**大数据学院**

**2018年11月3日**



**[第一单元 大数据概论和分布式介绍]**

hadoop标准化课件设计

**《Hadoop》**

**第十单元 mapreduce开发**

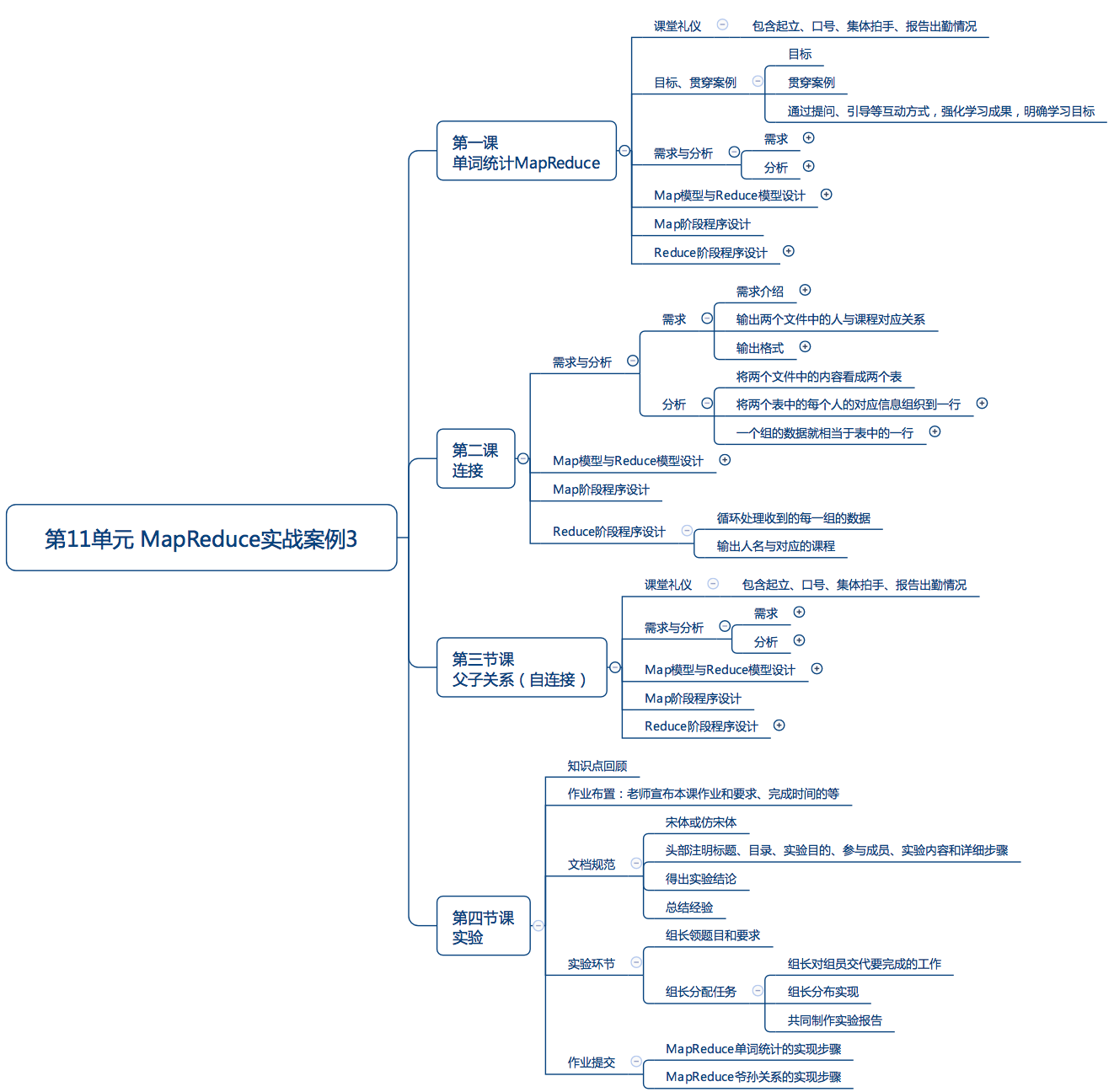
1. **本单元教学目标**
2. 知识目标：

* 强化从需求到模型到设计程序的能力

1. 能力目标：

* 能够自行根据需求设计模型
* 能够自行根据模型设计MapReduce程序

1. **课时分配**

****

1. **第一课时：单词统计**

本课时通过对大数据和分布式的宏观介绍，使学生初步建立大数据知识体系，使得学生能为学习以后的课程打下基础。

1. 知识目标：

**·**  强化MapReduce模型设计；

**·** 掌握如何处理多个文件；

**·**  掌握如何区分内容来源的文件；

1. 能力目标：

* 能够自行设计模型
* 能够处理并区分内容来源的文件

1. **第二课时：降雨量程序分区**
2. 知识目标：

* 了解关系模型设计
* 掌握具有关系的MapReduce程序的设计方法

1. 能力目标：

**·** 能够根据需求自行设计关系模型并设计根据模型开发程序

1. **第三课时：数据压缩与运行程序**
2. 知识目标：

* 强化关系模型设计

1. 能力目标：

* 能够根据根据需求设计关系模型并设计程序

1. **第四课时：实验**
2. 知识目标：

* 强化模型设计与程序设计

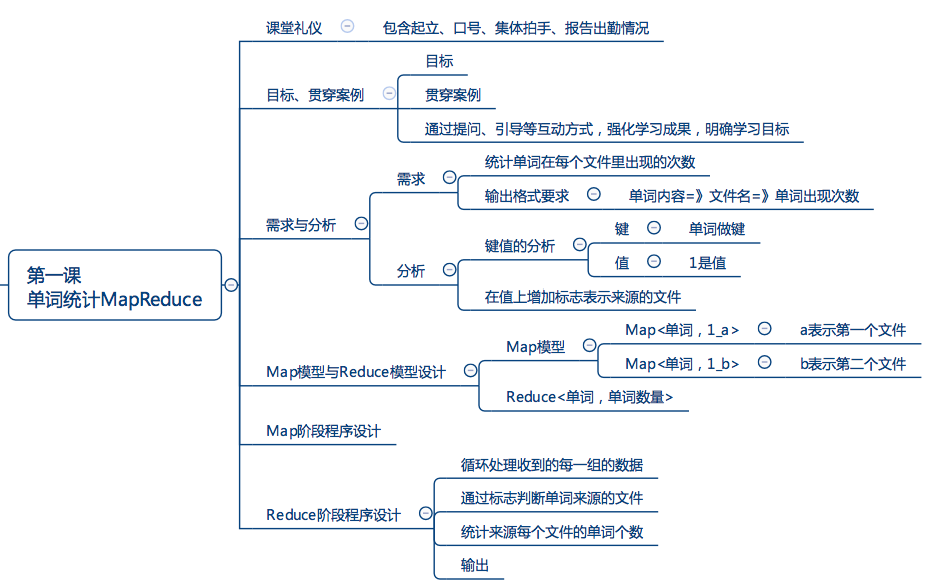
1. 能力目标：

* 能够根据需求设计模型并设计程序

**三、本单元概念图**

**四、教学过程脚本设计——第一课时**

1. **脚本设计思路**

****

1. **脚本切片设计思路**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **内容** | **小节目标** | **设计亮点** | **时间** |
| **1** | **课堂礼仪** | 集中学生的注意力，喊出士气，以饱满的精神状态投入本节课的学习 | 整齐、声音洪亮、铿锵有力，提升精气神！ |  |
| **2** | **为什么学单词统计这个例子** | 树立学习目标：多文件读取，如何在程序中区分内容来源的文件 |  |  |
| **3** | **单词统计程序需求** | 清楚具体项目需求的细节 |  |  |
| **4** | **涉及到的技术点** | 掌握获取文件名的方法 |  |  |
| **5** | **模型设计** | 通过需求构建Map与Reduce模型 |  |  |
| **6** | **流程梳理** | 强化流程 |  |  |
| **7** | **Map程序开发** | 学生根据模型设计自行开发Map程序 |  |  |
| **8** | **Reduce程序开发** | 学生根据模型设计自行开发Reduce程序 |  |  |
| **9** | **知识点总结** | 如何处理更多的文件，如何将处理的内容与来源文件关联 |  |  |

1. **脚本切片一：[课堂礼仪](file:///D:\\llq\\八维\\工作\\教学标准化计划\\切片(总共14片)\\切片1.mp4)**

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **课堂礼仪** | 使学生注意力集中并进入上课状态 |  |  |
| **讲师：**上课！①  **班长：**起立！  **讲师：**同学们好！  **学生：**老师好！  **班长：**我们的信念是：  **全体：**知识^很简单，学习^很快乐，潜力^无极限！  **班长：**我们的口号是：  **全体：**辉煌八维大数据，引领时代高科技！人工智能显神奇，高薪就业创佳绩！  **讲师：**请坐！② | 1. 进入教室前整理着装，提前30秒面带自信的笑容的走进教室。，给学生打招呼。当上课铃声响起，喊出“上课”，声音高于授课声音。 2. 细节要求：学生发型整齐、脸部整洁、面带微笑或者礼仪笑（八颗牙齿），严禁大笑，举止合理，避免指手画脚。   IMG_256IMG_256   1. 学生礼仪内容：学习信念，学院口号，班级口号。   讲师授课过程中要面带微笑。   1. 插入合理手势，避免课堂枯燥。   ①动作表情：上课起立手势  ②动作表情：全体请坐手势 |  |

1. **脚本切片二**：**单词统计程序需求**

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **降雨量需求** | 了解需求 | 正确理解需求 |  |
| **老师：**大家先看看这堂课例子的需求  两个文件a.txt与b.txt,每个文件里有不同的单词内容。统计出每个单词在各自文件里出现的次数。  输出的格式如下：  love a.txt=>1  love a.txt=>2  **老师：**字母love在a.txt里出现了1此，在b.txt里出现了2次 | **a.txt**  I like MapReduce  I like Hadoop  I love you  **b.txt**  I love dxx  I love me  I like tennis |  |

1. **脚本切片三：**知识点获取文件名

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **获取文件名** | 在Map程序中获取文件名的方法 | 了解并掌握Map程序中获取文件名的方法 |  |
| **老师**：输出中用到了文件名，在MapReduce程序中获取文件名的方式：  **老师**：获取文件名的方法：【ppt】 | FileSplit fs = (FileSplit) context.getInputSplit();  String fileName = fs.getPath().getName(); |  |

1. **脚本切片四：**构建模型

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **构建模型** | 强化模型设计 | 掌握强化模型设计 |  |
| **老师:**在开发之前要先构建map与reduce的模型。有了模型才能根绝模型开发  **老师：**Map的模型用什么作为键，用什么作为值那？  **老师**：咱们设计的map与reduce的模型如下，其中reduce的模型是根据输出的格式反推得到的。   * map<单词，文件名> * reducer<单词，文件名=>单词数量> | **a.txt**  I like MapReduce  I like Hadoop  I love you  **b.txt**  I love dxx  I love me  I like tennis  两个文件a.txt与b.txt,每个文件里有不同的单词内容。统计出每个单词在各自文件里出现的次数。  输出的格式如下：  love a.txt=>1  love a.txt=>2 |  |

1. **脚本切片四：**Map程序开发

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 编写自定义键值类型程序 | 老师快速过一下Map程序开发 | 同学课后自行开发 |  |
| 老师：现在咱们可以根据模型开发Map程序。  因为用单词作为map模型的键，所以会将键（单词）一样的分到一组。 | protected void map(LongWritable key, Text value, Mapper<LongWritable, Text, Text, Text>.Context context)  throws IOException, InterruptedException {  **// 获取文件名**  **FileSplit fs = (FileSplit) context.getInputSplit();**  **String fileName = fs.getPath().getName();**  **// 将当前处理的行切割成单词数组**  String inwords = value.toString();  String[] words = inwords.split(" ");  **// 循环单词数据，每次循环输出的模型为Map<单词，文件名>**  for (String w : words) {  context.write(new Text(w), new Text(fileName));  }  }  } |  |

