import java.io.File;

import javax.swing.JFrame;

import javax.swing.JScrollPane;

import javax.swing.JTree;

import javax.swing.SwingUtilities;

import javax.swing.tree.DefaultMutableTreeNode;

import javax.swing.tree.DefaultTreeModel;

public class FileBrowser implements Runnable

{

private DefaultMutableTreeNode root;

private DefaultTreeModel treeModel;

private JTree tree;

// @Override

public void run() {

JFrame frame = new JFrame("File Browser");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

File fileRoot = new File("C:/");

root = new DefaultMutableTreeNode(new FileNode(fileRoot));

treeModel = new DefaultTreeModel(root);

tree = new JTree(treeModel);

tree.setShowsRootHandles(true);

JScrollPane scrollPane = new JScrollPane(tree);

frame.add(scrollPane);

frame.setLocationByPlatform(true);

frame.setSize(640, 480);

frame.setVisible(true);

CreateChildNodes ccn = new CreateChildNodes(fileRoot, root);

new Thread(ccn).start();

}

public static void main(String[] args) {

SwingUtilities.invokeLater(new FileBrowser());

}

public class CreateChildNodes implements Runnable {

private DefaultMutableTreeNode root;

private File fileRoot;

public CreateChildNodes(File fileRoot,

DefaultMutableTreeNode root) {

this.fileRoot = fileRoot;

this.root = root;

}

// @Override

public void run() {

createChildren(fileRoot, root);

}

private void createChildren(File fileRoot,

DefaultMutableTreeNode node) {

File[] files = fileRoot.listFiles();

if (files == null) return;

for (File file : files) {

DefaultMutableTreeNode childNode =

new DefaultMutableTreeNode(new FileNode(file));

node.add(childNode);

if (file.isDirectory()) {

createChildren(file, childNode);

}

}

}}

public class FileNode {

private File file;

public FileNode(File file) {

this.file = file;

}

@Override

public String toString() {

String name = file.getName();

if (name.equals("")) {

return file.getAbsolutePath();

}

else {

return name;

}}}}

OUTPUT:

