

**数学与信息学院学生实验报告**

**实验课程名称：** **教师： \_\_**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验项目名称** | **熟悉常用的HDFS操作** | | | **实验成绩** |  |
| **学生姓名** |  | **学 号** |  | **年级专业班级** |  |
| **小组成员** | **无** | | | **实验日期** | **2019年 10 月** |

# 1. 实验目的和要求

## 1.1 实验目的

① 理解HDFS在Hadoop体系结构中的角色，

② 熟练使用HDFS操作常用的Shell命令，

③ 熟悉HDFS操作常用的Java API

## 1.2 实验软硬件环境

① 操作系统：Linux Ubuntu 16.04

② Java：open-jdk-1.8

③ Apache Hadoop：3.2.1

## 1.3 实验要求

（一）编程实现以下功能，并利用Hadoop提供的Shell命令完成相同任务：

（1） 向HDFS中上传任意文本文件，如果指定的文件在HDFS中已经存在，则由用户来指定是追加到原有文件末尾还是覆盖原有的文件；

（2） 从HDFS中下载指定文件，如果本地文件与要下载的文件名称相同，则自动对下载的文件重命名；

（3） 将HDFS中指定文件的内容输出到终端中；

（4） 显示HDFS中指定的文件的读写权限、大小、创建时间、路径等信息；

（5） 给定HDFS中某一个目录，输出该目录下的所有文件的读写权限、大小、创建时间、路径等信息，如果该文件是目录，则递归输出该目录下所有文件相关信息；

（6） 提供一个HDFS内的文件的路径，对该文件进行创建和删除操作。如果文件所在目录不存在，则自动创建目录；

（7） 提供一个HDFS的目录的路径，对该目录进行创建和删除操作。创建目录时，如果目录文件所在目录不存在，则自动创建相应目录；删除目录时，由用户指定当该目录不为空时是否还删除该目录；

（8） 向HDFS中指定的文件追加内容，由用户指定内容追加到原有文件的开头或结尾；

（9） 删除HDFS中指定的文件；

（10）在HDFS中，将文件从源路径移动到目的路径。

（二）编程实现一个类“MyFSDataInputStream”，该类继承“org.apache.hadoop.fs.FSDataInputStream”，要求如下：实现按行读取HDFS中指定文件的方法“readLine()”，如果读到文件末尾，则返回空，否则返回文件一行的文本。

（三）查看Java帮助手册或其它资料，用“java.net.URL”和“org.apache.hadoop.fs.FsURLStreamHandlerFactory”编程完成输出HDFS中指定文件的文本到终端中。

