# Mục tiêu

Luyện tập sử dụng thuật toán đệ quy.

# Mô tả

Trong phần này, chúng ta sẽ phát triển một ứng dụng cho phép liệt kê cây các thư mục con và file của một thư mục cho trước.

Ứng dụng cho phép nhập vào đường dẫn của một thư mục, sau đó sẽ hiển thị cây bao gồm các thư mục con, cháu… và file ở bên trong thư mục đó

# Hướng dẫn

*Sử dụng lớp java.io.File để làm việc với File hoặc thư mục.*

*- phương thức* ***exists()*** *để kiểm tra thư mục hoặc file có tồn tại hay không*

*- phương thưc* ***isDirectory()*** *để kiểm tra đối tượng có phải là thư mục không*

*- phương thức* ***listFiles()*** *để lấy danh sách thư mục hoặc file trong một thư mục*

*- phương thức* ***getCanonicalPath()*** *để lấy đường dẫn của thư mục hoặc file*

Bước 1: Viết hàm hiện thị cây thư mục với giải thuật đệ quy.

static void directoryTree(String path) throws IOException {  
 File dir = new File(path);  
 if (!dir.exists()) {  
 System.*out*.println("Folder " + path + " does not exist");  
 }  
 if (dir.isDirectory()) {  
 File[] list = dir.listFiles();  
 if (list != null) {  
 for (File f : list) {  
 System.*out*.println(f.getCanonicalPath());  
 *directoryTree*(f.getCanonicalPath());  
 }  
 }  
 }  
}

Với mỗi thư mục/file con trong thư mục cha, tiếp tục gọi lại hàm hiển thị cây thư mục với tham số là đường dẫn của thư mục/file con. Việc gọi đệ quy sẽ dừng lại khi ***path*** không phải là đường dẫn của thư mục hoặc bên trong nó không chứa bất cứ thư mục hoặc file nào.

Bước 2: Viết chương trình thực thi trong hàm main, khai báo biến lưu đường dẫn của thư mục cần làm việc.

public static void main(String[] args) throws IOException {  
 System.*out*.println("The program shows the directory tree");  
  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.println("Enter the path of the directory: ");  
 String path = scanner.nextLine();  
  
 //code below here  
  
}

Bước 3: Gọi hàm hiển thị cây thư mục với tham số là biến lưu đường dẫn thư mục

*directoryTree*(path);

Bước 4: Thực thi chương trình và quan sát kết quả.