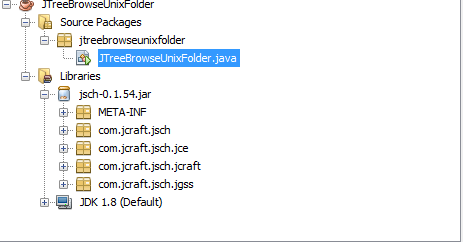
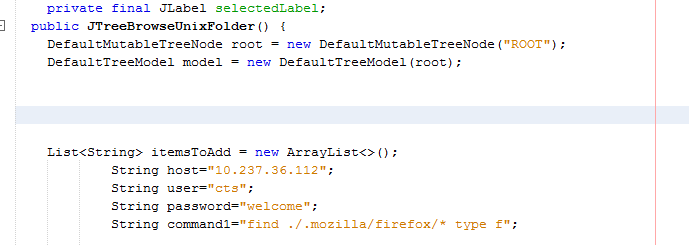
**JTREE TO BROWSE UNIX FOLDERS:**

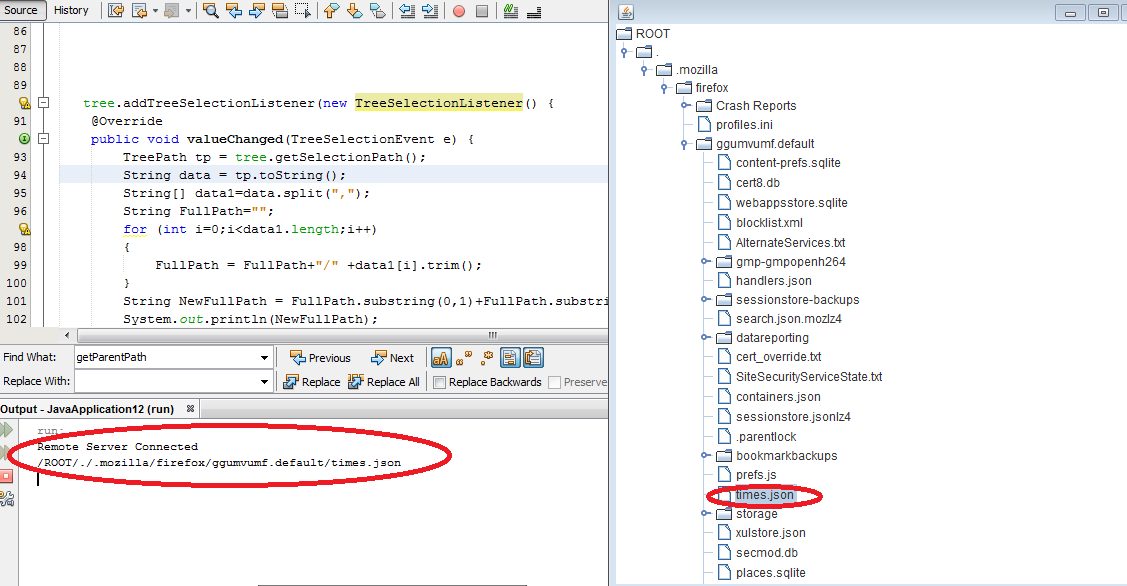
1. Required Libraries:



1. Change below code snippet host, user, password & command 1 if some other UNIX connection (or) get it from property file and pass it.



1. Run the code after setting above things



1. Code Snippet – Copy paste to NetBeans

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package jtreebrowseunixfolder;

import com.jcraft.jsch.Channel;

import com.jcraft.jsch.ChannelExec;

import com.jcraft.jsch.JSch;

import com.jcraft.jsch.Session;

import java.awt.BorderLayout;

import java.io.BufferedReader;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Enumeration;

import java.util.List;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JScrollPane;

import javax.swing.JTree;

import javax.swing.event.TreeSelectionEvent;

import javax.swing.event.TreeSelectionListener;

import javax.swing.tree.DefaultMutableTreeNode;

import javax.swing.tree.DefaultTreeModel;

import javax.swing.tree.TreePath;

/\*\*

\*

\* @author 257592

\*/

public class JTreeBrowseUnixFolder extends JFrame{

protected final JTree tree;

private final JLabel selectedLabel;

public JTreeBrowseUnixFolder() {

DefaultMutableTreeNode root = new DefaultMutableTreeNode("ROOT");

DefaultTreeModel model = new DefaultTreeModel(root);

List<String> itemsToAdd = new ArrayList<>();

String host="10.237.36.112";

String user="cts";

String password="welcome";

String command1="find ./.mozilla/firefox/\* -type f";

try

{

java.util.Properties config = new java.util.Properties();

config.put("PreferredAuthentications", "publickey,keyboard-interactive,password");

config.put("StrictHostKeyChecking", "no");

JSch jsch = new JSch();

Session session=jsch.getSession(user, host, 22);

session.setPassword(password);

session.setConfig(config);

session.connect();

System.out.println("Remote Server Connected");

Channel channel=session.openChannel("exec");

((ChannelExec)channel).setCommand(command1);

channel.setInputStream(null);

((ChannelExec)channel).setErrStream(System.err);

InputStream in=channel.getInputStream();

channel.connect();

BufferedReader reader = new BufferedReader(new InputStreamReader(in));

String line;

// int index = 0;

while ((line = reader.readLine()) != null)

{

itemsToAdd.add(line);

}

channel.disconnect();

session.disconnect();

// System.out.println("DONE");

}

catch(Exception e)

{

e.printStackTrace();

}

// String slist1[]=new String[itemsToAdd.size()] ;

// slist1 = itemsToAdd.toArray(slist1);

// System.out.println("Size:"+itemsToAdd.size());

for (String s : itemsToAdd )

{

buildTreeFromString(model, s);

}

tree = new JTree(model);

// ImageIcon imageIcon = new ImageIcon(TreeExample.class.getResource("/leaf.jpg"));

// DefaultTreeCellRenderer renderer = new DefaultTreeCellRenderer();

// renderer.setLeafIcon(imageIcon);

// tree.setCellRenderer(renderer);

tree.setShowsRootHandles(true);

tree.setRootVisible(false);

add(new JScrollPane(tree));

selectedLabel = new JLabel();

add(selectedLabel, BorderLayout.SOUTH);

// tree.addMouseListener(this);

// f.add(new JScrollPane(), BorderLayout.CENTER);

tree.addTreeSelectionListener(new TreeSelectionListener() {

@Override

public void valueChanged(TreeSelectionEvent e) {

TreePath tp = tree.getSelectionPath();

String data = tp.toString();

String[] data1=data.split(",");

String FullPath="";

for (int i=0;i<data1.length;i++)

{

FullPath = FullPath+"/" +data1[i].trim();

}

// System.out.println(FullPath);

String NewFullPath = FullPath.substring(0,1)+FullPath.substring(2,FullPath.length()-1);

System.out.println(NewFullPath);

// System.out.println(path.getPath());

// System.out.println(tp.());

}

});

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

this.setTitle("Browse Remote Folders");

this.setSize(1000, 1200);

this.setVisible(true);

}

private void buildTreeFromString(final DefaultTreeModel model,

final String str) {

DefaultMutableTreeNode root = (DefaultMutableTreeNode) model.getRoot();

String[] strings = str.split("/");

DefaultMutableTreeNode node = root;

for (String s : strings) {

int index = childIndex(node, s);

if (index < 0) {

DefaultMutableTreeNode newChild = new DefaultMutableTreeNode(s);

node.insert(newChild, node.getChildCount());

node = newChild;

} else {

node = (DefaultMutableTreeNode) node.getChildAt(index);

}

}

}

private int childIndex(final DefaultMutableTreeNode node,

final String childValue) {

Enumeration<DefaultMutableTreeNode> children = node.children();

DefaultMutableTreeNode child = null;

int index = -1;

while (children.hasMoreElements() && index < 0) {

child = children.nextElement();

if (child.getUserObject() != null

&& childValue.equals(child.getUserObject())) {

index = node.getIndex(child);

}

}

return index;

}

public static void main(String[] args) {

new JTreeBrowseUnixFolder();

// dialog.setVisible(true);

}

}