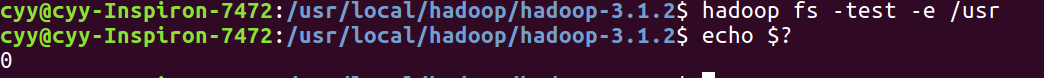
# 2. 实验记录

## 2.1 编程实现下列功能，利用Hadoop提供的Shell命令完成

（1） 向HDFS中上传任意文本文件，如果指定的文件在HDFS中已经存在，则由用户来指定是追加到原有文件末尾还是覆盖原有的文件；

将目录/home/Hadoop下的kc.txt文件传入hdfs的/usr目录下：

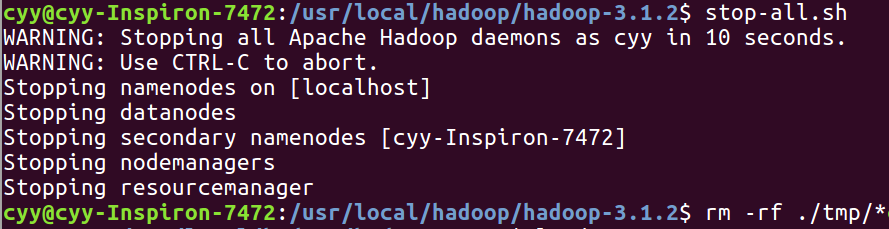




但是一开始我put的时候报以下错误，可能是hdfs的初始化有问题。



解决方法是：



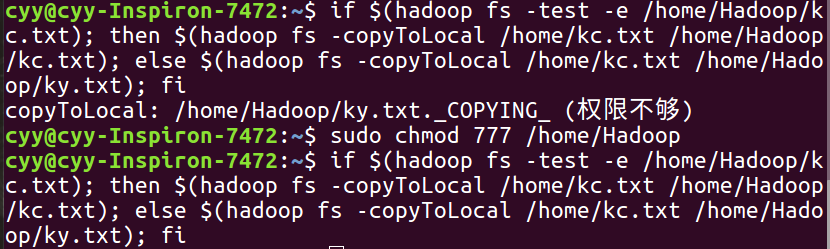
先stop-all.sh；

然后删除tmp下的内容；

再重新对hdfs进行初始化；

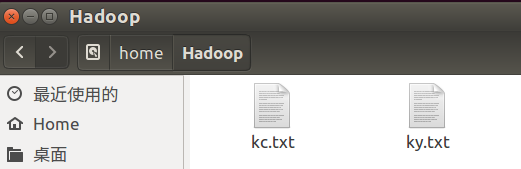
然后重新打开hadoop，即运行该目录下的sbin/start-all.sh,就没有问题了。

（2） 从HDFS中下载指定文件，如果本地文件与要下载的文件名称相同，则自动对下载的文件重命名；



一开始运行的时候提醒/home/Hadoop/ky.txt的权限不够，所以对其赋予权限，就解决了。

运行结果：



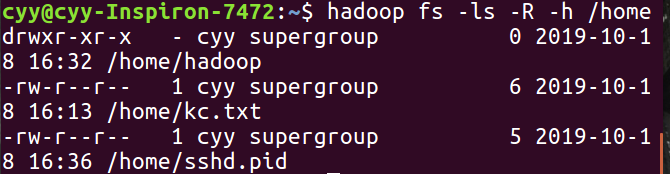
（3） 将HDFS中指定文件的内容输出到终端中；



（4） 显示HDFS中指定的文件的读写权限、大小、创建时间、路径等信息；

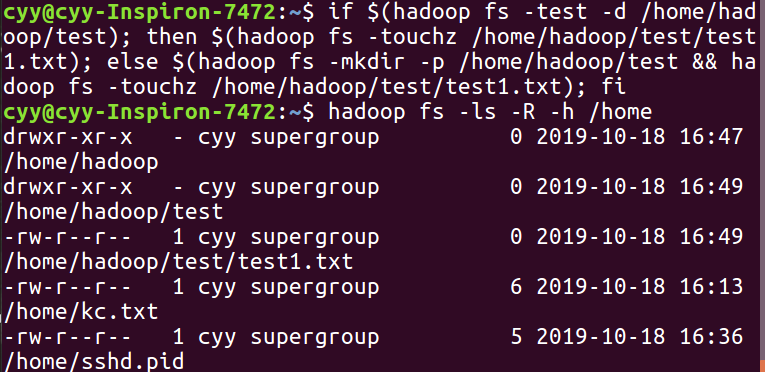


（5） 给定HDFS中某一个目录，输出该目录下的所有文件的读写权限、大小、创建时间、路径等信息，如果该文件是目录，则递归输出该目录下所有文件相关信息；



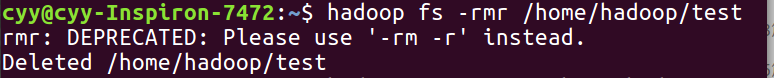
可以看到/home里的文件都被递归显示了。

（6） 提供一个HDFS内的文件的路径，对该文件进行创建和删除操作。如果文件所在目录不存在，则自动创建目录；



一开始是没有/home/hadoop/test这个目录的，命令运行之后把文件夹内容递归显示一下，就能发现已经加上了这个目录，并在这个目录下新建了test1.txt文件。

（7） 提供一个HDFS的目录的路径，对该目录进行创建和删除操作。创建目录时，如果目录文件所在目录不存在，则自动创建相应目录；删除目录时，由用户指定当该目录不为空时是否还删除该目录；



（8） 向HDFS中指定的文件追加内容，由用户指定内容追加到原有文件的开头或结尾；

（9） 删除HDFS中指定的文件；



（10）在HDFS中，将文件从源路径移动到目的路径。



## 2.2 编程实现一个类MyFSDataInputStream

package test1;

import java.io.\*;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.FSDataInputStream;

import org.apache.hadoop.fs.FileSystem;

import org.apache.hadoop.fs.Path;

public class MyFSDataInputStream extends FSDataInputStream{

public MyFSDataInputStream(InputStream in) {

super(in);

}

public static String readline(Configuration conf,String remo){

try(FileSystem fs=FileSystem.get(conf)){

Path remotePath=new Path(remo);

FSDataInputStream in=fs.open(remotePath);

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

String line=null;

if((line=d.readLine())!=null) {

d.close();

in.close();

fs.close();

return line;

}

return null;

}

catch(IOException e){

e.printStackTrace();

return null;

}

}

}

package demo2;

import java.io.IOException;

import org.apache.hadoop.conf.Configuration;

import test1.MyFSDataInputStream;

public class Main {

public static void main(String[] args) throws IOException {

// TODO Auto-generated method stub

Configuration conf=new Configuration();

conf.set("fs.dafault.name", "hdfs://localhost:9000");

String remo="/home/hadoop/kc.txt";

System.out.println("读取文件"+remo);

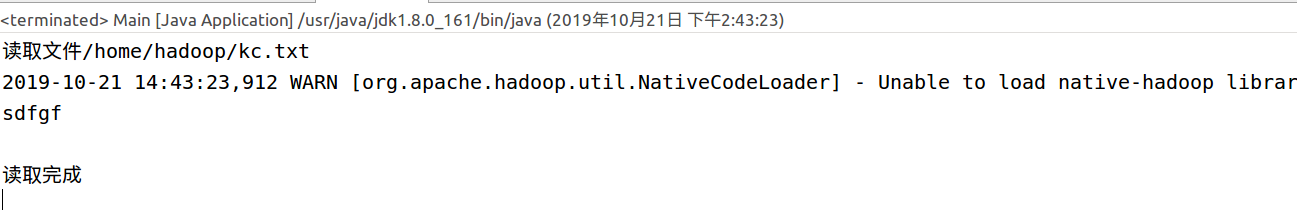
System.out.println(MyFSDataInputStream.readline(conf, remo));

System.out.println("\n读取完成");

