基本类型



CREATE TABLE log(

id string COMMENT 'this is id column',

phonenumber bigint,

mac string,

ip string,

url string,

tiele string,

colum1 string,

colum2 string,

colum3 string,

upflow int,

downflow int

)

COMMENT 'this is log table'

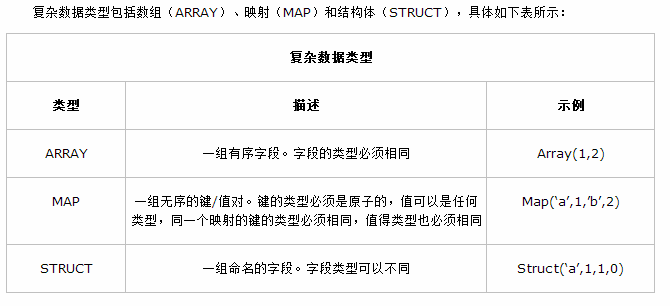
ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'

LINES TERMINATED BY '\n'

stored as textfile

## 复杂类型

在Hive 中如何使用符合数据结构 array，maps ，structs，union



1. Array的使用

创建数据库表，以array作为数据类型

create table person(name string,work\_locations array<string>)

ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'

COLLECTION ITEMS TERMINATED BY ',';

数据

biansutao beijing,shanghai,tianjin,hangzhou

linan changchu,chengdu,wuhan

入库数据

LOAD DATA LOCAL INPATH '/home/hadoop/person.txt' OVERWRITE INTO TABLE person;

查询

hive> select \* from person;

biansutao ["beijing","shanghai","tianjin","hangzhou"]

linan ["changchu","chengdu","wuhan"]

Time taken: 0.355 seconds

hive> select name from person;

linan

biansutao

Time taken: 12.397 seconds

hive> select work\_locations[0] from person;

changchu

beijing

Time taken: 13.214 seconds

hive> select work\_locations from person;

["changchu","chengdu","wuhan"]

["beijing","shanghai","tianjin","hangzhou"]

Time taken: 13.755 seconds

hive> select work\_locations[3] from person;

NULL

hangzhou

Time taken: 12.722 seconds

hive> select work\_locations[4] from person;

NULL

NULL

Time taken: 15.958 seconds

2. Map 的使用

创建数据库表

create table score(name string, score map<string,int>)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

COLLECTION ITEMS TERMINATED BY ','

MAP KEYS TERMINATED BY ':';

要入库的数据

biansutao '数学':80,'语文':89,'英语':95

jobs '语文':60,'数学':80,'英语':99

入库数据

LOAD DATA LOCAL INPATH '/home/hadoop/score.txt' OVERWRITE INTO TABLE score;

查询

hive> select \* from score;

biansutao {"数学":80,"语文":89,"英语":95}

jobs {"语文":60,"数学":80,"英语":99}

Time taken: 0.665 seconds

hive> select name from score;

jobs

biansutao

Time taken: 19.778 seconds

hive> select t.score from score t;

{"语文":60,"数学":80,"英语":99}

{"数学":80,"语文":89,"英语":95}

Time taken: 19.353 seconds

hive> select t.score['语文'] from score t;

60

89

Time taken: 13.054 seconds

hive> select t.score['英语'] from score t;

99

95

Time taken: 13.769 seconds

3 Struct 的使用

创建数据表

CREATE TABLE test(id int,course struct<course:string,score:int>)

ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t'

COLLECTION ITEMS TERMINATED BY ',';

数据

1 english,80

2 math,89

3 chinese,95

入库

LOAD DATA LOCAL INPATH '/home/hadoop/test.txt' OVERWRITE INTO TABLE test;

查询

hive> select \* from test;

OK

1 {"course":"english","score":80}

2 {"course":"math","score":89}

3 {"course":"chinese","score":95}

Time taken: 0.275 seconds

hive> select course from test;

{"course":"english","score":80}

{"course":"math","score":89}

{"course":"chinese","score":95}

Time taken: 44.968 seconds

select t.course.course from test t;

english

math

chinese

Time taken: 15.827 seconds

hive> select t.course.score from test t;

80

89

95

Time taken: 13.235 seconds

4. 数据组合 （不支持组合的复杂数据类型）

LOAD DATA LOCAL INPATH '/home/hadoop/test.txt' OVERWRITE INTO TABLE test;

create table test1(id int,a MAP<STRING,ARRAY<STRING>>)

row format delimited fields terminated by '\t'

collection items terminated by ','

MAP KEYS TERMINATED BY ':';

1 english:80,90,70

2 math:89,78,86

3 chinese:99,100,82

LOAD DATA LOCAL INPATH '/home/hadoop/test1.txt' OVERWRITE INTO TABLE test1;

5.Union类型-不常用

Union类型可以在同一时间点，保持恰好有一个指定的数据类型。

CREATE TABLE union\_test(foo UNIONTYPE<int, double, array<string>, struct<a:int,b:string>>);

0:1

1:2.0

2:"three","four"

3:"a":5,"b":"five"

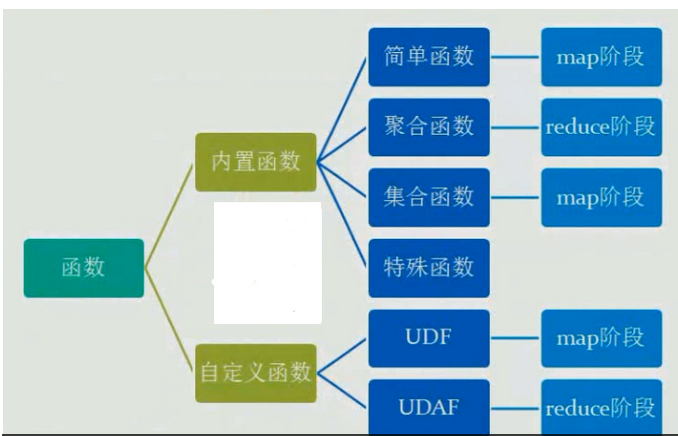
2:"six","seven"

3:"a":8,"b":"eight"

0:9

1:10.0

## 内置函数



hive内置操作符与内置函数

第一部分：关系运算

Hive支持的关系运算符

•常见的关系运算符

•等值比较: =

•不等值比较: <>

•小于比较: <</div>

•小于等于比较: <=

•大于比较: >

•大于等于比较: >=

•空值判断: IS NULL

•非空判断: IS NOT NULL

•LIKE比较: LIKE

•JAVA的LIKE操作: RLIKE

•REGEXP操作: REGEXP

•等值比较: =

语法：A=B

操作类型：所有基本类型

描述: 如果表达式A与表达式B相等，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where 1=1;

•不等值比较: <>

语法: A <> B

操作类型: 所有基本类型

描述:如果表达式A为NULL，或者表达式B为NULL，返回NULL；如果表达式A与表达式B不相等，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where 1 <> 2;

•小于比较: <</div>

语法: A < B

操作类型: 所有基本类型

描述:如果表达式A为NULL，或者表达式B为NULL，返回NULL；如果表达式A小于表达式B，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where 1 < 2;

•小于等于比较: <=

语法: A <= B

操作类型: 所有基本类型

描述:如果表达式A为NULL，或者表达式B为NULL，返回NULL；如果表达式A小于或者等于表达式B，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where 1 <= 1;

•大于等于比较: >=

语法: A >= B

操作类型: 所有基本类型

描述:如果表达式A为NULL，或者表达式B为NULL，返回NULL；如果表达式A大于或者等于表达式B，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where 1 >= 1;

•空值判断: IS NULL

语法: A IS NULL

操作类型: 所有类型

描述: 如果表达式A的值为NULL，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where null is null;

•非空判断: IS NOT NULL

语法: A IS NOT NULL

操作类型: 所有类型

描述: 如果表达式A的值为NULL，则为FALSE；否则为TRUE

举例：

hive> select 1 from dual where 1 is not null;

•LIKE比较: LIKE

语法: A LIKE B

操作类型: strings

描述:如果字符串A或者字符串B为NULL，则返回NULL；如果字符串A符合表达式B 的正则语法，则为TRUE；否则为FALSE。B中字符”\_”表示任意单个字符，而字符”%”表示任意数量的字符。

举例：

hive> select 1 from dual where ‘key' like 'foot%';

1

hive> select 1 from dual where ‘key ' like 'foot\_\_\_\_';

1

注意：否定比较时候用 NOTA LIKE B

hive> select 1 from dual where NOT ‘key ' like 'fff%';

•JAVA的LIKE操作: RLIKE

语法: A RLIKE B

操作类型: strings

描述:如果字符串A或者字符串B为NULL，则返回NULL；如果字符串A符合JAVA正则表达式B的正则语法，则为TRUE；否则为FALSE。

举例：

hive> select 1 from dual where 'footbar’ rlike '^f.\*r$’;

1

注意：判断一个字符串是否全为数字：

hive>select 1 from dual where '123456' rlike '^\\d+$';

1

hive> select 1 from dual where '123456aa' rlike '^\\d+$';

1

•REGEXP操作: REGEXP

语法: A REGEXP B

操作类型: strings

描述: 功能与RLIKE相同

举例：

hive> select 1 from dual where ‘key' REGEXP '^f.\*r$';

1

第二部分：逻辑运算与数学运算

Hive数学运算

•加法操作: +

•减法操作: -

•乘法操作: \*

•除法操作: /

•取余操作: %

•位与操作: &

•位或操作: |

•位异或操作: ^

•位取反操作: ~

•加法操作: +

语法: A + B

操作类型：所有数值类型

说明：返回A与B相加的结果。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。比如，int + int一般结果为int类型，而int + double 一般结果为double类型

举例：

hive> select 1 + 9 from dual;

10

•减法操作: -

语法: A – B

操作类型：所有数值类型

说明：返回A与B相减的结果。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。比如，int – int一般结果为int类型，而int – double 一般结果为double类型

举例：

hive> select 10 – 5 from dual;

5

• 乘法操作 :\*

语法: A \* B

操作类型：所有数值类型

说明：返回A与B相乘的结果。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。注意，如果A乘以B的结果超过默认结果类型的数值范围，则需要通过cast将结果转换成范围更大的数值类型

举例：

hive> select 40 \* 5 from dual;

200

• 除法操作 :/

语法: A / B

操作类型：所有数值类型

说明：返回A除以B的结果。结果的数值类型为double

举例：

hive> select 40 / 5 from dual;

8.0

注意： hive 中最高精度的数据类型是 double, 只精确到小数点后 16 位，在做除法运算的时候要 特别注意

hive>select ceil(28.0/6.999999999999999999999) from dual limit1;

结果为4

hive>select ceil(28.0/6.99999999999999) from dual limit1;

结果为5

• 取余操作 :%

语法: A % B

操作类型：所有数值类型

说明：返回A除以B的余数。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。

举例：

hive> select 41 % 5 from dual;

1

hive> select 8.4 % 4 from dual;

0.40000000000000036

注意：精度在 hive 中是个很大的问题，类似这样的操作最好通过 round 指定精度

hive> select round(8.4 % 4 , 2) from dual;

0.4

• 位与操作 :&

语法: A & B

操作类型：所有数值类型

说明：返回A和B按位进行与操作的结果。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。

举例：

hive> select 4 & 8 from dual;

0

hive> select 6 & 4 from dual;

4

• 位或操作 :|

语法: A | B

操作类型：所有数值类型

说明：返回A和B按位进行或操作的结果。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。

举例：

hive> select 4 | 8 from dual;

12

hive> select 6 | 8 from dual;

14

• 位异或操作 :^

语法: A ^ B

操作类型：所有数值类型

说明：返回A和B按位进行异或操作的结果。结果的数值类型等于A的类型和B的类型的最小父类型（详见数据类型的继承关系）。

举例：

hive> select 4 ^ 8 from dual;

12

hive> select 6 ^ 4 from dual;

2

• 位取反操作 :~

语法: ~A

操作类型：所有数值类型

说明：返回A按位取反操作的结果。结果的数值类型等于A的类型。

举例：

hive> select ~6 from dual;

-7

hive> select ~4 from dual;

-5

Hive逻辑运算

•逻辑与操作: AND

•逻辑或操作: OR

•逻辑非操作: NOT

• 逻辑与操作 :AND

语法: A AND B

操作类型：boolean

说明：如果A和B均为TRUE，则为TRUE；否则为FALSE。如果A为NULL或B为NULL，则为NULL

举例：

hive> select 1 from dual where 1=1 and 2=2;

1

• 逻辑或操作 :OR

语法: A OR B

操作类型：boolean

说明：如果A为TRUE，或者B为TRUE，或者A和B均为TRUE，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where 1=2 or 2=2;

1

• 逻辑非操作 :NOT

语法: NOT A

操作类型：boolean

说明：如果A为FALSE，或者A为NULL，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where not 1=2;

• 逻辑非操作 :NOT

语法: NOT A

操作类型：boolean

说明：如果A为FALSE，或者A为NULL，则为TRUE；否则为FALSE

举例：

hive> select 1 from dual where not 1=2 ;

第三部分：数值运算

•取整函数: round

•指定精度取整函数: round

•向下取整函数: floor

•向上取整函数: ceil

•向上取整函数: ceiling

•取随机数函数: rand

•自然指数函数: exp

•以10为底对数函数: log10

•以2为底对数函数: log2

• 对数函数: log

•幂运算函数: pow

•幂运算函数: power

•开平方函数: sqrt

•二进制函数: bin

•十六进制函数: hex

•反转十六进制函数: unhex

•进制转换函数: conv

•绝对值函数: abs

•正取余函数: pmod

•正弦函数: sin

•反正弦函数: asin

•余弦函数: cos

•反余弦函数: acos

•positive函数: positive

•negative函数: negative

• 取整函数 :round

语法: round(double a)

返回值: BIGINT

说明: 返回double类型的整数值部分 （遵循四舍五入）

举例：

hive> select round(3.1415926) from dual;

3

hive> select round(3.5) from dual;

4

hive> create table dual as select round(9542.158) fromdual;

hive> describe dual;

\_c0 bigint

• 指定精度取整函数 :round

语法: round(double a, int d)

返回值: DOUBLE

说明: 返回指定精度d的double类型

举例：

hive> select round(3.1415926,4) from dual;

3.1416

• 向下取整函数 :floor

语法: floor(double a)

返回值: BIGINT

说明: 返回等于或者小于该double变量的最大的整数

举例：

hive> select floor(3.1415926) from dual;

3

hive> select floor(25) from dual;

25

• 向上取整函数 :ceil

语法: ceil(double a)

返回值: BIGINT

说明: 返回等于或者大于该double变量的最小的整数

举例：

hive> select ceil(3.1415926) from dual;

4

hive> select ceil(46) from dual;

46

• 向上取整函数 :ceiling

语法: ceiling(double a)

返回值: BIGINT

说明: 与ceil功能相同

举例：

hive> select ceiling(3.1415926) from dual;

4

hive> select ceiling(46) from dual;

46

• 取随机数函数 :rand

语法: rand(),rand(int seed)

返回值: double

说明: 返回一个0到1范围内的随机数。如果指定种子seed，则会等到一个稳定的随机数序列

举例：

hive> select rand() from dual;

0.5577432776034763

• 自然指数函数 :exp

语法: exp(double a)

返回值: double

说明: 返回自然对数e的a次方

举例：

hive> select exp(2) from dual;

7.38905609893065

自然对数函数: ln

语法: ln(double a)

返回值: double

说明: 返回a的自然对数

• 以 10 为底对数函数 :log10

语法: log10(double a)

返回值: double

说明: 返回以10为底的a的对数

举例：

hive> select log10(100) from dual;

2.0

• 以 2 为底对数函数 :log2

语法: log2(double a)

返回值: double

说明: 返回以2为底的a的对数

举例：

hive> select log2(8) from dual;

3.0

• 对数函数 :log

语法: log(double base, double a)

返回值: double

说明: 返回以base为底的a的对数

举例：

hive> select log(4,256) from dual;

4.0

• 幂运算函数 :pow

语法: pow(double a, double p)

返回值: double

说明: 返回a的p次幂

举例：

hive> select pow(2,4) from dual;

16.0

• 幂运算函数 :power

语法: power(double a, double p)

返回值: double

说明: 返回a的p次幂,与pow功能相同

举例：

hive> select power(2,4) from dual;

16.0

• 开平方函数 :sqrt

语法: sqrt(double a)

返回值: double

说明: 返回a的平方根

举例：

hive> select sqrt(16) from dual;

4.0

• 二进制函数 :bin

语法: bin(BIGINT a)

返回值: string

说明: 返回a的二进制代码表示

举例：

hive> select bin(7) from dual;

111

• 十六进制函数 :hex

语法: hex(BIGINT a)

返回值: string

说明: 如果变量是int类型，那么返回a的十六进制表示；如果变量是string类型，则返回该字符串的十六进制表示

举例：

hive> select hex(17) from dual;

11

hive> select hex(‘abc’) from dual;

616263

• 反转十六进制函数 :unhex

语法: unhex(string a)

返回值: string

说明: 返回该十六进制字符串所代码的字符串

举例：

hive> select unhex(‘616263’) from dual;

abc

hive> select unhex(‘11’) from dual;

-

hive> select unhex(616263) from dual;

abc

• 进制转换函数 :conv

语法: conv(BIGINT num, int from\_base, int to\_base)

返回值: string

说明: 将数值num从from\_base进制转化到to\_base进制

举例：

hive> select conv(17,10,16) from dual;

11

hive> select conv(17,10,2) from dual;

10001

• 绝对值函数 :abs

语法: abs(double a) abs(inta)

返回值:double int

说明: 返回数值a的绝对值

举例：

hive> select abs(-3.9) from dual;

3.9

hive> select abs(10.9) from dual;

10.9

• 正取余函数 :pmod

语法: pmod(int a, int b),pmod(double a, double b)

返回值: int double

说明: 返回正的a除以b的余数

举例：

hive> select pmod(9,4) from dual;

1

hive> select pmod(-9,4) from dual;

3

• 正弦函数 :sin

语法: sin(double a)

返回值: double

说明: 返回a的正弦值

举例：

hive> select sin(0.8) from dual;

0.7173560908995228

• 反正弦函数 :asin

语法: asin(double a)

返回值: double

说明: 返回a的反正弦值

举例：

hive> select asin(0.7173560908995228) from dual;

0.8

• 余弦函数 :cos

语法: cos(double a)

返回值: double

说明: 返回a的余弦值

举例：

hive> select cos(0.9) from dual;

0.6216099682706644

• 反余弦函数 :acos

语法: acos(double a)

返回值: double

说明: 返回a的反余弦值

举例：

hive> select acos(0.6216099682706644) from dual;

0.9

• positive 函数 :positive

语法: positive(int a), positive(double a)

返回值: int double

说明: 返回a

举例：

hive> select positive(-10) from dual;

-10

hive> select positive(12) from dual;

12

• negative 函数 :negative

语法: negative(int a), negative(double a)

返回值: int double

说明: 返回-a

举例：

hive> select negative(-5) from dual;

5

hive> select negative(8) from dual;

-8

第四部分：日期函数

•UNIX时间戳转日期函数: from\_unixtime

• 获取当前UNIX时间戳函数: unix\_timestamp

•日期转UNIX时间戳函数: unix\_timestamp

• 指定格式日期转UNIX时间戳函数: unix\_timestamp

•日期时间转日期函数: to\_date

•日期转年函数: year

• 日期转月函数: month

• 日期转天函数: day

• 日期转小时函数: hour

• 日期转分钟函数: minute

• 日期转秒函数: second

• 日期转周函数: weekofyear

• 日期比较函数: datediff

• 日期增加函数: date\_add

• 日期减少函数: date\_sub

• UNIX 时间戳转日期函数 :from\_unixtime

语法: from\_unixtime(bigint unixtime[, string format])

返回值: string

说明: 转化UNIX时间戳（从1970-01-01 00:00:00 UTC到指定时间的秒数）到当前时区的时间格式

举例：

hive> select from\_unixtime(1323308943,'yyyyMMdd') fromdual;

20111208

• 获取当前 UNIX 时间戳函数 :unix\_timestamp

语法: unix\_timestamp()

返回值: bigint

说明: 获得当前时区的UNIX时间戳

举例：

hive> select unix\_timestamp() from dual;

1323309615

• 日期转 UNIX 时间戳函数 :unix\_timestamp

语法: unix\_timestamp(string date)

返回值: bigint

说明: 转换格式为"yyyy-MM-dd HH:mm:ss"的日期到UNIX时间戳。如果转化失败，则返回0。

举例：

hive> select unix\_timestamp('2011-12-07 13:01:03') fromdual;

1323234063

• 指定格式日期转 UNIX 时间戳函数 :unix\_timestamp

语法: unix\_timestamp(string date, string pattern)

返回值: bigint

说明: 转换pattern格式的日期到UNIX时间戳。如果转化失败，则返回0。

举例：

hive> select unix\_timestamp('20111207 13:01:03','yyyyMMddHH:mm:ss') from dual;

1323234063

• 日期时间转日期函数 :to\_date

语法: to\_date(string timestamp)

返回值: string

说明: 返回日期时间字段中的日期部分。

举例：

hive> select to\_date('2011-12-08 10:03:01') from dual;

2011-12-08

• 日期转年函数 :year

语法: year(string date)

返回值: int

说明: 返回日期中的年。

举例：

hive> select year('2011-12-08 10:03:01') from dual;

2011

hive> select year('2012-12-08') from dual;

2012

• 日期转月函数 :month

语法: month (string date)

返回值: int

说明: 返回日期中的月份。

举例：

hive> select month('2011-12-08 10:03:01') from dual;

12

hive> select month('2011-08-08') from dual;

8

• 日期转天函数 :day

语法: day (string date)

返回值: int

说明: 返回日期中的天。

举例：

hive> select day('2011-12-08 10:03:01') from dual;

8

hive> select day('2011-12-24') from dual;

24

• 日期转小时函数 :hour

语法: hour (string date)

返回值: int

说明: 返回日期中的小时。

举例：

hive> select hour('2011-12-08 10:03:01') from dual;

10

• 日期转分钟函数 :minute

语法: minute (string date)

返回值: int

说明: 返回日期中的分钟。

举例：

hive> select minute('2011-12-08 10:03:01') from dual;

3

• 日期转秒函数 :second

语法: second (string date)

返回值: int

说明: 返回日期中的秒。

举例：

hive> select second('2011-12-08 10:03:01') from dual;

1

• 日期转周函数 :weekofyear

语法: weekofyear (string date)

返回值: int

说明: 返回日期在当前的周数。

举例：

hive> select weekofyear('2011-12-08 10:03:01') from dual;

49

• 日期比较函数 :datediff

语法: datediff(string enddate, string startdate)

返回值: int

说明: 返回结束日期减去开始日期的天数。

举例：

hive> select datediff('2012-12-08','2012-05-09') fromdual;

213

• 日期增加函数 :date\_add

语法: date\_add(string startdate, int days)

返回值: string

说明: 返回开始日期startdate增加days天后的日期。

举例：

hive> select date\_add('2012-12-08',10) from dual;

2012-12-18

• 日期减少函数 :date\_sub

语法: date\_sub (string startdate, int days)

返回值: string

说明: 返回开始日期startdate减少days天后的日期。

举例：

hive> select date\_sub('2012-12-08',10) from dual;

2012-11-28

第五部分：条件函数

•If函数: if

•非空查找函数: COALESCE

•条件判断函数：CASE

• If 函数 :if

语法: if(boolean testCondition, T valueTrue, TvalueFalseOrNull)

返回值: T

说明: 当条件testCondition为TRUE时，返回valueTrue；否则返回valueFalseOrNull

举例：

hive> select if(1=2,100,200) from dual;

200

hive> select if(1=1,100,200) from dual;

100

• 非空查找函数 :COALESCE

语法: COALESCE(T v1, T v2, …)

返回值: T

说明: 返回参数中的第一个非空值；如果所有值都为NULL，那么返回NULL

举例：

hive> select COALESCE(null,'100','50′) from dual;

