实验一 大数据系统基本实验

第一部分 熟悉常用的Linux操作和Hadoop操作

一、 实验目的

Hadoop运行在Linux系统上，因此需要学习实践一些常用的Linux命令。本实验旨在熟悉常用的Linux操作和Hadoop操作，为顺利开展后续其它实验奠定基础。

二、 实验平台

1）操作系统：Linux（实验室当前版本为Ubuntu17.04）；

2）Hadoop版本：2.9.0；

3）JDK版本：1.8；

4）Java IDE：Eclipse 3.8。

三、 实验内容

1）cd命令：切换目录

(1) 切换到目录/usr/local。

(2) 切换到当前目录的上一级目录

(3) 切换到当前登录Linux系统的用户自己的主文件夹

2）ls命令：查看文件与目录

查看目录/usr下的所有文件和目录

3）mkdir命令：新建目录

(1) 进入/tmp目录，创建一个名为a的目录，并查看/tmp目录下已经存在哪些目录。

(2) 进入/tmp目录，创建目录a1/a2/a3/a4。

4）rmdir命令：删除空的目录

(1) 将上面创建的目录a(在/tmp目录下面)删除。

(2) 删除上面创建的目录a1/a2/a3/a4(在/tmp目录下面)，然后查看/tmp目录下面存在哪些目录。

5）cp命令：复制文件或目录

(1) 将当前用户的主文件夹下的文件.bashrc复制到目录“/usr”下，并重命名为bashrc1

(2) 在目录“/tmp”下新建目录test，再把这个目录复制到“/usr”目录下

6）mv命令：移动文件与目录，或更名

(1) 将“/usr”目录下的文件bashrc1移动到“/usr/test”目录下

(2) 将“/usr”目录下的test目录重命名为test2

7）rm命令：移除文件或目录

(1) 将“/usr/test2”目录下的bashrc1文件删除

(2) 将“/usr”目录下的test2目录删除

8）cat命令：查看文件内容

查看当前用户主文件夹下的.bashrc文件内容

9）tac 命令：反向查看文件内容

反向查看当前用户主文件夹下的.bashrc 文件的内容

10）more 命令：一页一页翻动查看

翻页查看当前用户主文件夹下的.bashrc 文件的内容

11）head 命令：取出前面几行

(1) 查看当前用户主文件夹下.bashrc 文件内容前 20 行

(2) 查看当前用户主文件夹下.bashrc 文件内容，后面 50 行不显示，只显示前面几行

12）tail 命令：取出后面几行

(1) 查看当前用户主文件夹下.bashrc 文件内容最后 20 行

(2) 查看当前用户主文件夹下.bashrc 文件内容，并且只列出 50 行以后的数据

13）touch 命令：修改文件时间或创建新文件

(1) 在“/tmp”目录下创建一个空文件 hello，并查看文件时间

(2) 修改 hello 文件，将文件时间整为 5 天前

14）chown 命令：修改文件所有者权限

将 hello 文件所有者改为 root 帐号，并查看属性

15）find 命令：文件查找

找出主文件夹下文件名为.bashrc 的文件

16）tar 命令：压缩命令

(1) 在根目录“/”下新建文件夹 test，然后在根目录“/”下打包成 test.tar.gz

(2) 把上面的 test.tar.gz 压缩包，解压缩到“/tmp”目录

17）grep 命令：查找字符串

从“～/.bashrc”文件中查找字符串'examples'

18）使用 hadoop 用户登录 Linux 系统，启动 Hadoop（Hadoop 的安装目录为“/usr/local/hadoop”），为 hadoop 用户在 HDFS 中创建用户目录“/user/hadoop”

19）接着在 HDFS 的目录“/user/hadoop”下，创建 test 文件夹，并查看文件列表

20）将 Linux 系统本地的“～/.bashrc”文件上传到 HDFS 的 test 文件夹中，并查看 test

21）将 HDFS 文件夹 test 复制到 Linux 系统本地文件系统的“/usr/local/hadoop”目录下

第二部分　熟悉常用的 HDFS 操作

一、 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。

二、 实验平台

1）操作系统：Linux（Ubuntu17.04）；

2）Hadoop 版本：2.9.0；

3）JDK 版本：1.8；

4）Java IDE：Eclipse 3.8。

三、 实验内容

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，递归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;

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，当该目录为空时删除，当该目录不为空时不删除该目录；

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。

第三部分 熟悉常用的 HBase 操作

一、 实验目的

1）理解 HBase 在 Hadoop 体系结构中的角色。

2）熟练使用 HBase 操作常用的 shell 命令。

3）熟悉 HBase 操作常用的 Java API。

二、 实验平台

1）操作系统：Linux（Ubuntu17.04）；

2）Hadoop 版本：2.9.0；

3）HBase 版本：1.2.6；

4）JDK 版本：1.8；

5）Java IDE：Eclipse 3.8。

三、 实验内容

1）编程实现以下指定功能，并用 Hadoop 提供的 HBase Shell 命令完成相同任务：

(1) 列出 HBase 所有的表的相关信息，例如表名；

(2) 在终端打印出指定的表的所有记录数据；

(3) 向已经创建好的表添加和删除指定的列族或列；

(4) 清空指定的表的所有记录数据；

(5) 统计表的行数。

2）HBase 数据库操作

现有以下关系型数据库中的表和数据，要求将其转换为适合于 HBase 存储的表并插入数据：

学生表（Student）

学号（S_No）	姓名（S_Name）	性别（S_Sex）	年龄（S_Age）
2015001	Zhangsan	male	23
2015003	Mary	female	22
2015003	Lisi	male	24

课程表（Course）

课程号（C_No）	课程名（C_Name）	学分（C_Credit）
123001	Math	2.0
123002	Computer Science	5.0
123003	English	3.0

选课表（SC）

学号（SC_Sno）	课程号（SC_Cno）	成绩（SC_Score）
2015001	123001	86
2015001	123003	69
2015002	123002	77
2015002	123003	99
2015003	123001	98
2015003	123002	95

3）请编程实现以下功能：

createTable(String tableName, String[] fields)

创建表，参数 tableName 为表的名称，字符串数组 fields 为存储记录各个字段名称的数组。要求当 HBase 已经存在名为 tableName 的表的时候，先删除原有的表，然后再创建新的表。

addRecord(String tableName, String row, String[] fields, String[] values)

向表 tableName、行 row（用 S_Name 表示）和字符串数组 fields 指定的单元格中添加对应的数据 values。其中，fields 中每个元素如果对应的列族下还有相应的列限定符的话，用“columnFamily:column”表示。例如，同时向“Math”、“Computer Science”、“English”三列添加成绩时，

字符串数组 fields 为{“Score:Math”, ”Score:Computer Science”, ”Score:English”}，数组 values 存储这三门课的成绩。

scanColumn(String tableName, String column)

浏览表 tableName 某一列的数据，如果某一行记录中该列数据不存在，则返回 null。要求当参数 column 为某一列族名称时，如果底下有若干个列限定符，则要列出每个列限定符代表的列的数据；当参数 column 为某一列具体名称（例如“Score:Math”）时，只需要列出该列的数据。

modifyData(String tableName, String row, String column)

修改表 tableName，行 row（可以用学生姓名 S_Name 表示），列 column 指定的单元格的数据。

deleteRow(String tableName, String row)

删除表 tableName 中 row 指定的行的记录。

第四部分 NoSQL 和关系数据库的比较

一、 实验目的

1）理解 4 种数据库（MySQL、HBase、Redis 和 MongDB）的概念及不同点。

2）熟练使用 4 种数据库操作常用的 shell 命令。

3）熟悉 4 种数据库操作常用的 Java API。

二、 实验平台

1）操作系统：Linux（Ubuntu17.04）；

2）Hadoop 版本：2.9.0；

3）MySQL 版本：5.6；

4）HBase 版本：1.2.6；

5）Redis 版本：4.0.8；

6）MongDB 版本：3.2.19；

7）JDK 版本：1.8；

8）Java IDE：Eclipse 3.8。

三、 实验内容

1. MySQL 数据库操作

学生表 Student

Name	English	Math	Computer
Zhangsan	69	86	77
Lisi	55	100	88

1）根据上面给出的 Student 表，在 MySQL 数据库中完成如下操作：

(1) 在 MySQL 中创建 Student 表，并录入数据；

(2) 用 SQL 语句输出 Student 表中的所有记录；

(3) 查询 zhangsan 的 Computer 成绩；

(4) 修改 lisi 的 Math 成绩，改为 95。

2）根据上面已经设计出的 Student 表，使用 MySQL 的 JAVA 客户端编程实现以下操作：

(1) 向 Student 表中添加如下所示的一条记录：

Scofield	45	89	100

(2) 获取 scofield 的 English 成绩信息。

2. HBase 数据库操作

学生表 Student

Name	score		
	English	Math	Computer
Zhangsan	69	86	77
Lisi	55	100	88

1）根据上面给出的学生表 Student 的信息，执行如下操作：

(1) 用 Hbase Shell 命令创建学生表 Student；

(2) 用 scan 命令浏览 Student 表的相关信息；

(3) 查询 zhangsan 的 Computer 成绩；

(4) 修改 lisi 的 Math 成绩，改为 95。

2）根据上面已经设计出的 Student 表，用 HBase API 编程实现以下操作：

(1) 添加数据：English:45 Math:89 Computer:100

scofield	45	89	100

(2) 获取 scofield 的 English 成绩信息。

3. Redis 数据库操作

Student 键值对如下：

```
                    zhangsan: {
                                English: 69
                                Math: 86
                                Computer: 77
                    }
                    lisi: {
                                English: 55
                                Math: 100
                                Computer: 88
}
```

1）根据上面给出的键值对，完成如下操作：

(1) 用 Redis 的哈希结构设计出学生表 Student（键值可以用 student.zhangsan 和 student.lisi 来表示两个键值属于同一个表）；

(2) 用 hgetall 命令分别输出 zhangsan 和 lisi 的成绩信息；

(3) 用 hget 命令查询 zhangsan 的 Computer 成绩；

(4) 修改 lisi 的 Math 成绩，改为 95。

2）根据上面已经设计出的学生表 Student，用 Redis 的 JAVA 客户端编程(jedis)，实现如下操作：

(1) 添加数据：English:45 Math:89 Computer:100

该数据对应的键值对形式如下：

```
scofield: {
                        English: 45
                        Math: 89
                        Computer: 100
}
```

(2) 获取 scofield 的 English 成绩信息。

4. MongDB 数据库操作

Student 文档如下:

```
{
    “name”: “zhangsan”,
    “score”: {
        “English”: 69,
        “Math”: 86,
        “Computer”: 77
    }
}
{
    “name”: “lisi”,
    “score”: {
        “English”: 55,
        “Math”: 100,
        “Computer”: 88
    }
}
```

1）根据上面给出的文档，完成如下操作：

(1) 用 MongoDB Shell 设计出 student 集合；

(2) 用 find()方法输出两个学生的信息；

(3) 用 find()方法查询 zhangsan 的所有成绩(只显示 score 列)；

(4) 修改 lisi 的 Math 成绩，改为 95。

2）根据上面已经设计出的 Student 集合，用 MongoDB 的 Java 客户端编程，实现如下操作：

(1) 添加数据：English:45 Math:89 Computer:100

与上述数据对应的文档形式如下：

```
{
    “name”: “scofield”,
    “score”: {
        “English”: 45,
        “Math”: 89,
        “Computer”: 100
```

```
                }
}
```

(2) 获取 scofield 的所有成绩信息(只显示 score 列)。

第五部分 MapReduce 初级编程

一、 实验目的

1）通过实验掌握基本的 MapReduce 编程方法。

2）掌握用 MapReduce 解决一些常见的数据处理问题，包括数据去重、数据排序和数据挖掘等。

二、 实验平台

1）操作系统：Linux（Ubuntu17.04）；

2）Hadoop 版本：2.9.0；

三、 实验内容

1）编程实现文件的合并和去重

对于两个输入文件，即文件 A 和文件 B，请编写 MapReduce 程序，对两个文件进行合并，并剔除其中重复的内容，得到一个新的输出文件 C。下面是输入文件和输出文件的一个样例供参考。

输入文件 A 的样例如下：

```
20170101    x
20170102    y
20170103    x
20170104    y
20170105    z
20170106    x
```

输入文件 B 的样例如下：

```
20170101    y
20170102    y
20170103    x
20170104    z
20170105    y
```

根据输入文件 A 和 B 合并得到的输出文件 C 的样例如下：

```
20170101    x
20170101    y
20170102    y
20170103    x
20170104    y
20170104    z
20170105    y
```

```
20170105    z
20170106    x
```

2）编程实现对输入文件的排序

现在有多个输入文件，每个文件中的每行内容均为一个整数。要求读取所有文件中的整数，进行升序排序后，输出到一个新的文件中，输出的数据格式为每行两个整数，第一个数字为第二个整数的排序位次，第二个整数为原待排列的整数。下面是输入文件和输出文件的一个样例供参考。

输入文件 1 的样例如下：

```
33
37
12
40
```

输入文件 2 的样例如下：

```
4
16
39
5
```

输入文件 3 的样例如下：

```
1
45
25
```

根据输入文件 1、2 和 3 得到的输出文件如下：

```
1 1
2 4
3 5
4 12
5 16
6 25
7 33
8 37
9 39
10 40
11 45
```

3）对指定的表格进行信息挖掘

下面给出一个 child-parent 的表格，要求挖掘其中的父子辈关系，给出祖孙辈关系的表格。

输入文件内容如下：

```
child    parent
Steven   Lucy
Steven   Jack
Jone     Lucy
Jone     Jack
```

Lucy	Mary
Lucy	Frank
Jack	Alice
Jack	Jesse
David	Alice
David	Jesse
Philip	David
Philip	Alma
Mark	David
Mark	Alma

输出文件内容如下：

grandchild	grandparent
Steven	Alice
Steven	Jesse
Jone	Alice
Jone	Jesse
Steven	Mary
Steven	Frank
Jone	Mary
Jone	Frank
Philip	Alice
Philip	Jesse
Mark	Alice
Mark	Jesse

第六部分 实验报告

计算机科学与技术学院 大数据管理与分析 课程实验报告

实验题目：		学号：201500000000
日期：2018.3.20	班级：2015 级 1 班/菁英班	姓名：张三
Email：zhangsan@qq.com		
实验目的：		
实验软件和硬件环境：		

实验原理和方法：
实验步骤：（不要求罗列完整源代码）
结论分析与体会：
就实验过程中遇到和出现的问题，你是如何解决和处理的，自拟 1－3 道问答题：