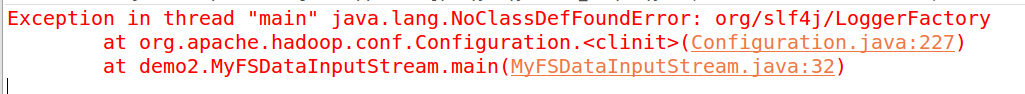
}

}

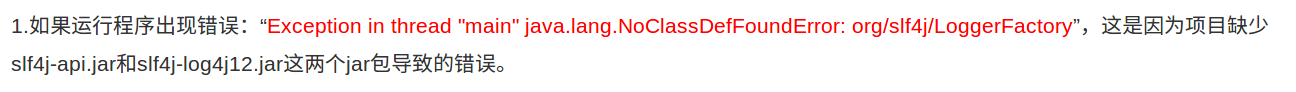
运行结果：



最初开始运行出现了以下问题：



查了下网上的资料，说的是：



然后我就把这两个jar包下载了并且导入，但还是抛出一样的错误，不懂是不行还是导入错误。

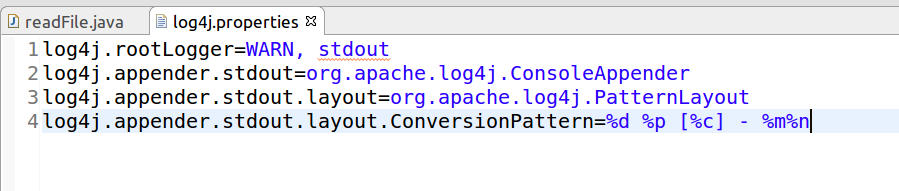
问了问同学，说是要把全部的包都导进这个java project里面，也就是Hadoop目录下的/share/common和/share/common/bin和/share/hdfs以及/share/hdfs/bin下的所有包。

导入后重新运行，还是出现了以下错误：

log4j:WARN No appenders could be found for logger (org.apache.hadoop.fs.File)

log4j:WARN Please intialize the log4j systemproperly.

查了网上的资料，提示说是缺少log4j的日志文件。于是在该项目下新建文件log4j.properties，然后加入以下内容：



就能成功运行了，运行之前要确保Hadoop已经开启。

## 2.3 编程完成输出HDFS中指定文件的文本到终端中

package test2;

import java.io.IOException;

import java.io.InputStream;

import java.net.URL;

import org.apache.hadoop.fs.FsUrlStreamHandlerFactory;

import org.apache.hadoop.io.IOUtils;

public class readFile {

static {;

URL.setURLStreamHandlerFactory(new FsUrlStreamHandlerFactory());

}

public static void cat(String remo) {

try(InputStream in=new URL("hdfs","localhost",9000,remo).openStream()){

IOUtils.copyBytes(in, System.out, 4096,false);

IOUtils.closeStream(in);

}catch(IOException e) {

e.printStackTrace();

}

}

public static void main(String[] args) {

// TODO Auto-generated method stub

String remo="/home/hadoop/kc.txt";

try{

System.out.println("读取文件："+remo);

readFile.cat(remo);

System.out.println("\n读取完成");

}catch(Exception e){

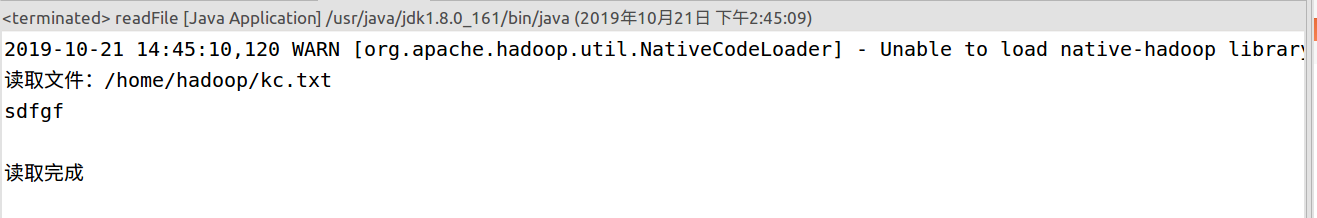
e.printStackTrace();

}

}

}

运行结果：



实验结果与设想结果一致。

# 3. 实验总结

这次实验还是一如既往的艰难，遇到了不少奇奇怪怪的问题。

一开始是hadoop fs指令报错，后来才知道是初始化多次导致的错误，需要删除节点，重新初始化。

再后来就是java程序缺jar包以及log4j日志导致的一系列错误，花了不少时间，但最终还是解决了。

除开上面讲的问题，还有一个D盘挂载的问题。在Ubuntu系统上，我的DATA盘和OS盘偶尔能实现自动挂载，偶尔则需要手动挂载。手动挂载还只能挂DATA，OS挂不动。并且在强行对D盘进行挂载后，往里面放文件后，用Windows打开查看就没了，非常奇怪。这是我暂时还没有解决的问题。