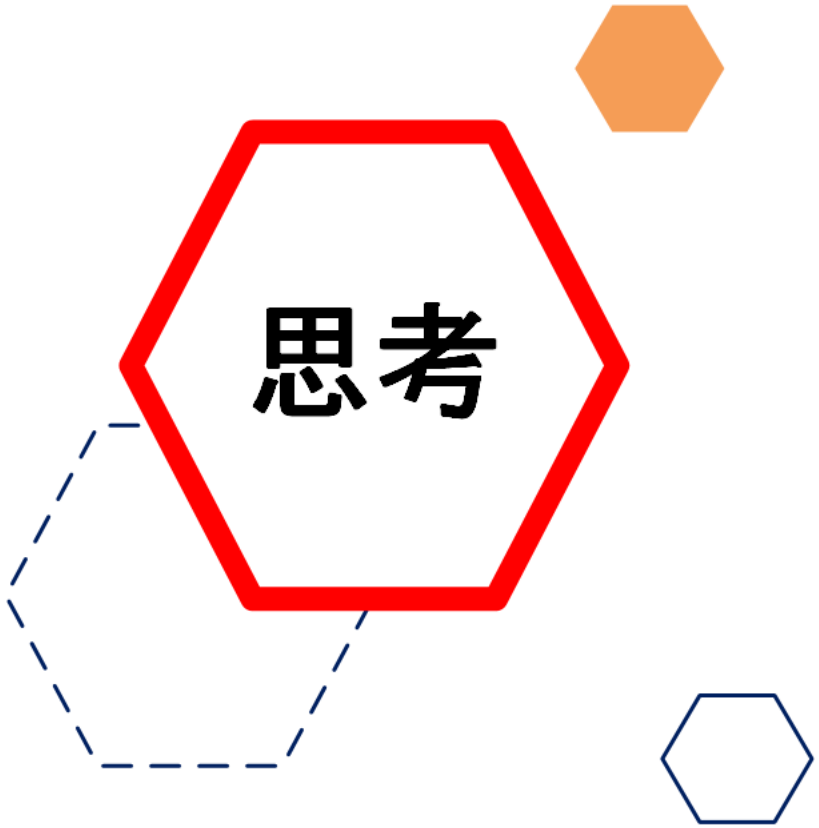
任务 2-4

Hive 体验

2.4.1 体验1：Hive使用起来和MySQL差不多吗?



1. 对于初次接触Apache Hive的人来说，最大的疑惑就是：
Hive从数据模型看起来和关系型数据库MySQL等好像。
Hive SQL也是一种类SQL语言。
那么实际使用起来如何?

体验过程

- 按照MySQL的思维，在hive中创建、切换数据库，创建表并执行插入数据操作，最后查询是否插入成功。

```
1
2   -- 体验1
3   hive-bee
4
5   show databases; --列出所有数据库
6   create database itcast; --创建数据库
7   use itcast; --切换数据库
8   show tables;
9
10  create table t_user (id int, name string);
11
12  -- 여기서 소요시간 체크 : 약 30초 소요됨)
13  insert into table t_user  values(1,'zhangsan');
14  select * from t_user;
15
```

```
0: jdbc:hive2://localhost:10000> show databases;
+---------------+
| database_name  |
+---------------+
| default        |
+---------------+
0: jdbc:hive2://localhost:10000> create database itcast;
0: jdbc:hive2://localhost:10000> use itcast;
0: jdbc:hive2://localhost:10000> show tables;
+-----------+
| tab_name  |
+-----------+
+-----------+
0: jdbc:hive2://localhost:10000> create table t_user (id int, name string);
0: jdbc:hive2://localhost:10000> insert into table t_user  values(1,'zhangsan');
0: jdbc:hive2://localhost:10000> select * from t_user;
+------------+--------------+
| t_user.id  | t_user.name  |
+------------+--------------+
| 1          | zhangsan     |
+------------+--------------+
```