1. **脚本切片五：**Reduce程序开发

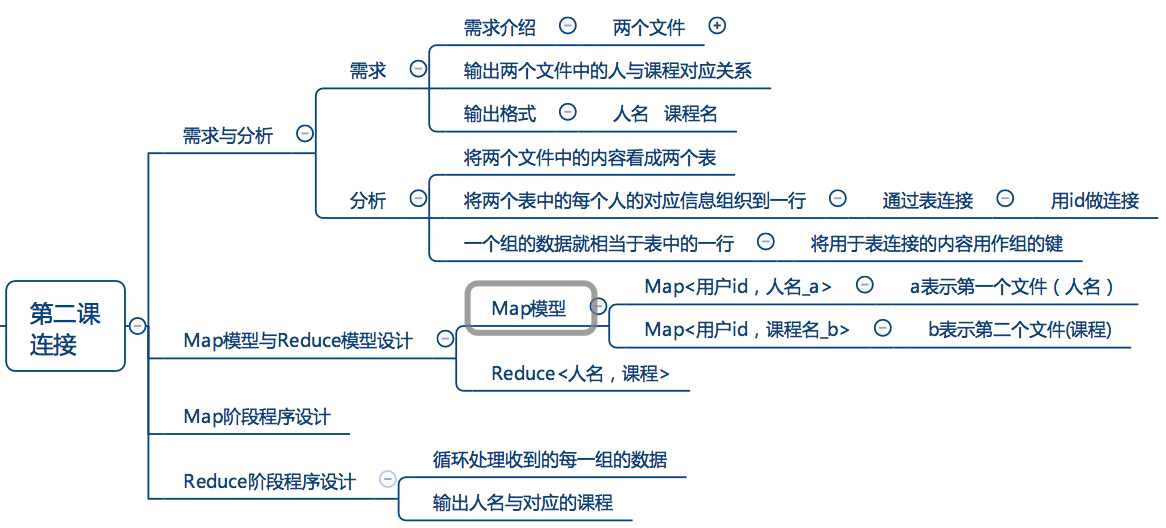
|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **Map程序开发** | 老师快速过一下Reducer程序开发 | 同学课后自行开发 |  |
| **老师**：现在咱们可以根据模型开发Reduce程序。 | public class TongJiReducer extends Reducer<Text, Text, Text, Text> {  @Override  protected void reduce(Text key, Iterable<Text> values, Reducer<Text, Text, Text, Text>.Context context)  throws IOException, InterruptedException {  Iterator ita = values.iterator();  Integer a = 0;**// 计数变量，用于统计a.txt出现的次数**  Integer b = 0;**// 计数变量，用于统计b.txt出现的次**数  // 循环当前组的值（值是文件名）  while (ita.hasNext()) {  String filename = (String) ita.next();  if (filename.equals("a.txt")) {  a++;  }  if (filename.equals("b.txt")) {  b++;  }  }  **// 分别输出单词在各自文件里出现的次数**  **context.write(key, new Text("a.txt=>" + a));**  **context.write(key, new Text("b.txt=>" + b));**  }  } |  |

1. **脚本切片五：**知识点总结

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **知识点总结** | **知识点总结** |  |  |
| 老师：今天学些的知识点总结如下，需要大家完全掌握：   1. 如何处理更多的文件 2. 如何将处理的内容与来源文件关联 |  |  |

**五、教学过程脚本设计——第二课时**

1. **脚本设计思路**

****

1. **脚本切片设计思路**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **内容** | **小节目标** | **设计亮点** | **时间** |
| **1** | **课程关系程序需求** | 了解课程关系程序需求 |  |  |
| **2** | **模型设计分析** | 掌握关系模型设计窍门 | 对比关系数据库的关联关系讲解 |  |
| **3** | **模型设计** | 正确设计模型 |  |  |
| **4** | **Map阶段程序设计** | 学生自行开发 |  |  |
| **5** | **Reduce阶段程序设计** | 学生自行开发 |  |  |
| **6** | **知识点总结** | 关系数据模型设计窍门 |  |  |

1. **脚本切片三：**课程关系程序需求

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **课程关系程序需求** | **了解需求** | 准确了解需求 |  |
| 老师：  课程关系程序的需求：有两个文件，一个是学生文件stu.txt,一个是课程文件course.txt.两个文件内容是：  stu.txt:两列（id,x姓名）  course.txt:两列（课程，id）  要求输出如下格式：  zhangsan hadoop  zhangsan web  zhangsan java  lisi c  lisi ui  lisi php  要求输出人与课程的对应关系，无对应关系的不输出。（如：id为004的人无对应课程，id为005的课程无对应的人，所有不需要输出） | **stu.txt**  001 zhangsan  002 lisi  003 wangwu  004 zhaoliu  **course.txt**  java 001  web 001  hadoop 001  php 002  ui 002  c 002  python 005 |  |

1. **脚本切片三：**模型设计分析

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 分区 | 设计Map与Reduce模型分析 | 强化模型设计分析习惯 |  |
| **老师**：可以通过以下3点推导模型设计   * 1：将两个文件看成两个各有两列的表 * 2：第一个表的列：id，name;第二个表的列：课程，id * 3：多个表开发MapReduce:找出可以将两个表连接起来的字段。这个字段就作为Map模型的键 |  |  |

1. **脚本切片三：**构建模型

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 什么是分区 | **分区概述** | 理解分区与分区的定义 |  |
| 老师：分区的概述   * map<id，姓名\_a或课程\_b>   因为属于同一个id的数据来自两个的文件（姓名，课程。所以用a,与b标志区分来源）   * reducer<姓名，课程> | **stu.txt**  001 zhangsan  002 lisi  003 wangwu  004 zhaoliu  001 zhangsan\_a  001 java\_b  001 web\_b  001 hadoop\_b  **course.txt**  java 001  web 001  hadoop 001  php 002  ui 002  c 002  python 005 |  |

1. **脚本切片三：**Map程序

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 分区 | 分区注意事项 | 自行开发实现数据分区 |  |
| 老师：大家课后自行开发，现在先简单过一下Map程序 | **public** **class** CourseMapper **extends** Mapper<LongWritable, Text, Text, Text> {  // 1:key 读入的当前行的位置 2:value 读入的当前行的文本  @Override  **protected** **void** map(LongWritable key, Text value, Mapper<LongWritable, Text, Text, Text>.Context context)  **throws** IOException, InterruptedException {  String v = value.toString();  String[] a = v.split("\t");  // 获取当前行对应的文件文件名  FileSplit fs = (FileSplit) context.getInputSplit();//  String fileName = fs.getPath().getName();// stu.txt 或者 cource.txt  **if** (fileName.equals("stu.txt")) {  context.write(**new** Text(a[0]), **new** Text(a[1] + "\_a"));  }  **if** (fileName.equals("course.txt")) {  context.write(**new** Text(a[1]), **new** Text(a[0] + "\_b"));  }  }  } |  |

