Q1-Write a Java program, to take a HDFS Path as input and display all the files and sub-directories in that HDFS path.

Program:

package hdfs;

import java.io.\*;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.FileSystem;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.fs.FileStatus;

public class FileListing {

public static void main(String[] args) {

if (args.length != 1)

{

System.out.println("Pass one argument");

System.exit(1);

}

Path path = new Path(args[0]);

try

{

Configuration conf = new Configuration();

FileSystem fileSystem = FileSystem.get(path.toUri(), conf);

FileStatus[] fileStatus=fileSystem.listStatus(path);

for (FileStatus fStat : fileStatus)

{

if (fStat.isDirectory())

{

System.out.println("Directory: " + fStat.getPath());

}

else if (fStat.isFile())

{

System.out.println("File: " + fStat.getPath());

}

else if (fStat.isSymlink())

{

System.out.println("Symlink: " + fStat.getPath());

}

}

}

catch (IOException e)

{

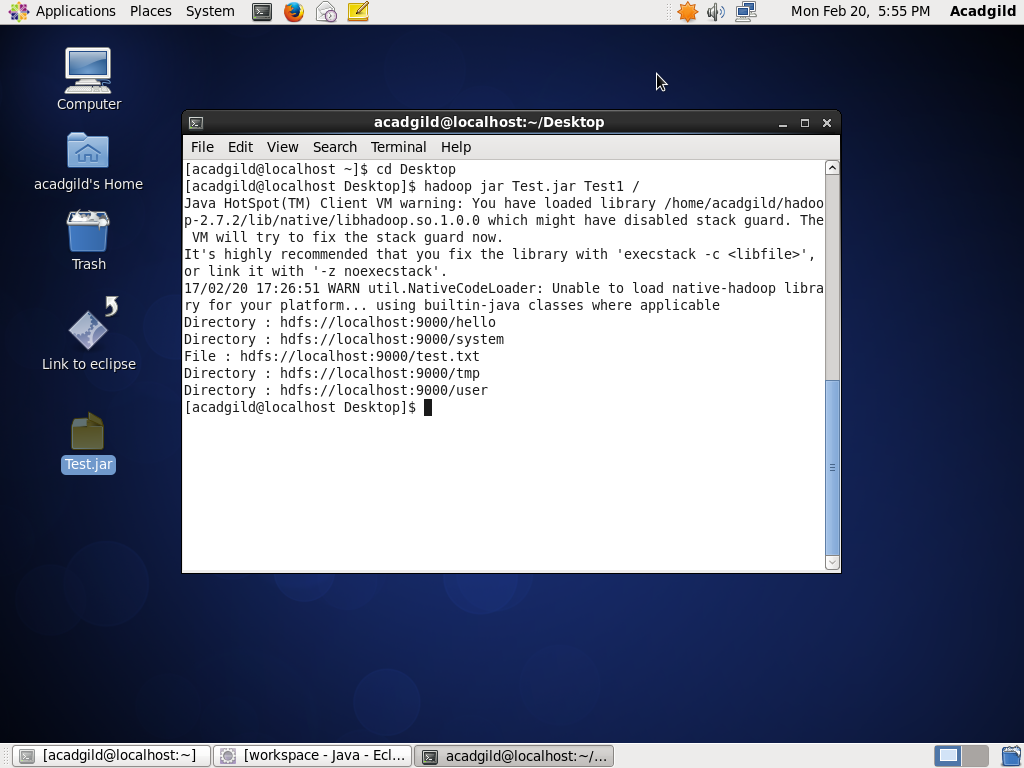
e.printStackTrace();

}

}

}

Output:



Q2-Modify the previous program to list all the files and sub-directories in the HDFS path recursively.