100

条件判断函数： CASE

语法 : CASEa WHEN b THEN c [WHEN d THEN e]\* [ELSE f] END

返回值 :T

说明：如果 a 等于 b ，那么返回 c ；如果 a 等于 d ，那么返回 e ；否则返回 f

举例：

hive> Select case 100 when 50 then 'tom' when 100 then'mary' else 'tim' end fromdual;

mary

第六部分：字符串函数

•字符串长度函数：length

•字符串反转函数：reverse

•字符串连接函数：concat

• 带分隔符字符串连接函数：concat\_ws

• 字符串截取函数：substr,substring

• 字符串截取函数：substr,substring

• 字符串转大写函数：upper,ucase

• 字符串转小写函数：lower,lcase

• 去空格函数：trim

• 左边去空格函数：ltrim

• 右边去空格函数：rtrim

•正则表达式替换函数：regexp\_replace

•正则表达式解析函数：regexp\_extract

•URL解析函数：parse\_url

•json解析函数：get\_json\_object

•空格字符串函数：space

•重复字符串函数：repeat

•首字符ascii函数：ascii

•左补足函数：lpad

•右补足函数：rpad

•分割字符串函数: split

•集合查找函数: find\_in\_set

• 字符串长度函数： length

语法: length(string A)

返回值: int

说明：返回字符串A的长度

举例：

hive> select length('abcedfg') from dual;

7

2.其他练习

工作中用到了一些，总结一下：

　　1、 行拆分、合并

　　1． 分拆explode(ARRAY)

　　返回值：多行

　　SELECT explode(myCol) AS myNewCol FROM myTable;

　　注：1.使用UDTF函数时，select中不可以包含其他表达式；

　　2.UDTF函数不能嵌套使用；

　　3.UDTF不支持GROUP BY / CLUSTER BY / DISTRIBUTE BY / SORT BY；

　　2． 合并去重collect\_set(col)

　　返回值：数组（去重的效果）

　　Xjt2原始数据：

　　1 one

　　2 two

　　2 two

　　2 two

　　3 three

　　4 four

　　4 four

　　1 one one

　　1 one two

　　2 twotwo

　　2 twoone

　　select id,collect\_set(name) from xjt2 group by id;

　　1 ["one two","oneone","one"]

　　2 ["twoone","twotwo","two"]

　　3 ["three"]

　　4 ["four"]

　　select collect\_set(name) from xjt2;

　　["three","one","four","two"]

　　注：当collect\_set(col)与其他字段同时在select语句中时，必须使用group by other\_fields；

　　2、 时间函数

　　1. 获取当前Unix时间戳unix\_timestamp()

　　返回值类型：BIGINT

　　select unix\_timestamp() from xjt1;

　　1383012276

　　2. 将日期转时间戳unix\_timestamp(string date)

　　返回值类型：BIGINT，若转换失败，则返回0

　　select unix\_timestamp('2013-01-13 00:00:00') from xjt1;

　　1358006400

　　3. 转化指定格式（pattern）日期转时间戳unix\_timestamp(string date, string pattern)

　　返回值类型：BIGINT，若转化失败，则返回0

　　select unix\_timestamp('2013-01-13 00:00:00','yyyyMMdd')from xjt1;

　　1354291200

　　4. 将Unix时间戳转日期from\_unixtime(BIGINT,’format’)

　　select from\_unix(unix\_timestamp(),'yyyyMMdd') from xjt1;

　　20131029

　　5. 取日期to\_date()、取年year()、取月month()、取天数day()

　　返回值类型：SRING

　　select to\_date('1990-10-10 00:00:00') from xjt1;

　　1990-10-10

　　6. 日期增加函数date\_add(string startdate, int days)

　　返回值类型：STRING

　　select date\_add('2013-10-29',10) from xjt1;

　　2013-11-08

　　7. 日期减少函数date\_sub(string startdate, intdays)

　　返回值类型：STRING

　　select date\_sub('2013-10-29',10) from xjt1;

　　2013-10-19

　　8. 日期比较函数datediff(string enddate, string startdate)

　　返回值类型：INT（结束日期减去开始日期，结束日期放在前面）

　　select datediff('2013-10-29','2013-12-10') from xjt1;

　　-42

　　3、 条件判断函数CASE

　　返回值：T/F

　　语法：CASE a WHEN b THENc [WHEN d THEN e]\* [ELSE f] END

　　说明：如果 a 等于 b ，那么返回 c ；如果 a 等于 d ，那么返回 e ；否则返回 f

　　4、 字符串分割函数split(stringstr, string pat)

　　返回值类型：ARRAY

　　selectsplit('hello world hello hive',' ') from xjt1;

　　["hello","world","hello","hive"]