1. **脚本切片三：**Reduce程序

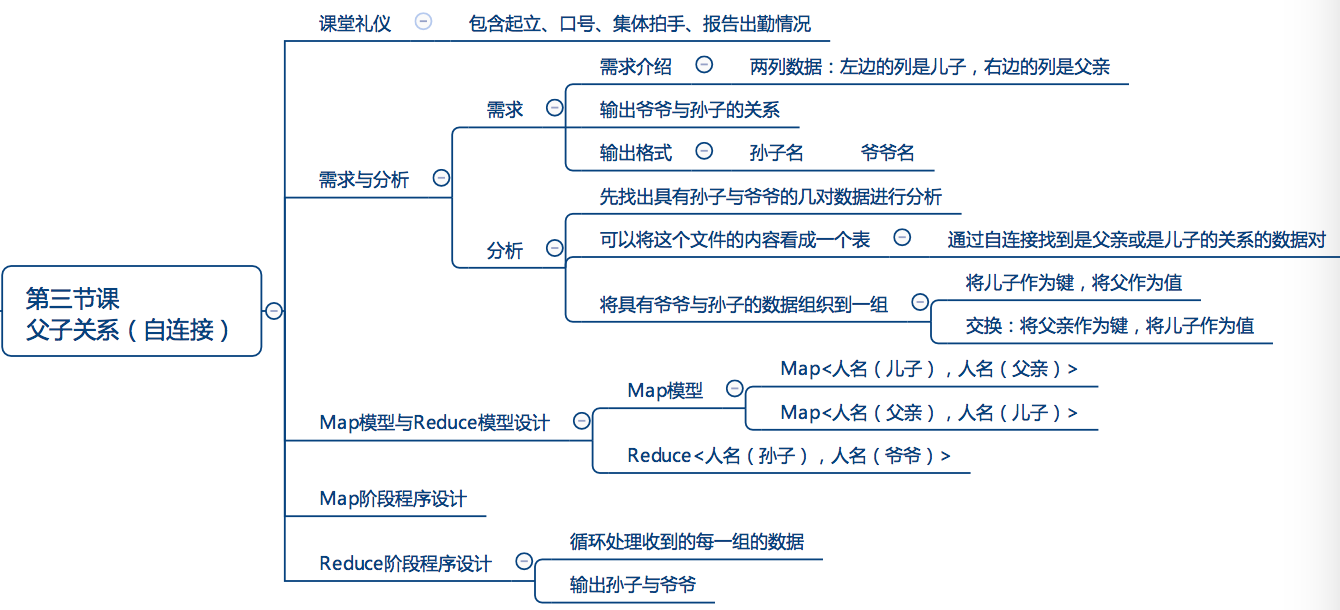
|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 分区 | **分区必须继承的类** | 掌握分区必须继承的类 |  |
| 老师：大家课后自行开发，现在先简单过一下Reduce程序 | public class CourseReduce extends Reducer<Text, Text, Text, Text> {  protected void reduce(Text key, Iterable<Text> value, Reducer<Text, Text, Text, Text>.Context context)  throws IOException, InterruptedException {  Iterator ita = value.iterator();  String stuname="";  java.util.List<String> l = new java.util.ArrayList<String>();  //判断当前集合里有没有内容  while(ita.hasNext()){  //next取出当前集合指针对应的内容，并将指针的位置下移一个位置。  Text t = (Text)ita.next();  String c = t.toString();    if(c.endsWith("\_a")){//等于a表明这个c是人，来自于stu.txt  //1：indexOf("\_"): 返回指定“\_”在字符串c中的位置。  stuname=c.substring(0, c.indexOf("\_"));//substring: 截取从开始位置0到结束位置的字符串  }  if(c.endsWith("\_b")){//等于b表明这个c是课程，可能有很多课程，所以放到list集合里  l.add(c.substring(0, c.indexOf("\_")));  }    }  //循环list，将姓名为键，课程为值，放到reduce里的输出集合里  if((stuname!=null&&!stuname.equals(""))&&l.size()>0){  for(String s:l){  context.write(new Text(stuname), new Text(s));  }  }    } |  |

1. **脚本切片三：**知识点回顾

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 知识点回顾 | 知识点回顾 | 自行开发实现数据分区 |  |
| 老师：多个文件存在关系的模型设计的窍门:  将多个文件想想成多个相互存在关系的表。找到用来连接的键。 | **stu.txt**  001 zhangsan  002 lisi  003 wangwu  004 zhaoliu  001 zhangsan\_a  001 java\_b  001 web\_b  001 hadoop\_b  **course.txt**  java 001  web 001  hadoop 001  php 002  ui 002  c 002  python 005 |  |

**六、教学过程脚本设计——第三课时**

1. **脚本设计思路**



1. **脚本切片设计思路**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **内容** | **小节目标** | **设计亮点** | **时间** |
| **1** | **课堂礼仪** | 集中学生的注意力，喊出士气，以饱满的精神状态投入本节课的学习 | 整齐、声音洪亮、铿锵有力，提升精气神！ |  |
| **2** | **为什么要学习爷孙关系这个例子** | 扩展解决问题的思路与设计模型的思路 |  |  |
| **3** | **需求** | 了解需求 |  |  |
| **4** | **模型设计分析** | 继续强化关系模型设计窍门 |  |  |
| **5** | **构建模型** | 构建模型 |  |  |
| **6** | **map程序设计** | 简略过代码，学生能够根据模型自行编写 |  |  |
| **7** | **reduce程序设计** | 简略过代码，学生能够根据模型自行编写 |  |  |
| **8** | **知识点总结** | 设计单个文件中具有关系的数据的模型 |  |  |

1. **脚本切片三：**为什么学习爷孙关系这个例子

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 为什么要压缩 | **强化关系模型的设计** | 能够设计具有自连接关系的模型 |  |
| 老师：上节课讲了具有两个文件存在连接关系的模型设计。这节课讲具有自连接关系的关系模型 | 1. 学习如何用MapReduce处理单个文件中具有关系的数据 2. 继续强化模型设计 |  |

1. **脚本切片三：**需求

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **压缩的定义** | 了解需求 | 清晰理解需求 |  |
| 老师：   * 左边是具有儿子与父亲的两列文本文件的内容 * 要求：找出并输出具有孙子与爷爷关系的人（输出两列、第一列是孙子、第二列是爷爷）。 | 孙子 爷爷  Jone Alice  Jone Jesse  Tom Alice  Tom Jesse  Jone Mary  Jone Ben  Tom Mary  Tom Ben  Mark Alice  Mark Jesse  Philip Alice  Philip Jesse  儿子 父亲  Tom Jack  Jone Lucy  Jone Jack  Lucy Mary  Lucy Ben  Tom Lucy  Jack Alice  Mark Terry  Jack Jesse  Mark Alma  Terry Alice  Terry Jesse  Philip Terry  Philip Alma |  |

1. **脚本切片三：**模型设计窍门

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **压缩的注意事项** | **了解模型设计的窍门** | 强化关系模型设计 |  |
| 老师：  1：针对这样的独立文件可以看做是一个表。  2：分析发现这样的表具有自连接关系（可以将自连接看成是两个记录内容完全相同的表）  3：通过连接写出连接两个内容完全相同的表的sql语句。连接的关键字就作为Map模型的键。  4：连接的关键字就作为Map的key | 本例子：sql语句的条件形式：找出身份具有儿子或父亲的人 |  |

1. **脚本切片三：**分析视图

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **压缩需要继承的类** | 了解压缩需要继承的类 |  |  |
| 孩子 父亲  Tom Jack  Jone Lucy  Jone Jack  Lucy Mary  Lucy Ben  Tom Lucy  Jack Alice  Mark Terry  Jack Jesse  Mark Alma  Terry Alice  Terry Jesse  Philip Terry  Philip Alma  孩子 父亲  Tom Jack  Jone Lucy  Jone Jack  Lucy Mary  Lucy Ben  Tom Lucy  Jack Alice  Mark Terry  Jack Jesse  Mark Alma  Terry Alice  Terry Jesse  Philip Terry  Philip Alma  Ben是tom，Jone的爷爷  Mary是tom ， jone的爷爷  Lucy Ben\_父  Lucy Jone\_儿  Lucy Mary\_父  Lucy Tom\_儿  将右侧的表列调换位置 | |  |

1. **脚本切片四：构建模型**

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **开发combiner** | 根据需求分析构建模型 | 准确构建模型  Lucy Ben\_父  Lucy Jone\_儿  Lucy Mary\_父  Lucy Tom\_儿  Ben是tom，Jone的爷爷  Mary是tom ， jone的爷爷 |  |
| 讲师：   * map<儿子，父亲\_p> 或map<父亲，儿子\_c> * reducer<孙子，爷爷> |  |  |

1. **脚本切片四：Map程序**

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **编写并启动程序** | 根据模型开发Map程序 | 根据模型自行开发Map程序 |  |
| 讲师：过一下Map程序，课后每个同学自行完成Map程序开发 | public class YessunMapper extends Mapper<LongWritable, Text, Text, Text> {  @Override  protected void map(LongWritable key, Text value, Mapper<LongWritable, Text, Text, Text>.Context context)  throws IOException, InterruptedException {  String a = value.toString();  String b[] = a.split("\t");//数组b里包括了两个人，子与父  String child=b[0];//儿子  String parent = b[1];//父亲  context.write(new Text(b[0]), new Text(b[1]+“\_p”)); //相当于第一个表（文件）  context.write(new Text(b[1]), new Text(b[0]+“\_c”)); //相当于第二个表（文件）  }  } |  |

