Problem Statement 1: Write a Java program, to take a HDFS Path as input and display all the files and sub-directories in that HDFS pathimport java.io.\*;

import org.apache.hadoop.cimportonf.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();

}

}

}

Problem Statement 2: Modify the previous program to list all the files and sub-directories in the HDFS path recursively.

package hdfs;

import java.io.\*;

import org.apache.hadoop.cimportonf.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);

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

While(it.hasNext())

{

LocatedFileStatus f = it.next();

System.out.println(“Directory” + f.getPath());

}

catch (IOException e)

{

e.printStackTrace();

}

}

}

Problem Statement 3: 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.

import java.io.\*;

import org.apache.hadoop.cimportonf.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);

}

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

{

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

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();

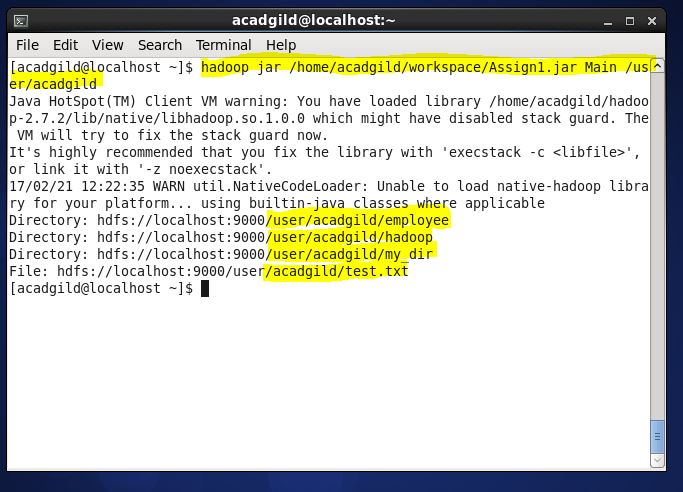
}

}

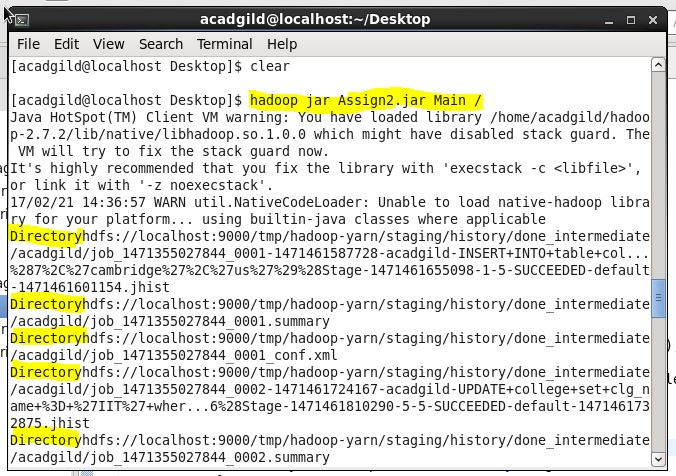
}

}

1. Answer



1. Answer



1. Answer

