任务2-1 深入理解RDD的操作

|  |  |
| --- | --- |
| 任务书 | |
| 一、任务描述与要求 | |
| 任务描述 | 深入理解RDD（弹性分布式数据集）的定义、特性和内部结构，包括分区、依赖关系和计算阶段等核心概念。 |
| 具体要求 | 1.选择适当的RDD操作来优化数据处理流程；  2.了解RDD的核心特性。 |
| 二、任务目标 | |
| 知识目标 | * 理解 RDD 的基本概念和特性； * 掌握 RDD 的不同转换和行动操作。 |
| 能力目标 | * 能够熟练运用 RDD 操作进行数据处理； * 具备基于 RDD 优化数据处理流程的能力。 |
| 三、知识储备 | |
| 1.什么是RDD？  RDD（Resilient Distributed Dataset）叫做弹性分布式数据集，是Spark中最基本的数据抽象，代表一个不可变、可分区、里面的元素可并行计算的集合。    RDD是三个英文单词的缩写，分别是：Resilient、Distributed以及Dataset，三个单词分别代表各自的含义。首先，Resilient代表RDD中的数据可以存储在内存中或者磁盘中；其次，Distributed代表RDD中的数据是分布式存储的，可用于分布式计算；最后，Dataset代表它是一个数据集合，用于存放数据的。    2. RDD的创建方法  RDD创建的方式有很多种，最主流的方式有两种：从现有集合创建和从外部数据源创建。当然，除了这两种以外还有一些其他的创建方法，这里展开说明，具体创建方法如下：  （1）从现有集合创建(2)从外部数据源创建（3）从其他RDD转换（4）从Hive查询创建（5）从数据流创建（6）使用wholeTextFiles方法（7）从序列化文件创建（8）使用SparkContext的emptyRDD方法 | |

|  |
| --- |
| 操作指南 |
| 一、制定计划 |
| |  |  |  |  | | --- | --- | --- | --- | | **序号** | **作业项目** | **序号** | **作业项目** | | 1 |  | 5 |  | | 2 |  | 6 |  | | 3 |  | 7 |  | | 4 |  | 8 |  | | 计划审核 | 审核意见： | | | |
| 二、实施方案 |
| 1. 创建 SparkContext 和模拟数据  这个案例将模拟一个简单的学生成绩分析系统。创建一些学生成绩数据，然后进行一些基本的分析。  这一步创建了 SparkContext 并使用 parallelize 方法从内存中的列表创建了一个 RDD，具体代码如下：  import org.apache.spark.{SparkConf, SparkContext}  val conf = new SparkConf()  .setAppName("StudentGradeAnalysis").setMaster("local[\*]")  val sc = new SparkContext(conf)  // 创建模拟的学生成绩数据：(学生名字, 科目, 成绩)  val gradesData = List(  ("张三", "数学", 85),  ("李四", "数学", 76),  ("王五", "数学", 90),  ("马六", "英语", 93),  ("牛二", "英语", 82),  ("小明", "英语", 88),  ("小红", "语文", 88),  ("小绿", "语文", 94),  ("小蓝", "语文", 91)  )  val gradesRDD = sc.parallelize(gradesData)  2. 计算每个学生的平均成绩  这里使用 map、groupBy 和 mapValues 操作来计算每个学生的平均成绩。具体代码如下：  val studentAvgGrades = gradesRDD  .map { case (name, \_, grade) => (name, (grade, 1)) }  .groupBy(\_.\_1)  .mapValues { grades =>  val (total, count) = grades.map(\_.\_2).reduce((a, b)  => (a.\_1 + b.\_1, a.\_2 + b.\_2))  total.toDouble / count  }  println("每个学生平均分:")  studentAvgGrades.collect().foreach(println)  3.找出每个科目的最高分  在这一步中重组数据并使用 groupBy 和 mapValues 来找出每个科目的最高分，具体代码如下：  val topGradesPerSubject = gradesRDD  .map { case (name, subject, grade) => (subject, (name, grade)) }  .groupBy(\_.