体验过程

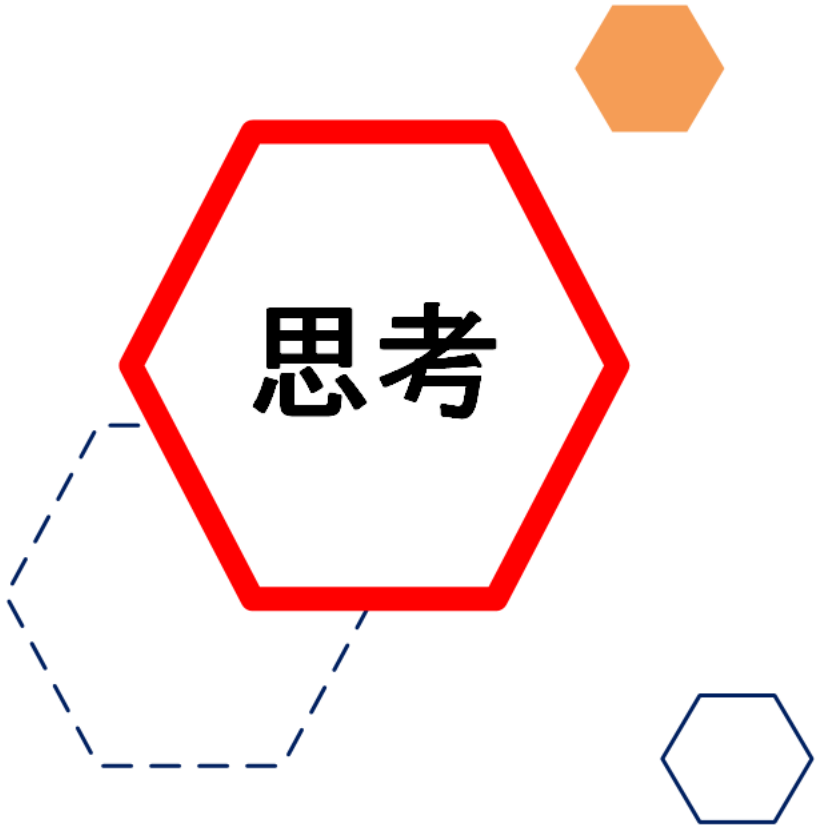
- 在执行插入数据的时候，发现 **插入速度极慢**，sql执行时间很长，为什么?
  - Hive是为**批量数据分析**设计的，不是为**实时事务处理**设计的
  - 单条插入操作会触发整个MapReduce/Tez作业的启动和调度
  - 每条插入都会生成小文件，对HDFS Namenode造成压力

- 最终 **插入一条数据**，历时**30秒**的时间。查询表数据，显示数据插入成功

结论

- 在Hive SQL语法和标准SQL很类似，使得学习成本降低不少。
- Hive底层是通过MapReduce执行的数据插入动作，所以速度慢。
- 如果大数据集这么一条一条插入的话是非常不现实的，成本极高。
- Hive应该具有自己特有的数据插入表方式，结构化文件映射成为表。

2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?



1. 在Hive中，使用insert+values语句插入数据，底层是通过MapReduce执行的，效率十分低下。此时回到Hive的本质上：可以将结构化的数据文件映射成为一张表，并提供基于表的SQL查询分析。

2. 假如，现在有一份结构化的数据文件，如何才能映射成功呢？在映射成功的过程中需要注意哪些问题？

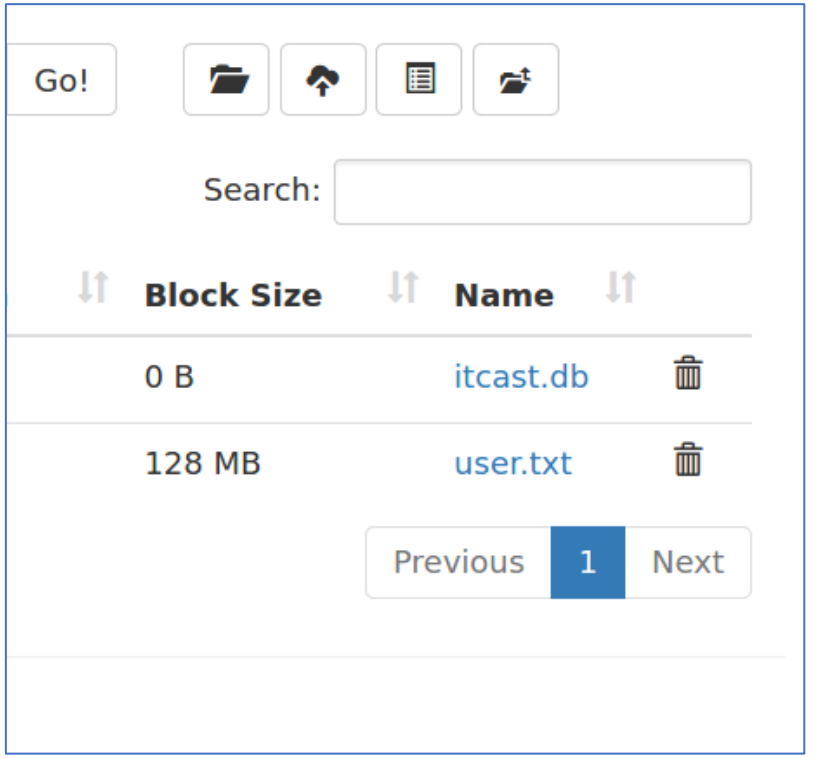
3. 文件存储路径？字段类型？字段顺序？字段之间分隔符？

2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

体验过程

- 在HDFS /user/hive/warehouse目录下创建一个结构化数据文件user.txt、里面内容如下:

```
1,zhangsan,18
2,lisi,24
3,allen,22
4,james,28
```



http://localhost:9870/explorer.html#

user →hive → warehouse →user.txt

2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

验证

```
+-------------+---------------+--------------+
| t_user.id   | t_user.name   | t_user.age   |
+-------------+---------------+--------------+
+-------------+---------------+--------------+
```

- 执行数据查询操作，发现表中并没有数据。

```
0: jdbc:hive2://localhost:10000> select * from t_user;
+-------------+---------------+--------------+
| t_user.id   | t_user.name   | t_user.age   |
+-------------+---------------+--------------+
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
+-------------+---------------+--------------+
```

- 猜想1：难道数据文件要放置在表对应的HDFS路径下才可以成功?

使用HDFS命令将数据移动到表对应的路径下。

2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

验证

- 重建张新表，指定分隔符。

```
create table t_user1 (id int, name  string, age int)
row format delimited
fields terminated by ',';

show tables;

hdfs dfs -put user.txt /user/hive/warehouse/itcast.db/t_user1
hdfs dfs -ls -R /user/hive/warehouse

select * from t_user1;
```

```
+-------------+---------------+--------------+
| t_user.id   | t_user.name   | t_user.age   |
+-------------+---------------+--------------+
+-------------+---------------+--------------+
```

```
0: jdbc:hive2://localhost:10000> select * from t_user;
+-------------+---------------+--------------+
| t_user.id   | t_user.name   | t_user.age   |
+-------------+---------------+--------------+
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
+-------------+---------------+--------------+
```

```
0: jdbc:hive2://localhost:10000> select * from t_user1;
+--------------+----------------+--------------+
| t_user1.id   | t_user1.name   | t_user1.age  |
+--------------+----------------+--------------+
| 1            | zhangsan       | 18           |
| 2            | lisi           | 24           |
| 3            | allen          | 22           |
| 4            | james          | 28           |
+--------------+----------------+--------------+
```

2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

验证

- 此时再创建一张表，保存分隔符语法，但是故意使得字段类型和文件中不一致。
- 此时发现，有的列显示null，有的列显示正常。

name字段本身是字符串，但是建表的时候指定int，类型转换不成功；age是数值类型，建表指定字符串类型，可以转换成功。说明hive中具有自带的类型转换功能，但是不一定保证转换成功。

```
+-------------+---------------+--------------+
| t_user.id   | t_user.name   | t_user.age   |
+-------------+---------------+--------------+
+-------------+---------------+--------------+
```

```
0: jdbc:hive2://localhost:10000> select * from t_user;
+-------------+---------------+--------------+
| t_user.id   | t_user.name   | t_user.age   |
+-------------+---------------+--------------+
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
| NULL        | NULL          | NULL         |
+-------------+---------------+--------------+
```