1. **脚本切片四：Reduce程序**

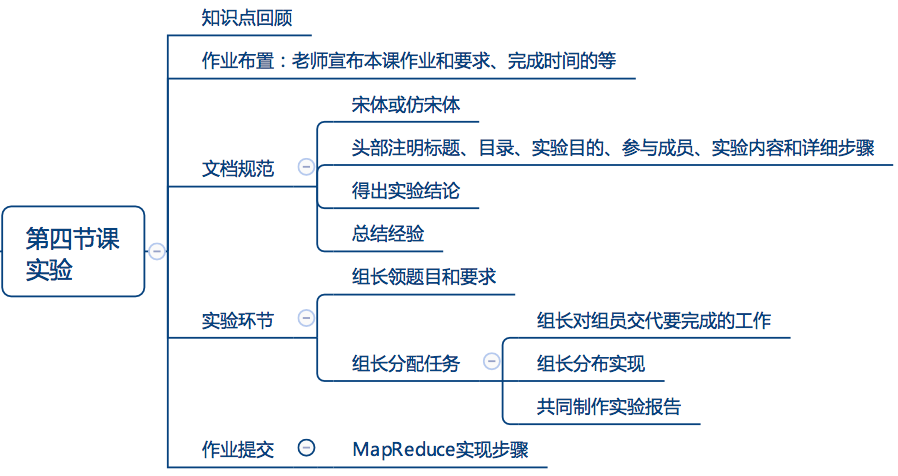
|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **编写并启动程序** | 根据模型开发Reduce程序 | 根据模型自行开发Reduce程序 |  |
| 讲师：过一下Reduce程序，课后每个同学自行完成Reduce程序开发 | public class YesunReducer extends Reducer<Text, Text, Text, Text> {  @Override  protected void reduce(Text key/\*传进来的键  jack\*/, Iterable<Text> values/\*值的集合\*/, Reducer<Text, Text, Text, Text>.Context context) throws IOException, InterruptedException {  Iterator ita = values.iterator();//获取循环的遍历器  List<String> lc= new ArrayList<String>();//存放\_c结尾的人  List<String> lp= new ArrayList<String>();//存放\_p结尾的人  while(ita.hasNext()){//先判断values里有没有内容  Text t = (Text)ita.next();//返回当前的内容并将指针移动到下一个内容  String s1 = t.toString();  if(s1.endsWith("\_c")){//如果以\_c结尾  s1 = s1.substring(0,s1.indexOf("\_"));//indexof获取字符在字符串中的位置  lc.add(s1);  }  if(s1.endsWith("\_p")){//如果以\_p结尾  s1 = s1.substring(0,s1.indexOf("\_"));//indexof获取字符在字符串中的位置  lp.add(s1);  }  }  for(String c:lc){  for(String p:lp){  context.write(new Text(c), new Text(p));  }  }  }} |  |

1. **脚本切片四：编只是点回顾**

|  |  |  |  |
| --- | --- | --- | --- |
| **模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| **知识点总结** | 知识点总结 |  |  |
| 讲师：  设计具有自连接关系的表模型 |  |  |

**七、教学过程脚本设计——第四课时**

1. **脚本设计思路**



1. **脚本切片设计思路**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **内容** | **小节目标** | **设计亮点** | **时间** |
| **1** | **需求** | 理解，掌握，自行编写 |  |  |
| **2** | **实践文档提交要求** | 文档的提交要求 |  |  |

1. **脚本切片三：**降雨量程序实践

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 降雨量程序实践 | **需求** | 强化编程思想 |  |
| **老师：**大家先熟悉一下需求  以下是要计算的文件内容，共5列内容：  第一列：id  第二列：电话号  第三列：月收入  第四列：地址  如：  1 135\*\*\*\*9365 林\*彬 2484 北京市\*\*\*\*三旗村  **要求:**  计算每个用户的年收入。按照电话号码分区，分区要求按照电话号段分，如下：  第一区：130，133，134，135，136，137，138，139  第二区：150，151，153，158，159  第三区：170，180，181，183，185，186，187，188，189 |  |  |

1. **脚本切片三：**实践文档提交要求

|  |  |  |  |
| --- | --- | --- | --- |
| **课堂模块** | **教学内容/台词** | **教学标准/素材** | **时长** |
| 实践文档提交要求 |  |  |  |
| 老师：  本课时根据上节课的安装讲解，由教师进行安装辅导，学生小组成员为每4个人一组（按照班级已有组别）进行MapReduce程序的调试。下课提交Word版实验报告，要求每组每个同学调试并运行通讲课的案例。  步骤5教师会在下节课细讲实验步骤，学生请比较与自己的实现有什么不同。  实验报告要求：   1. 提交MapReduce的运行流程说明（map阶段业务说明，reduce阶段业务说明） 2. 提交自定义键值的描述，提交分区与压缩的实现机制以及运行原理说明；代码要有注释   3. 提交团队成员实验主要分工；  4. 对mapreduce中的示例代码进行开发调试(1组、2组……)；  5.提交疑难问题列表  6. 程序分析疑难问题解决 |  |  |