\_1)  .mapValues(\_.map(\_.\_2).maxBy(\_.\_2))  println("\n每个科目最高分:")  topGradesPerSubject.collect().foreach(println)  4.计算所有成绩的总和和平均值  这里使用 map 和 reduce 操作来计算所有成绩的总和和平均值，具体代码如下：  val (totalSum, totalCount) = gradesRDD  .map { case (\_, \_, grade) => (grade, 1) }  .reduce((a, b) => (a.\_1 + b.\_1, a.\_2 + b.\_2))  val averageGrade = totalSum.toDouble / totalCount  println(s"\n总分: $totalSum")  println(f"平均分: $averageGrade%.2f")  5.统计每个分数段的学生人数  最后，使用 map 和 countByValue 操作来统计每个分数段的学生人数，具体代码如下：  val gradeDistribution = gradesRDD  .map { case (\_, \_, grade) =>  grade match {  case g if g >= 90 => "A"  case g if g >= 80 => "B"  case g if g >= 70 => "C"  case \_ => "D"  }  }  .countByValue()  println("\n登记:")  gradeDistribution.foreach { case (grade, count) =>  println(s"$grade: $count")  }  // 关闭SparkContext  sc.stop() |
| 三、实施记录 |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **序号** | **关键步骤记录** |  | **序号** | **解决问题记录** | | 1 |  |  | 1 |  | | 2 |  |  | 2 |  | | 3 |  |  | 3 |  | | 4 |  |  | 4 |  | | 5 |  |  | 5 |  | | 6 |  |  | 6 |  | | 7 |  |  | 7 |  | | 8 |  |  | 8 |  | | 9 |  |  | 9 |  | |
| 四、任务拓展 |
| 对比不同的 RDD 操作组合在处理速度和资源消耗上的差异，探索如何优化操作流程以提高效率。同时，可以尝试结合 Spark 的其他组件，进一步拓展 RDD 的应用场景。 |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 评价标准 | | | | | | | | |
| 一、基本信息 | | | | | | | | |
| 项目名称 | | 单元二 深入RDD | | | | | | |
| 任务名称 | | 任务2-1 深入理解RDD的操作 | | | | | | |
| 班级 |  | | 学号 | |  | 姓名 | |  |
| 组名 |  | | 学时 | |  | 日期 | |  |
| 组员  分工 |  | | |  | | |  | |
|  | | |  | | |  | |
| 二、任务检查评价单 | | | | | | | | |
| |  |  |  | | --- | --- | --- | | **评价内容** | **评价标准** | **得分** | | 知识运用  （20分） | 掌握相关理论知识，理解本次任务要求，制定详细计划，计划条理清晰，逻辑正确（20分） |  | | 理解相关理论知识，能根据本次任务要求、制定合理计划（15分） | | 了解相关理论知识，有制定计划（10分） | | 无制定计划（0分） | | 专业技能  （40分） | 结果验证全部满足。（40分） |  | | 结果验证只有一个功能不能实现，其它功能全部实现（30分） | | 结果验证只有一个功能实现，其它功能全部没有实现（20分） | | 结果验证功能均未实现（0分） | | 核心素养  （20分） | 具有良好的自主学习能力、分析解决问题的能力、整个任务过程中有指导他人（20分） |  | | 具有较好的学习能力和分析解决问题的能力，任务过程中无指导他人（15分） | | 能够主动学习并收集信息，有请教他人进行解决问题的能力（10分） | | 不主动学习（0分） | | 课堂纪律  （20分） | 设备无损坏、设备摆放整齐、工位区域内保持整洁、无干扰课堂秩序（20分） |  | | 设备无损坏、无干扰课堂秩序（15分） | | 无干扰课堂秩序（10分） | | 干扰课堂秩序（0分） | | 得分（满分100） | |  | | 综合评价 | ☆☆☆☆☆ | | | | | | | | | | |