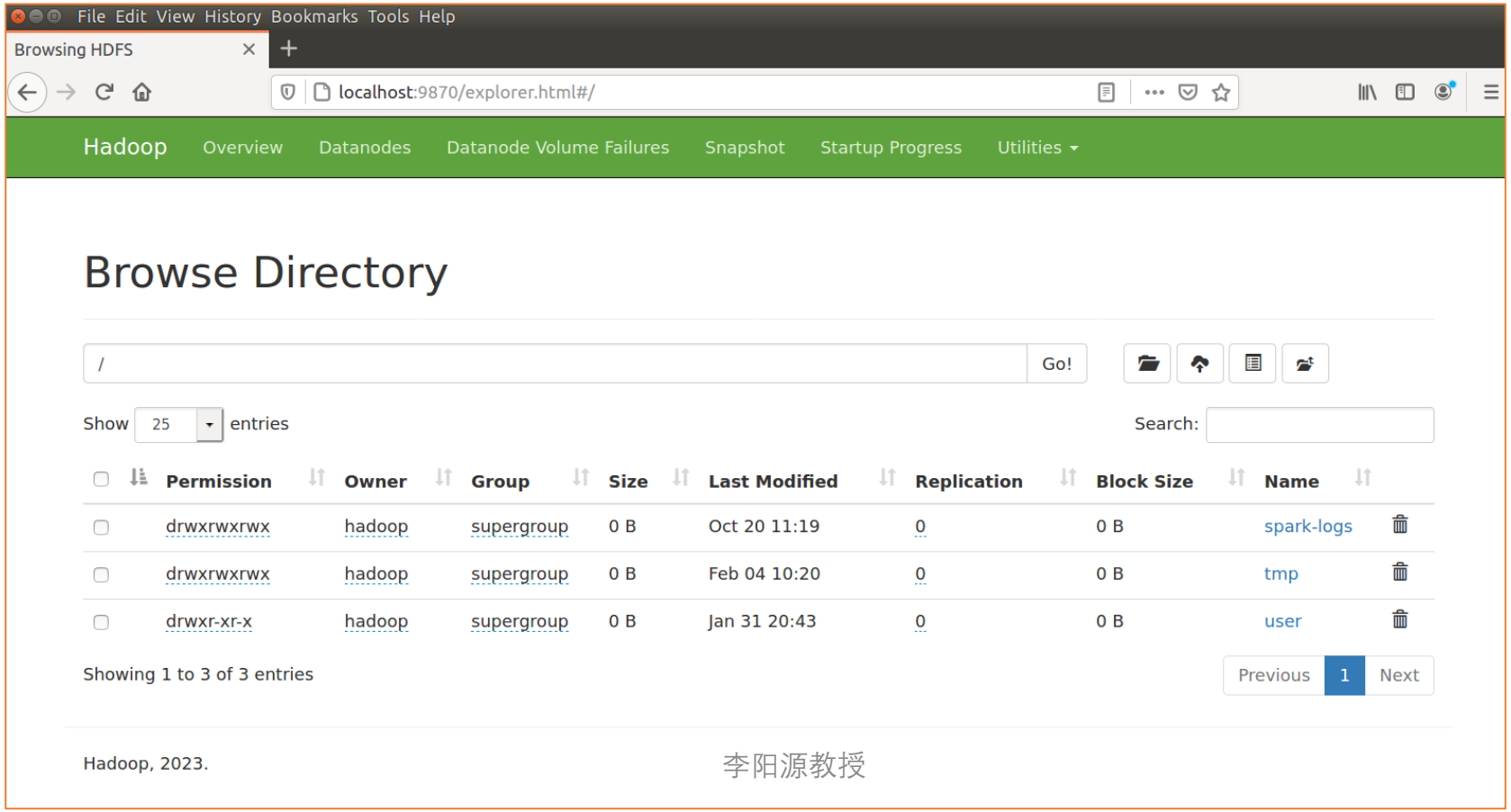
```
0: jdbc:hive2://localhost:10000> select * from t_user1;
+--------------+----------------+--------------+
| t_user1.id   | t_user1.name   | t_user1.age  |
+--------------+----------------+--------------+
| 1            | zhangsan       | 18           |
| 2            | lisi           | 24           |
| 3            | allen          | 22           |
| 4            | james          | 28           |
+--------------+----------------+--------------+
```

```
0: jdbc:hive2://localhost:10000> select * from t_user2;
+--------------+----------------+--------------+
| t_user2.id   | t_user2.name   | t_user2.age  |
+--------------+----------------+--------------+
| 1            | NULL           | 18           |
| 2            | NULL           | 24           |
| 3            | NULL           | 22           |
| 4            | NULL           | 28           |
+--------------+----------------+--------------+
```

2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

体验过程

- http://localhost:9870/explorer.html#/

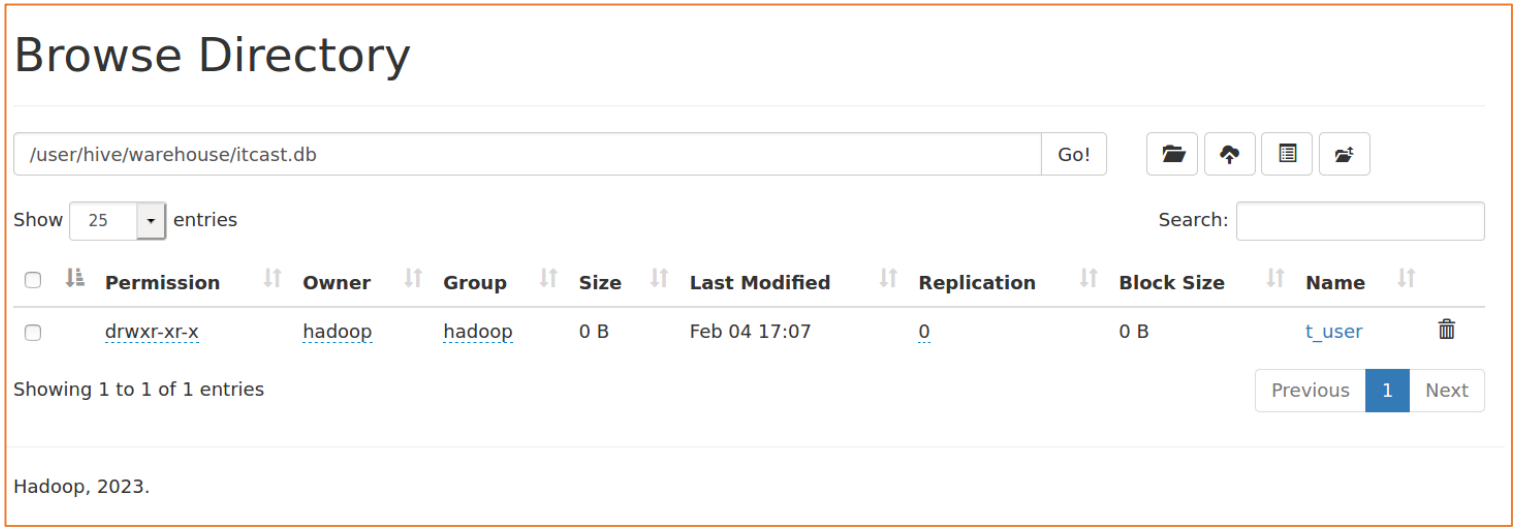


2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

体验过程

- http://localhost:9870/explorer.html#/

user →hive → warehouse → itcast.db → t_user →



2.4.2 体验2：Hive如何才能够将结构化数据映射成为表?

