối tượng File cho file c:\book\test.dat, cả hai đều trên Windows. Bạn có thể sử dụng phương thức isDirectory() của lớp File để kiểm tra xem đối tượng có đại diện cho một thư mục hay không và phương thức isFile() để kiểm tra xem đối tượng có đại diện cho một file hay không.



Không sử dụng tên file tuyệt đối trong chương trình của bạn. Nếu bạn sử dụng tên file như c:\\book\\Welcome.java, nó sẽ hoạt động trên Windows nhưng không hoạt động trên các nền tảng khác. Bạn nên sử dụng một tên file tương đối với thư mục hiện tại. Ví dụ: bạn có thể tạo một đối tượng File bằng cách sử dụng câu lệnh new File("Welcome.java") cho file Welcome.java trong thư mục hiện tại. Bạn có thể tạo một đối tượng File sử dụng câu lệnh new File("image/us.gif") cho file us.gif trong thư mục image trong thư mục hiện tại. Dấu gạch chéo (/) là dấu phân tách thư mục trong Java.

Ví dụ sau trình bày cách tạo đối tượng File và sử dụng các phương thức trong File để có được các thuộc tính của nó. Chương trình tạo một đối tượng File cho file us.gif. File này được lưu trữ dưới thư mục image trong thư mục hiện tại.

TestFileClass.java

1 public class TestFileClass {

2 public static void main(String[] args) {

3 java.io.File file = new java.io.File("image/us.gif");

4 System.out.println("Does it exist? " + file.exists());

5 System.out.println("The file has " + file.length() + " bytes");

6 System.out.println("Can it be read? " + file.canRead());

7 System.out.println("Can it be written? " + file.canWrite());

8 System.out.println("Is it a directory? " + file.isDirectory());

9 System.out.println("Is it a file? " + file.isFile());

10 System.out.println("Is it absolute? " + file.isAbsolute());

11 System.out.println("Is it hidden? " + file.isHidden());

12 System.out.println("Absolute path is " +

13 file.getAbsolutePath());

14 System.out.println("Last modified on " +

15 new java.util.Date(file.lastModified()));

16 }

17 }

Kết quả:

