

大数据学科hadoop阶段考试（二）

成绩单

答题人

张超

您的得分

0/100

答对题数

0/9

答题解析

全部题目

错题集

基本信息： * [矩阵文本题]

- (1) 姓名：张超
- (2) 班级：SH200317

1. 请用Hive完成

已知一个表order_tab，有如下字段:Date， Order_id， User_id， amount。请给出sql进行统计：
数据样例：

2017-01-01,10029028,1000003251,33.57。

- 1) 给出 2017年每个月的订单数、用户数、总成交金额。
- 2) 给出2017年11月的新客数(指在11月才有第一笔订单)

建表语句如下：

create table order_tab(dt string,order_id string,user_id string,amount decimal(10,2)) row form
at delimited fields terminated by '\t'; [分值： 20]

您的回答： select count(user_id) from (select user_id from order_tab group by user_id);
select count(order_id) order_count, count(distinct(user_id)) user_count, sum(amount) all, substring(dt,1,
7) month from order_tab
where substr(ing(dt, 1,4))='2017'
group by month;

select count(1) from

```
( select order_id, log(dt, 1) over(partition by user_id order by date) first_Order from order_tab) t1
where substring(dt, 1, 7) = '2017-11' and first_Order is null;
```

(简答题需要人工评分)

答案解析：

1)select

```
date_format(dt,'yyyy-MM'),
count(order_id),
count(distinct user_id),
sum(amount)
```

from

```
order_tab
```

```
group by date_format(dt,'yyyy-MM');
```

2)select

```
tmp.mn,
count(*)
```

from

```
(
```

```
select
```

```
date_format(min(dt),'yyyy-MM') mn,
user_id
```

```
from
```

```
order_tab
```

```
group by user_id
```

```
having date_format(min(dt),'yyyy-MM')='2017-11'
```

```
) tmp
```

```
group by tmp.mn;
```

2. 请列举常用Linux命令，并说明命令用途。 [分值：10]

您的回答：ls 列出当前目录下文件

cd 切换至某位置

cat 显示某文件内容 less more 分屏显示内容

mkdir 创建文件夹 touch 创建文件

vim 使用vim编辑

ifconfig 查看网络信息等

grep 输出筛选 find 查找文件

systemctl start/stop/status xxx 开启/关闭/查看状态 xx

mv 移动或重命名

cp 复制

ssh ssh-keygen ssh-copy-id ssh相关命令

> >> 输出重定向 覆盖写/追加写

head 显示文件头部信息 tail 显示文件尾部信息

chmod 改变文件权限 chown 改变文件所有者 chgrp 改变所属组

ps 查看系统进程 kill 终止进程 pstree 以树的形式查看进程 top 查看系统状态和资源占用情况 netstat 查看网络和端口信息

(简答题需要人工评分)

答案解析:

命令 命令解释

top 查看内存

df -h 查看磁盘存储情况

iotop 查看磁盘IO读写(yum install iotop安装)

iotop -o 直接查看比较高的磁盘读写程序

netstat -tunlp | grep 端口号 查看端口占用情况

uptime 查看报告系统运行时长及平均负载

ps aux 查看进程

3. 请列举hadoop常用端口号。 [分值: 10]

您的回答: 内部通讯:

9820 namenode内部通讯

9869 2nn内部通讯

8032 resourcemanager内部通讯

10020 jobhistoryserver内部通讯

web:

9870 namenode web访问

9868 2nn web访问

8088 resourcemanager web访问

19888 jobhistoryserver web访问

(简答题需要人工评分)

答案解析:

dfs.namenode.http-address: 50070/9870

dfs.datanode.http-address:50075/9864

SecondaryNameNode辅助名称节点端口号: 50090/9868

dfs.datanode.address:50010/9866

fs.defaultFS:8020 /9000/9820

yarn.resourcemanager.webapp.address:8088

历史服务器web访问端口：19888

4. 简述hadoop的MapReduce的Shuffle过程(文字描述)。 [分值：10]

您的回答：1. 在 map 方法之后, reduce 方法之前的处理过程就是 shuffle 过程.

2. map 方法写出去的 k-v, 会被一个收集线程收集到缓冲区中.

3. 缓冲区大小默认是 100M, 达到 80% 发生溢写, 缓冲区记录了 k-v, k-v 的下标, k-v 的分区等信息.

4. 溢写的时候, 是按照 k-v 的分区进行排序, 采用快排对索引进行排序, 再按照分区进行溢写, 从而完成 map 端的第一次排序.

5. 每个 maptask 有可能发生多次溢写, 最终需要将多次溢写的文件归并成一个大的文件.

6. 每个 reducetask 按照所要处理的分区, 到每个 maptask 中拷贝对应的分区数据, 先放到内存, 放不下写磁盘. 等数据全部拷贝过来进行归并排序.

7. reduce 端排好序的数据进行分组, 然后进入 reduce 方法进行业务处理.

(简答题需要人工评分)

答案解析：

- 1) MapTask收集我们的map()方法输出的kv对，放到内存缓冲区中
- 2) 从内存缓冲区不断溢出本地磁盘文件，可能会溢出多个文件
- 3) 多个溢出文件会被合并成大的溢出文件
- 4) 在溢出过程及合并的过程中，都要调用Partitioner进行分区和针对key进行排序
- 5) ReduceTask根据自己的分区号，去各个MapTask机器上取相应的结果分区数据
- 6) ReduceTask会取到同一个分区的来自不同MapTask的结果文件，ReduceTask会将这些文件再进行合并（归并排序）
- 7) 合并成大文件后，Shuffle的过程也就结束了，后面进入ReduceTask的逻辑运算过程（从文件中取出一个的键值对Group，调用用户自定义的reduce()方法）

5. 简述Flume组成（三个组件）及每个组件的常用类型（两个），并说明其特点。 [分值：10]

您的回答：source channel sink

source:

Spooldir Source 适合用于同步新文件, 但不适合对实时追加日志的文件进行监听和同步;

Taildir Source 适合用于监听多个实时追加的文件, 并且能够实现断点续传

Channel:

Memory Channel 效率高, 但可靠性低

File Channel 可靠性高, 但效率低

Sink:

hdfs 写入hdfs

avro 写入avro端口

file 写入某文件

logger 以日志方式输出

(简答题需要人工评分)

答案解析:

一、(1) Taildir Source: 断点续传、多目录。Flume1.6以前需要自己自定义Source记录每次读取文件位置,实现断点续传。(2) Avro Source: Avro端口监听并接收来自外部的Avro客户流的事件。(3) Exec Source: Exec Source的配置就是设定一个Unix(linux)命令,然后通过这个命令不断输出数据。如果进程退出,Exec Source也一起退出,不会产生进一步的数据。(4) Spooling Directory Source: Spooling Directory Source监测配置的目录下新增的文件,并将文件中的数据读取出来。

二、(1) File Channel: 数据存储在磁盘,宕机数据可以保存。但是传输速率慢。适合对数据传输可靠性要求高的场景,比如,金融行业。(2) Memory Channel: 数据存储在内存中,宕机数据丢失。传输速率快。适合对数据传输可靠性要求不高的场景,比如,普通的日志数据。(3) Kafka Channel: 减少了Flume的Sink阶段,提高了传输效率。

三、(1)HDFS Sink: 当需要将事件消息写入到Hadoop分布式文件系统(HDFS)时,可以使用HDFS Sink。

(2) Avro Sink: 和 Avro Source一起工作,用于构建Flume分层收集数据消息结构。(3) Kafka Sink: 通过该Sink可将事件消息数据发布到Kafka topic 上。其目标是将Flume与Kafka集成,以便基于拉式的处理系统可以处理来自各种Flume Source的数据。目前支持Kafka 0.9.x以上系列版本。

6. 结合数仓项目说明HDFS存储大量小文件造成的影响,以及HDFS Sink如何避免生成大量小文件。 [分值: 10]

您的回答: 对 hdfs 来说,一个文件需要由 namenode 在内存中维护一个 150 byte 的元数据信息,大量小文件会产生大量的元数据信息,比如128G就只能存9亿文件块. 所以,将小文件采用 har 归档的方式归档,或瞎用 CombineTextInputFormat, 或 KVM 重用都可以有效处理小文件带来的问题.

可以通过调整 hdfs sink 的参数:

hdfs.rollInterval=3600 每小时将临时文件滚动成正式文件

hdfs.rollSize=134217728 tmp 文件到达128m时才生成正式文件

hdfs.rollCount=0 不按照事务的数量滚动生成正式文件

(简答题需要人工评分)

答案解析:

(1) HDFS存入大量小文件,有什么影响?

元数据层面: 每个小文件都有一份元数据,其中包括文件路径,文件名,所有者,所属组,权限,创建时间等,这些信息都保存在Namenode内存中。所以小文件过多,会占用Namenode服务器大量内存,影响Namenode性能和使用寿命

计算层面: 默认情况下MR会对每个小文件启用一个Map任务计算,非常影响计算性能。同时也影响磁盘寻址时间。

(2) HDFS小文件处理

官方默认的这三个参数配置写入HDFS后会产生小文件，hdfs.rollInterval、hdfs.rollSize、hdfs.rollCount,基于以上hdfs.rollInterval=3600, hdfs.rollSize=134217728, hdfs.rollCount =0, hdfs.roundValue=10, hdfs.roundUnit= second几个参数综合作用，效果如下：

- (1) tmp文件在达到128M时会滚动生成正式文件
- (2) tmp文件创建超10秒时会滚动生成正式文件

举例：在2018-01-01 05:23的时候sink接收到数据，那会产生如下tmp文件：

/atguigu/20180101/atguigu.201801010520.tmp

即使文件内容没有达到128M，也会在05:33时滚动生成正式文件

7. 请简单说明Kafka消费者的分区分配策略。 [分值：10]

您的回答：一个消费者 group 中有许多的消费者，一个 topic 有多个分区，消费者组内每个消费者负责不同分区的数据，一个分区只能由一个组内消费者消费，但消费者之间互不影响。

kafka 有 roundrobin 和 range 两种分区分配策略。

roundrobin 即轮询，多个分区依次分给组内的消费者；range 即先划分再分配，先按照同一组内消费者数量将分区均分，再将均分后的分区依次分给每一个消费者。

(简答题需要人工评分)

答案解析：

在 Kafka内部存在两种默认的分区分配策略：Range和 RoundRobin。

Range是默认策略。Range是对每个Topic而言的（即一个Topic一个Topic分），首先对同一个Topic里面的分区按照序号进行排序，并对消费者按照字母顺序进行排序。然后用Partitions分区的个数除以消费者线程的总数来决定每个消费者线程消费几个分区。如果除不尽，那么前面几个消费者线程将会多消费一个分区。

RoundRobin：前提：同一个Consumer Group里面的所有消费者的num.streams（消费者消费线程数）必须相等；每个消费者订阅的主题必须相同。

8. 请对Hive的内部表和外部表做出说明。 [分值：10]

您的回答：内部表 MANAGED TABLE也叫管理表。hive 会控制内部表数据的生命周期。当我们删除一个内部表时，hive 也会删除相应的这个表中的数据，所以内部表并不适合和其他工具共享数据。

与之对应的是外部表 EXTERNAL TABLE, hive 并非认为其完全拥有这份数据。删除外部表并不会删除相应的数据，只会删除元数据信息。

内部表和外部表可以相互转换。

一些日志文件生为基础的表适合作为外部表，而由此进行分析得到的中间表和结果表适合用内部表。

(简答题需要人工评分)

答案解析：

- 1) 管理表：当我们删除一个管理表时，Hive也会删除这个表中数据。管理表不适合和其他工具共享数据。
- 2) 外部表：删除该表并不会删除掉原始数据，删除的是表的元数据

9. 简述使用sqoop进行hive与mysql的导入导出时应该注意哪些问题？如何解决？ [分值：10]

您的回答： 需要对不同种类的数据表区别处理, 存储完整数据的全量表, 存储新增加的数据的增量表, 存储新增加的数据和变化的数据的新增及变化表, 只需要存储一次的特殊表等.

全量同步策略: 每天存出一份完整的数据作为一个分区. 适合数据量不大且每天既会有新数据的插入, 也会有旧数据的修改的场景.

增量同步策略: 每天存储一份增量数据, 作为一个分区. 适合表数据量大, 且每天只会有新数据插入的场景.

新增及变化同步策略: 存储创建时间和操作时间都是今天的数据. 适合表的数据量大, 既有新增, 又有变化.

特殊策略: 一些特殊维度的表, 存储固定值或一次生成一个时间维度的数据.

(简答题需要人工评分)

答案解析：

Hive中的Null在底层是以“\N”来存储，而MySQL中的Null在底层就是Null，为了保证数据两端的一致性。在导出数据时采用--input-null-string和--input-null-non-string两个参数。导入数据时采用--null-string和--null-non-string。

--staging-table一致性问

对于近期的学习你有什么想说的？

您的回答： (空)

恭喜您获得了1次抽奖机会！

20元
微信红包

感谢参与

6瓶
安慕希酸奶

感谢参与

立即
抽奖

感谢参与

10元
微信红包

感谢参与

200元
中小学名师课

举报

问卷星 提供技术支持