Hive 数据仓库应用教程

项目四 Hive数据定义

—— 第1回 强义原稿（jiǎng yì yuán gǎo）——

讲课对象：专科大学学生

讲课语言：中文（口语体）

目录 / CONTENTS

1. 任务4-1 HiveQL操作数据库
2. 任务4-2 HiveQL操作数据表
3. 任务4-3 HiveQL操作分区表
4. 任务4-4 HiveQL操作分桶表
5. 任务4-5 HiveQL操作视图
6. 任务4-6 HiveQL相关操作

课程开始 任务 4-2 HiveQL操作数据表

幻灯片15：任务4-2 HiveQL操作数据表（标题页）

好，接下来我们进入第二个任务——HiveQL操作数据表。

数据库我们已经会创建了，现在要在数据库里面创建表。

表才是真正存放数据的地方。

幻灯片16：4.2.1 任务描述

数据库建好之后，我们需要在数据库中完成数据表的基本操作。

包括：创建数据表、查看数据表、修改数据表和删除数据表。

跟数据库的操作类似（lèi sì），但是表的操作更多一些，也更复杂（fù zá）一些。

幻灯片17-18：4.2.2 任务准备知识 —— CREATE TABLE句式分析

创建数据表的语法比创建数据库复杂很多。

不要害怕（hài pà），我们慢慢看。

CREATE TABLE 语句里面主要包含（bāo hán）这些内容：

表名、列名和列的数据类型、行格式（háng gé shì）、存储格式（cún chǔ gé shì）等。

我们后面会通过例子来看，这样更容易理解。

幻灯片19：4.2.2 任务准备知识 —— SerDe是什么

这里有一个新概念（gài niàn）叫做 SerDe。

SerDe 是 Serializer 和 Deserializer 的简称（jiǎn chēng）。

意思就是「序列化（xù liè huà）」和「反序列化」。

听起来很复杂对不对？其实很简单：

序列化就是把对象（duì xiàng）变成字节码，就像把一本书变成快递（kuài dì）包裹。

反序列化就是把字节码变回对象，就像收到快递后拆开包裹拿出书。

【图片说明】这张幻灯片上有两个流程图：

上面的图是「Reading Rows（读取行）」的流程：

HDFS File（HDFS文件）→ INPUTFORMAT → (SerDe) Deserializer（反序列化器）→ Record（记录）

意思是：从HDFS读取文件，经过输入格式处理，再通过反序列化，变成一条一条的记录。

下面的图是「Writing Rows（写入行）」的流程：

Record（记录）→ (SerDe) Serializer（序列化器）→ OUTPUTFORMAT → HDFS File（HDFS文件）

意思是：把记录通过序列化，经过输出格式处理，写到HDFS文件里。

Hive用SerDe来读和写表的数据。

简单记住：SerDe就是Hive读写数据时候的「翻译官（fān yì guān）」。

幻灯片20：4.2.2 任务准备知识 —— 数据表简介

Hive的数据表主要分为两种：内部表和外部表。

内部表，也叫做托管表（tuō guǎn biǎo），英文叫 Managed Table。

外部表，英文叫 External Table。

在内部表和外部表的基础上，还可以创建分区（fēn qū）表和分桶（fēn tǒng）表。

后面我们会详细讲分区表和分桶表。

幻灯片21-22：Hive 内、外部表 —— 什么是内部表

什么是内部表?

内部表就是Hive完全拥有和管理的表。

默认情况下，你创建的表就是内部表。

Hive会完全管理这个表的生命周期（shēng mìng zhōu qī），包括元数据（yuán shù jù）和实际数据。

什么意思呢？就是说当你删除内部表的时候，Hive不仅会删除表的定义信息，还会把数据文件也一起删掉。

> *可以用 desc formatted tablename 来查看一个表是内部表还是外部表。*

幻灯片23-24：Hive 内、外部表 —— 什么是外部表

外部表跟内部表不一样。

外部表的数据不是由Hive拥有和管理的，Hive只管理表的元数据。

创建外部表的时候，需要加上 EXTERNAL 这个关键字。

删除外部表的时候，只会删除元数据（也就是表的定义），但实际数据不会被删掉。

数据还在HDFS上，在Hive外面还可以访问（fǎng wèn）。

实际工作中，外部表经常搭配（dā pèi）LOCATION来指定数据路径，这样数据更安全。

幻灯片25：Hive 内、外部表 —— 差异

总结一下内部表和外部表的区别（qū bié）：

不管是内部表还是外部表，Hive都会在Metastore中管理表的元数据信息。

删除内部表的时候：元数据删了，HDFS上的数据文件也删了。彻底没了。

删除外部表的时候：只删除元数据，HDFS上的数据文件还在，不会丢。

> 💡 *就好比：内部表就像你自己养的宠物，你不要了，宠物也没了。外部表就像你只是给邻居的猫起了个名字，你不叫它了，猫还在邻居家。*

幻灯片26：如何选择内、外部表

什么时候用内部表？什么时候用外部表？

如果你需要Hive完全管理控制表的整个生命周期，用内部表。

如果数据来之不易（lái zhī bù yì），怕不小心删掉，就用外部表。

因为删外部表的时候数据文件还在，不会丢失（diū shī）。

幻灯片27-28：Location 关键字探究

关于LOCATION这个关键字，大家可能有疑问（yí wèn）：

创建外部表的时候如果不指定LOCATION会怎样？

答案：外部表的默认路径也是在 /user/hive/warehouse 下面。

内部表可以用LOCATION指定位置吗？

答案：可以的！内部表也能用LOCATION。

所以，不管内部表还是外部表，数据默认都在 /user/hive/warehouse 下面。

但是你可以通过LOCATION关键字指定任意（rèn yì）路径。

幻灯片29：4.2.2 任务准备知识 —— 3. 查看数据表

查看数据表有两种方式：

第一种，看当前数据库有哪些表：

```
SHOW TABLES [LIKE 'identifier_with_wildcards'];
```

第二种，看某个表的结构信息：

```
DESCRIBE|DESC [formatted] table_name;
```

加上 formatted 可以看到更详细的信息。

幻灯片30-31：4.2.2 任务准备知识 —— 4. 修改数据表

修改表用 ALTER TABLE，有很多种操作：

（1）重命名表：

```
ALTER TABLE table_name RENAME TO new_table_name;
```

（2）修改表属性：

```
ALTER TABLE table_name SET TBLPROPERTIES(property_name = property_value);
```

（3）修改表的列——可以改列名、类型、描述（miáo shù）、位置：

```
ALTER TABLE table_name CHANGE [COLUMN] col_old_name col_new_name column_type
[COMMENT col_comment] [FIRST|AFTER column_name];
```

（4）添加列：

```
ALTER TABLE table_name ADD COLUMNS (col_name data_type [COMMENT col_comment], ...);
```

（5）替换（tì huàn）所有列：

```
ALTER TABLE table_name REPLACE COLUMNS (col_name data_type [COMMENT
col_comment], ...);
```

注意：CHANGE是修改某一个列，ADD是在最后加新列，REPLACE是把所有列全部换成新的。

幻灯片32：4.2.2 任务准备知识 —— 5. 删除数据表

删除数据表的语法：

```
DROP TABLE [IF EXISTS] table_name [PURGE];
```

PURGE 的意思是彻底（chè dǐ）删除，不放到回收站（huí shōu zhàn）。

如果不加PURGE，删除的表会先放到回收站，还可以恢复（huī fù）。

加了PURGE就直接永久删除了，找不回来了。

幻灯片33：4.2.3 任务实施 —— 1. 创建内部表

好，现在我们来动手（dòng shǒu）实践。

首先在数据库 h_sample 中创建一个内部表 managed_table。

这个表有很多列，包括员工ID、姓名、工资，还有一些复杂类型的列。

【实习命令 / 실습 명령】

```
CREATE TABLE IF NOT EXISTS
h_sample.managed_table(
  staff_id INT COMMENT 'This is staffid',
  staff_name STRING COMMENT 'This is staffname',
  salary FLOAT COMMENT 'This is staff salary',
  hobby ARRAY<STRING> COMMENT 'This is staff hobby',
  deductions MAP<STRING, FLOAT> COMMENT 'This is staff deduction',
  address STRUCT<street:STRING,city:STRING> COMMENT 'This is staff address')
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
COLLECTION ITEMS TERMINATED BY '_'
MAP KEYS TERMINATED BY ':'
LINES TERMINATED BY '\n'
STORED AS textfile
TBLPROPERTIES('comment'='This is a managed table');
```

让我解释一下每一部分的意思：

ARRAY<STRING>——是一个数组类型，可以存多个值，比如多个爱好（ài hào）。

MAP<STRING, FLOAT>——是一个键值对（jiàn zhí duì）类型，比如扣除项（kòu chú xiàng）和对应金额。

STRUCT<street:STRING,city:STRING>——是一个结构（jié gòu）类型，比如地址有街道和城市。

ROW FORMAT DELIMITED——说明行格式是用分隔符（fēn gé fú）分隔的。

FIELDS TERMINATED BY ','——列和列之间用逗号（dòu hào）分隔。

COLLECTION ITEMS TERMINATED BY '_'——数组里面的每个元素（yuán sù）用下划线（xià huá xiàn）分隔。

MAP KEYS TERMINATED BY ':'——Map的key和value之间用冒号（mào hào）分隔。

STORED AS textfile——数据以文本文件格式存储。

幻灯片34：4.2.3 任务实施 —— 创建外部表

再来创建一个外部表 external_table，注意多了一个 EXTERNAL 关键字和 LOCATION：

【实习命令 / 실습 명령】

```
CREATE EXTERNAL TABLE IF NOT EXISTS
h_sample.external_table(
  staff_id INT, staff_name STRING, salary FLOAT,
  hobby ARRAY<STRING>,
  deductions MAP<STRING, FLOAT>,
  address STRUCT<street:STRING,city:STRING>)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
COLLECTION ITEMS TERMINATED BY '_'
MAP KEYS TERMINATED BY ':'
LINES TERMINATED BY '\n'
STORED AS textfile
LOCATION '/user/hive_external/external_table/'
TBLPROPERTIES('comment'='This is a external table');
```

大家注意：跟内部表相比，外部表多了 EXTERNAL 关键字，还有 LOCATION 指定路径。

幻灯片35-38：4.2.3 任务实施 —— 2. 查看数据表

接下来做查看表的操作。

（1）切换到 h_sample 数据库：

【实习命令 / 실습 명령】

```
USE h_sample;
```

（2）查看所有的表：

【实习命令 / 실습 명령】

```
SHOW TABLES;
```

（3）模糊查询以 m 开头的表：

【实习命令 / 실습 명령】

```
SHOW TABLES LIKE 'm*';
```

（4）查看内部表 managed_table 的基本信息：

【实习命令 / 실습 명령】

```
DESC managed_table;
```

（5）查看外部表 external_table 的详细信息：

【实习命令 / 실습 명령】

```
DESC formatted external_table;
```

幻灯片39-42：4.2.3 任务实施 —— 3. 修改与删除数据表

现在来修改表。

（1）重命名表：把 managed_table 改名为 managed_table_new：

【实习命令 / 실습 명령】

```
ALTER TABLE managed_table RENAME TO managed_table_new;
```

（2）先创建一个新的简单表来做修改练习：

【实习命令 / 실습 명령】

```
CREATE TABLE IF NOT EXISTS alter_managed_table(id STRING, sex STRING, name STRING);
```

（3）修改列名：把 sex 改名为 gender，移动到 name 后面：

【实习命令 / 실습 명령】

```
ALTER TABLE alter_managed_table CHANGE sex gender STRING COMMENT "This is gender"
AFTER name;
```

（4）修改列的数据类型：把 gender 从 STRING 改为 VARCHAR(30)：

【实习命令 / 실습 명령】

```
ALTER TABLE alter_managed_table CHANGE gender gender VARCHAR(30);
```

（5）添加新列 age 和 phone：

【实习命令 / 실습 명령】

```
ALTER TABLE alter_managed_table ADD COLUMNS (age INT COMMENT "This is age", phone
STRING COMMENT "This is phone");
```

（6）替换所有列：

【实习命令 / 실습 명령】

```
ALTER TABLE alter_managed_table REPLACE COLUMNS (username STRING COMMENT "This is
username", passowrd STRING COMMENT "This is password");
```

（7）最后，删除表：

【实习命令 / 실습 명령】

```
DROP TABLE IF EXISTS alter_managed_table PURGE;
```

好，数据表的基本操作我们练习完了。

大家记住：CREATE创建、SHOW查看列表、DESC查看详情、ALTER修改、DROP删除。

项目小结

幻灯片101：项目小结

好，同学们，今天的课程到这里就全部结束了！
我们来回顾（huí gù）一下今天学了什么。

任务4-1：学习了HiveQL操作数据库——创建、查询、查看、切换、修改和删除数据库。
任务4-2：学习了HiveQL操作数据表——创建内部表和外部表、查看、修改和删除表。
任务4-3：学习了HiveQL操作分区表——分区的概念、创建、添加、重命名和删除分区。
任务4-4：学习了HiveQL操作分桶表——分桶的概念、创建分桶表、分桶的好处。
任务4-5：学习了HiveQL操作视图——视图的概念、创建、查看、修改和删除视图。
任务4-6：学习了HiveQL相关操作——索引的概念、创建、查看、重建和删除索引。

今天的内容比较多，但是大部分操作的模式（mó shì）是一样的：
CREATE 创建、SHOW 查看列表、DESCRIBE 查看详情、ALTER 修改、DROP 删除。
掌握（zhǎng wò）了这个规律（guī lǜ），就能举一反三（jǔ yī fǎn sān）了。

课后大家一定要在虚拟机（xū nǐ jī）上动手练习这些命令，光看不练是学不会的！
下次课我们会继续学习Hive的数据操作。
同学们，今天辛苦（xīn kǔ）了，下课！谢谢大家！