结论

- 要想在hive中创建表跟结构化文件映射成功，需要注意以下几个方面问题：

  创建表时，字段顺序、字段类型要和文件中保持一致。

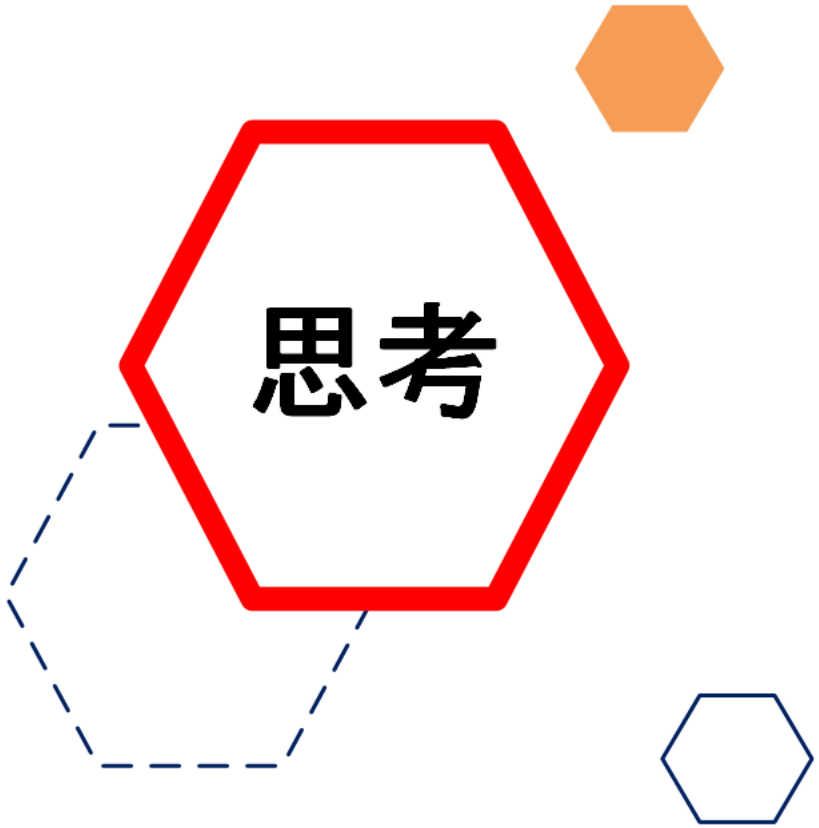
  如果类型不一致，hive会尝试转换，但是不保证转换成功。不成功显示null。

  文件好像要放置在Hive表对应的HDFS目录下，其他路径可以吗？值得探讨。

  建表的时候好像要根据文件内容指定分隔符，不指定可以吗？值得探讨。

- 其实上面这些问题都是Hive的各个知识点，有待于我们接下去不断深入学习。

2.4.3 体验3：使用Hive小数据分析如何?



1. 因为Hive是基于HDFS进行文件的存储，所以理论上能够支持的数据存储规模很大，天生适合大数据分析。假如Hive中的数据是小数据，再使用Hive开展分析效率如何呢?

2.4.3 体验3：使用Hive小数据分析如何?

体验过程

- 之前我们创建好了一张表t_user1, 现在通过Hive SQL找出当中年龄大于20岁的有几个。

```
1  -- 体验3：使用Hive小数据分析如何？
2
3  show tables;
4  select * from t_user1;
5
6  select count(id) from t_user1 where age > 20;
```

- 发现又是通过MapReduce程序执行的数据查询功能。

```
0: jdbc:hive2://localhost:10000> select * from t_user1;
+-------------+---------------+--------------+
| t_user1.id  | t_user1.name  | t_user1.age  |
+-------------+---------------+--------------+
| 1           | zhangsan      | 18           |
| 2           | lisi          | 24           |
| 3           | allen         | 22           |
| 4           | james         | 28           |
+-------------+---------------+--------------+
```

```
0: jdbc:hive2://localhost:10000> select count(id) from t_user1 where age > 20;
+------+
| _c0  |
+------+
| 3    |
+------+
```

47秒

2.4.3 体验3：使用Hive小数据分析如何?

结论

- Hive底层的确是通过MapReduce执行引擎来处理数据的
- 执行完一个MapReduce程序需要的时间不短
- 如果是小数据集，使用hive进行分析将得不偿失，延迟很高
- 如果是大数据集，使用hive进行分析，底层MapReduce分布式计算，很爽

感谢您的观看



《Hive数据仓库应用教程》