Program:

package hdfs;

import java.io.\*;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.FileSystem;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.fs.RemoteIterator;

import org.apache.hadoop.fs.LocatedFileStatus;

public class FileListing {

public static void main(String[] args) {

if (args.length != 1) {

System.out.println("Pass one argument");

System.exit(1);

}

Path path = new Path(args[0]);

try

{

Configuration conf = new Configuration();

FileSystem fileSystem = FileSystem.get(path.toUri(), conf);

RemoteIterator<LocatedFileStatus>it=fileSystem.listFiles(path,true);

//recursive set to true(second parameter)

While(it.hasNext())

{

System.out.println(it.next().getPath());

}

catch (IOException e)

{

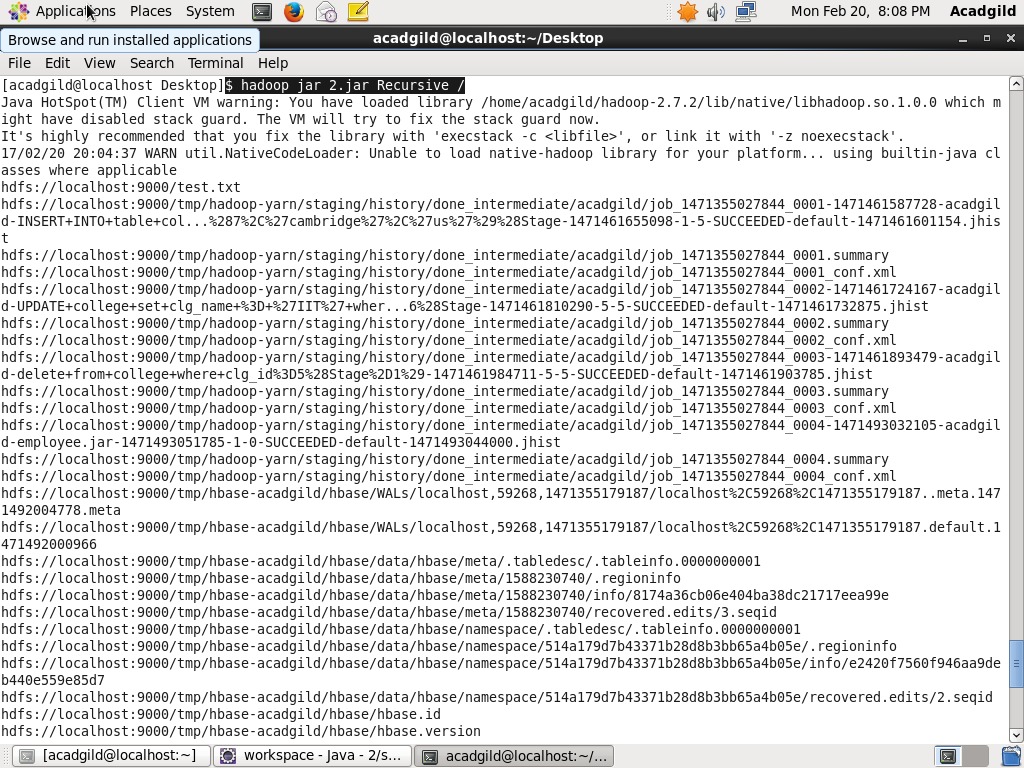
e.printStackTrace();

}

}

}

Output:



Q3- Modify the previous program to take multiple HDFS paths (separated by space) and list all the files and sub-directories in those HDFS paths recursively.

Program:

package hdfs;

import java.io.\*;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.FileSystem;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.fs.RemoteIterator;

import org.apache.hadoop.fs.LocatedFileStatus;

public class FileListing {

public static void main(String[] args) {

if (args.length != 1) {

System.out.println("Pass one argument");

System.exit(1);

}

Int l=args.length;

Path[] path = new Path[l];

try

{­

Configuration conf = new Configuration();

For(int i=0;i<l;i++)

{

Path[i]=new Path(args[i]

FileSystem fileSystem = FileSystem.get(path[i].toUri(), conf);

RemoteIterator<LocatedFileStatus>it=fileSystem.listFiles(path[i],true)

//recursive set to true(second parameter)

System.out.println(“for argument : “+args[i]);

While(it.hasNext())

{

System.out.println(it.next().getPath());

}

}

catch (IOException e)

{

e.printStackTrace();

}

}

}

Output:

