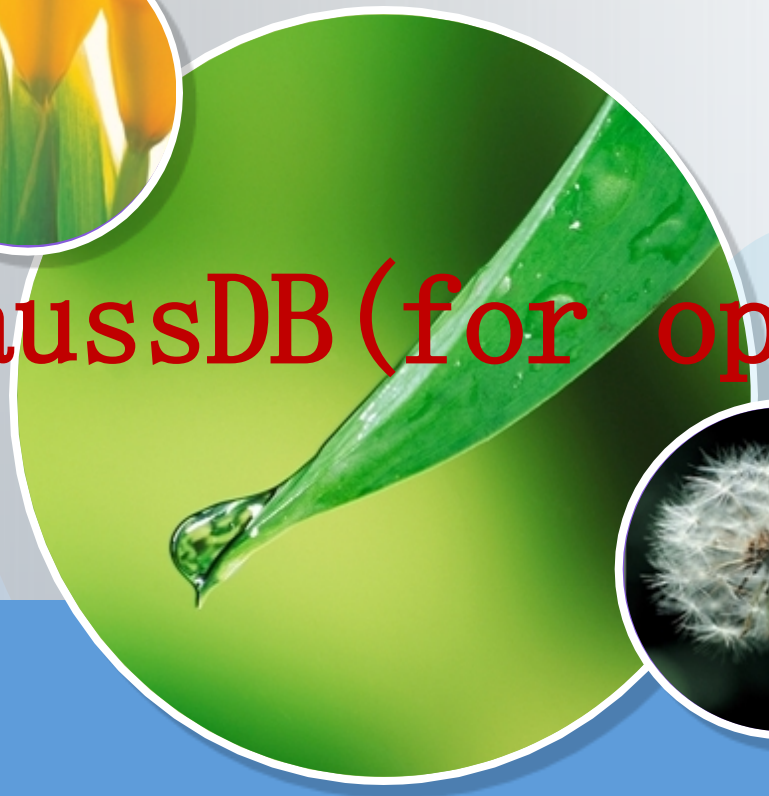





GaussDB(for openGauss) ——————

存储过程与触发器

存储过程

存储过程

- 存储过程(Stored Procedure) 是在数据库服务器 执行的一组Transact-SQL语句的集合， 经编译后存 放在数据库服务器端，是数据库中运用得十分 广泛的一种数据库对象。
- 存储过程作为一个单元进行处理并以一个名称 来标识。它能够向用户返回数据，也可向数据 库中写入或修改数据，还可以执行系统函数和 管理操作，用户在编程中只需要给出存储过程 的名称和必须的参数，便可以方便地调用它们。
- 执行速度快、效率高。存储过程在运行存储过程时不需要再对存储过程进行编译，从而加快执行的速度。
- 存储过程在创建完毕之后，可以在程序中被 多次调用，而不必重新编写该Transact-SQL语句。

存储过程

存储过程

- 如果创建存储过程时参数或返回值带有精度，不进行精度检测。
- 创建存储过程时，存储过程定义中对表对象的操作建议都显示指定模式，否则可能会导致存储过程执行异常。
- 在创建存储过程时，存储过程内部通过SET语句设置current_schema和search_path无效。执行完函数search_path和current_schema与执行函数前的search_path和current_schema保持一致。
- 如果存储过程参数中带有出参，SELECT调用存储过程必须缺省出参，CALL调用存储过程时调用非重载函数必须指定出参，对于重载的package函数，out参数可以缺省。

存储过程

存储过程

- 存储过程指定package属性时支持重载。
- 在创建procedure时，不能在avg函数外面嵌套其他agg函数，或者其他系统函数。
- 函数定义时如果指定为IMMUTABLE和SHIPPABLE类型，应该尽量避免函数中存在INSERT，UPDATE，DELETE，MERGE和DDL操作，因为上述操作应该由CN判断对应的执行节点，否则执行结果可能产生错误。
- 存储过程中不支持需要return集合的操作。

存储过程

语法格式

```
postgres=# CREATE [ OR REPLACE ] PROCEDURE procedure_name
   [ ( {[ argmode ] [ argname ] argtype [ { DEFAULT | := | = } expression ]}[,...]) ]
   [
     { IMMUTABLE | STABLE | VOLATILE }
     | { SHIPPABLE | NOT SHIPPABLE }
     | {PACKAGE}
     | [ NOT ] LEAKPROOF
     | { CALLED ON NULL INPUT | RETURNS NULL ON NULL INPUT | STRICT }
     | {[ EXTERNAL ] SECURITY INVOKER | [ EXTERNAL ] SECURITY DEFINER | AUTHID DEFINER | AUTHID
CURRENT_USER}
     | COST execution_cost
     | SET configuration_parameter { [ TO | = ] value | FROM CURRENT }
   ][ ... ]
 { IS | AS }
plsql_body
/
```

存储过程

语法格式

参数说明

OR REPLACE

当存在同名的存储过程时，替换原来的定义。

procedure_name

创建的存储过程名称，可以带有模式名。

取值范围：字符串，要符合标识符的命名规范。

argmode

参数的模式。

须知：

VARIADIC用于声明数组类型的参数。

取值范围： IN，OUT，INOUT或VARIADIC。缺省值是IN。只有OUT模式的参数后面能跟VARIADIC。并且OUT和INOUT模式的参数不能用在RETURNS TABLE的过程定义中。

argname

参数的名称。

取值范围：字符串，要符合标识符的命名规范。

argtype

参数的数据类型。取值范围：可用的数据类型。

IMMUTABLE、STABLE等

行为约束可选项。

plsql_body

PL/SQL存储过程体。

创建存储过程

❖ **实验步骤：**

1.创建存储过程：在全国各省累计数据统计表中增加一条记录。执行存储过程：增加2021年10月8日吉林省累计确诊578例，累计治愈571例，累计死亡3例。

2.定义存储过程：查询美国指定州指定日期的新冠肺炎累计确诊总数与累计死亡总数。通过该存储过程统计California州截至2021年1月1日的新冠疫情数据情况。

3.调用存储过程。

管理存储过程

- 管理存储过程，切换到 库管理 -> 对象列表，选择 存储过程 ， 选择 insert_data 存储过程中的操作，单击查看存储过程详情：

Schema列表 | 对象列表 | 元数据采集

对象列表数据来自实时查询(最多显示10000条)，对您的数据库有一定的性能消耗 立即采集

表
视图
存储过程
触发器
序列

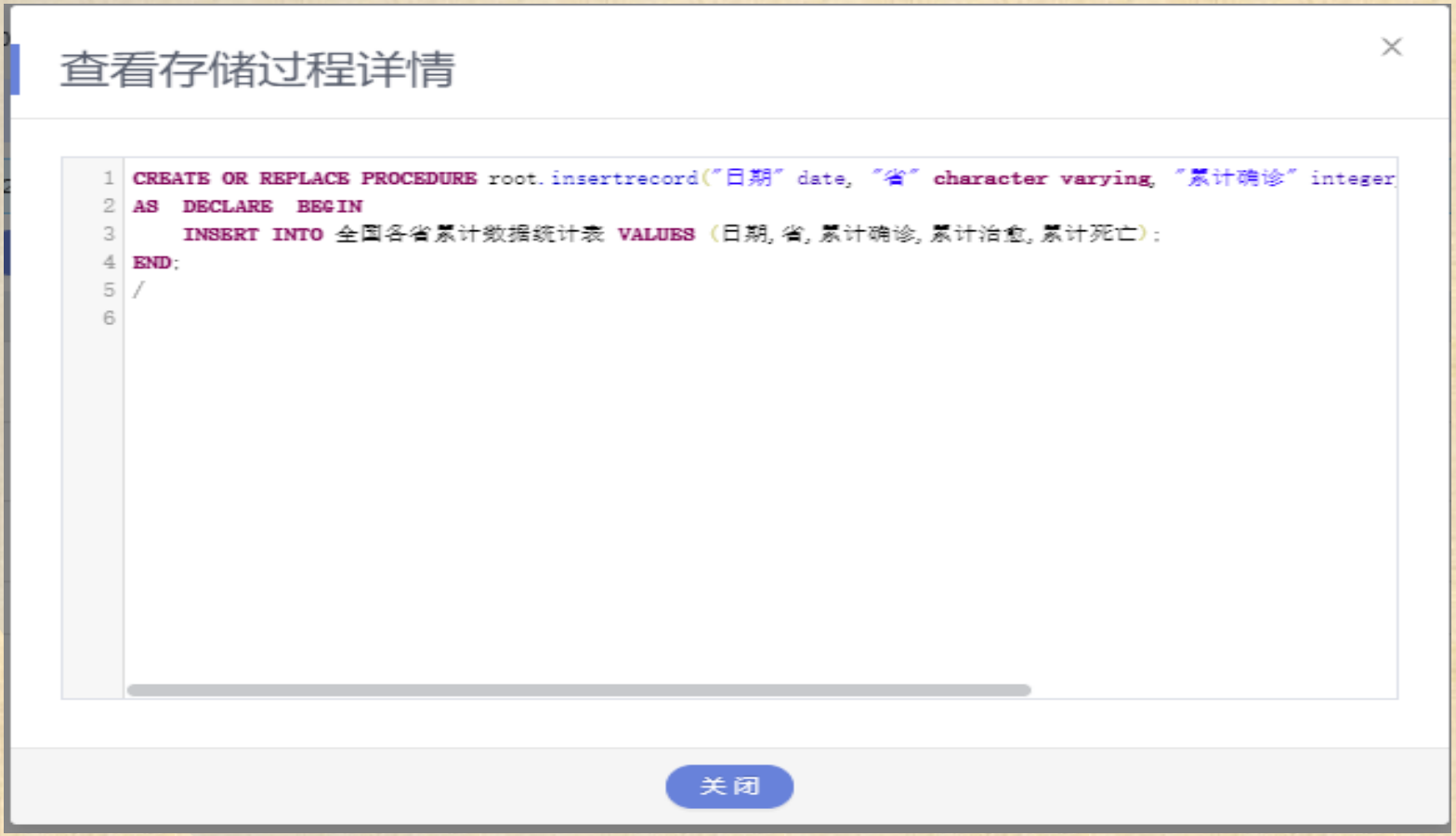
Schema: root ＋新建存储过程

存储过程名称	参数	操作
get_statu	(IN in_zhou varchar,IN in_date date,OUT 州累计死亡 int8,OUT 周累计确诊 int8)	修改或执行 | 删除存储过程 | 查看存储过程详情
insertrecord	(IN 日期 date,IN 省 varchar,IN 累计确诊 int4,IN 累计治愈 int4,IN 累计死亡 int4)	修改或执行 | 删除存储过程 | 查看存储过程详情
us_insert_trigger_fun	()	修改或执行 | 删除存储过程 | 查看存储过程详情

10 条/页 总条数: 3 1

管理存储过程

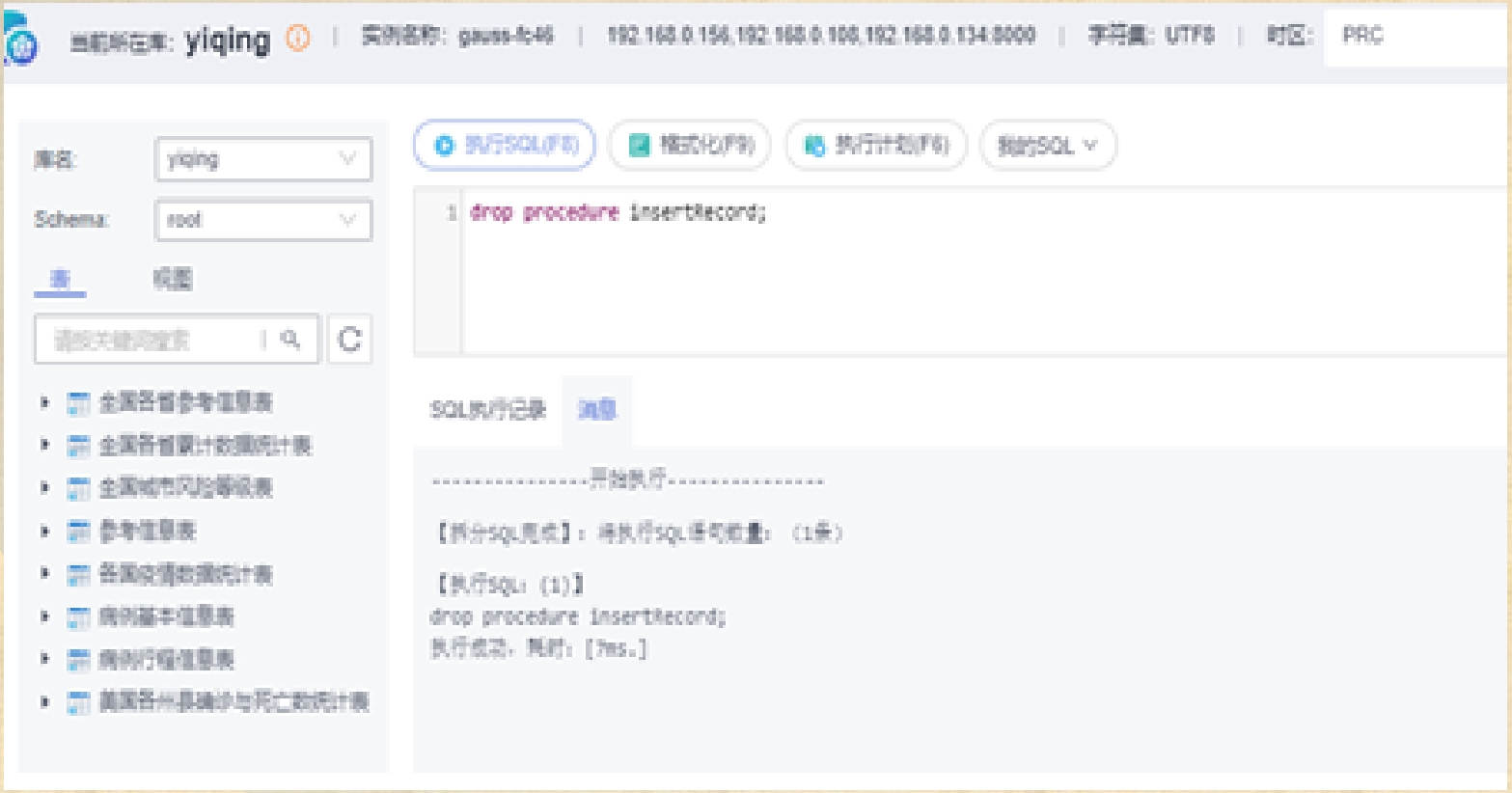
查看/调用存储过程详情：



删除存储过程，命令如下：drop procedure 过程名;

管理存储过程

删除存储过程，命令如下：drop procedure insertrecord;



创建和管理触发器

触发器是将与指定的表或视图关联，并在特定条件下执行指定的函数。

注意事项

- GAUSSDB当前仅支持在普通行存表上创建触发器，不支持在列存表、临时表、unlogged表等类型表上创建触发器。
- 如果为同一事件定义了多个相同类型的触发器，则按触发器的名称字母顺序触发它们。
- 触发器常用于多表间数据关联同步场景，对SQL执行性能影响较大，不建议在大数据量同步及对性能要求高的场景中使用。
- 当触发器满足如下条件时，触发语句能和触发器一起下推到DN执行并提升触发器执行性能：
- 源表触发器使用的触发器函数为plpgsql类型（推荐类型）。
- 源表与触发表分布键的类型、数量完全相同，均为行存表，且所属相同的nodegroup。
- 原INSERT/UPDATE/DELETE语句条件中包含所有分布键与NEW/OLD等值比较表达式。
- 原INSERT/UPDATE/DELETE语句在没有触发器的情况下原本就能query shipping。
- 源表上只有INSERT BEFORE FOR EACH ROW/INSERT AFTER FOR EACH ROW/UPDATE BEFORE FOR EACH ROW/UPDATE AFTER FOR EACH ROW/DELETE BEFORE FOR EACH ROW/DELETE AFTER FOR EACH ROW六类触发器，且所有触发器都可下推。
- INSERT ON DUPLICATE KEY UPDATE语句无法触发触发器。

创建和管理触发器

语法格式:

```
CREATE [ CONSTRAINT ] TRIGGER trigger_name { BEFORE | AFTER | INSTEAD OF } { event [
OR ... ] }
    ON table_name
    [ FROM referenced_table_name ]
    { NOT DEFERRABLE | [ DEFERRABLE ] { INITIALLY IMMEDIATE | INITIALLY DEFERRED } }
    [ FOR [ EACH ] { ROW | STATEMENT } ]
    [ WHEN ( condition ) ]
    EXECUTE PROCEDURE function_name ( arguments );
```

其中event包含以下几种:

```
    INSERT
    UPDATE [ OF column_name [, ... ] ]
    DELETE
    TRUNCATE
```

创建和管理触发器

语法格式

参数说明

CONSTRAINT

可选项，指定此参数将创建约束触发器，即触发器作为约束来使用。除了可以使用SET CONSTRAINTS调整触发器触发的时间之外，这与常规触发器相同。 约束触发器必须是AFTER ROW触发器。

trigger_name

触发器名称，该名称不能限定模式，因为触发器自动继承其所在表的模式，且同一个表的触发器不能重名。 对于约束触发器，使用13.14.143 修改触发器行为时也使用此名称。

取值范围：符合标识符命名规范的字符串，且最大长度不超过63个字符。

BEFORE

触发器函数是在触发事件发生前执行。

AFTER

触发器函数是在触发事件发生后执行，约束触发器只能指定为AFTER。

INSTEAD OF

触发器函数直接替代触发事件。

创建和管理触发器

语法格式

event

启动触发器的事件，取值范围包括：INSERT、UPDATE、DELETE或TRUNCATE，也可以通过OR同时指定多个触发事件。

对于UPDATE事件类型，可以使用下面语法指定列：

UPDATE OF column_name1 [, column_name2 ...]

表示只有这些列作为UPDATE语句的目标列时，才会启动触发器，但是INSTEAD OF UPDATE类型不支持指定列信息。

table_name

需要创建触发器的表名称。

取值范围：数据库中已经存在的表名称。

referenced_table_name

约束引用的另一个表的名称。 只能为约束触发器指定，常见于外键约束。由于当前不支持外键，因此不建议使用。

取值范围：数据库中已经存在的表名称。

DEFERRABLE | NOT DEFERRABLE

约束触发器的启动时机，仅作用于约束触发器。这两个关键字设置该约束是否可推迟。

INITIALLY IMMEDIATE | INITIALLY DEFERRED

如果约束是可推迟的，则这个子句声明检查约束的缺省时间，仅作用于约束触发器。

创建和管理触发器

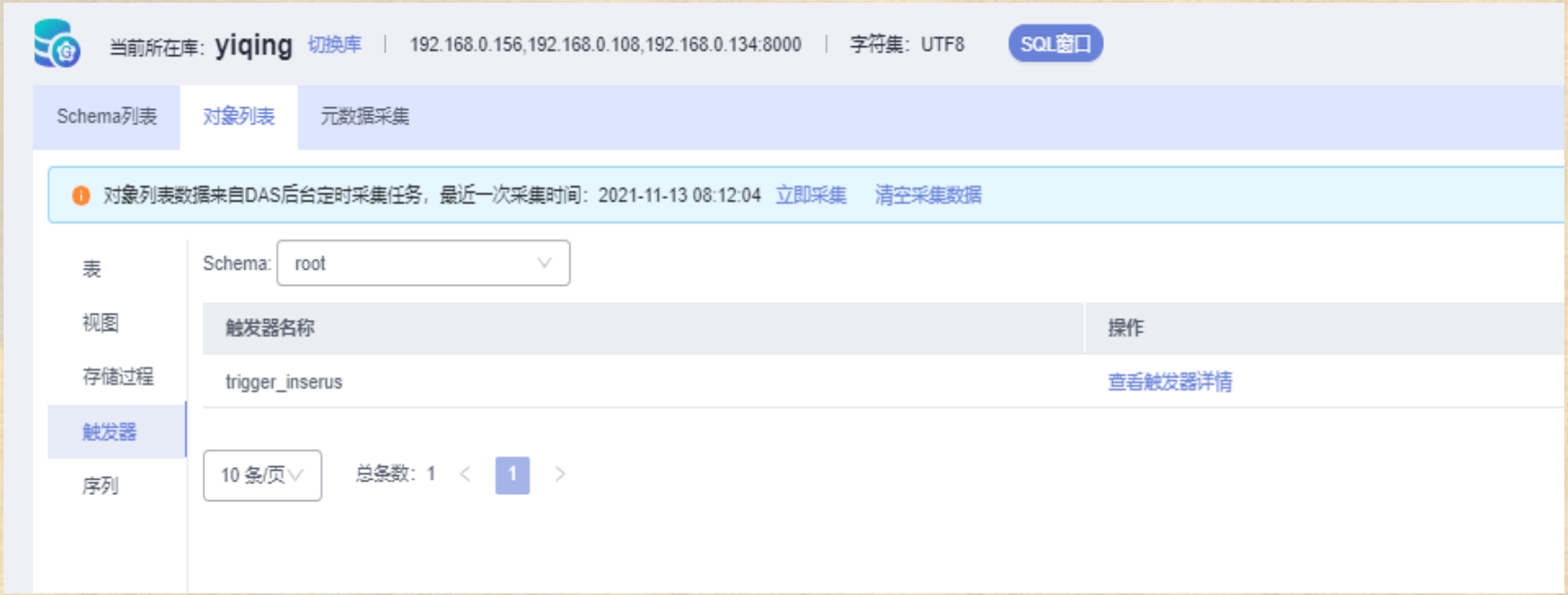
语法格式

INITIALLY IMMEDIATE | INITIALLY DEFERRED
如果约束是可推迟的，则这个子句声明检查约束的缺省时间，仅作用于约束触发器。
详细介绍请参见13.14.69 。
FOR EACH ROW | FOR EACH STATEMENT
触发器的触发频率。
FOR EACH ROW是指该触发器是受触发事件影响的每一行触发一次。
FOR EACH STATEMENT是指该触发器是每个SQL语句只触发一次。
未指定时默认值为FOR EACH STATEMENT。约束触发器只能指定为FOR EACH ROW。
condition
决定是否实际执行触发器函数的条件表达式。当指定WHEN时，只有在条件返回true时才会调用该函数。
在FOR EACH ROW触发器中，WHEN条件可以通过分别写入OLD.column_name或NEW.column_name来引用旧行或新行值的列。 当然，INSERT触发器不能引用OLD和DELETE触发器不能引用NEW。
INSTEAD OF触发器不支持WHEN条件。
WHEN表达式不能包含子查询。
对于约束触发器，WHEN条件的评估不会延迟，而是在执行更新操作后立即发生。 如果条件返回值不为true，则触发器不会排队等待延迟执行。
function_name
用户定义的函数，必须声明为不带参数并返回类型为触发器，在触发器触发时执行。
arguments
执行触发器时要提供给函数的可选的以逗号分隔的参数列表。参数是文字字符串常量，简单的名称和数字常量也可以写在这里，但它们都将被转换为字符串。 请检查触发器函数的实现语言的描述，以了解如何在函数内访问这些参数。
关于触发器种类：
说明：
INSTEAD OF的触发器必须标记为FOR EACH ROW，并且只能在视图上定义。
BEFORE和AFTER触发器作用在视图上时，只能标记为FOR EACH STATEMENT。
TRUNCATE类型触发器仅限FOR EACH STATEMENT。

创建和管理触发器

❖ 实验步骤：

1. 创建INSERT触发器，向美国各州县确诊与死亡数统计表中插入记录时，检查该记录的州县在参考信息表中是否存在。如果不存在，则不允许插入。



创建和管理触发器

- 实验步骤：

查看触发器详情

查看触发器详情

```
1 CREATE TRIGGER trigger_inserus BEFORE INSERT ON root."美国各州县确诊与死亡数统计表" FOR EACH ROW
```

管理触发器

❖ 实验步骤

2） 创建DELETE触发器，当从病例基本信息表中删除一条记录时，该病例ID对应的行程信息记录也进行删除操作。

3）创建UPDATE触发器，禁止修改全国各省累计数据统计表中的累计确诊、累计治愈和累计死亡数据。

管理触发器

❖ 实验步骤

4）验证INSERT触发器、DELETE触发器、UPDATE触发器；

执行SQL(F8)　格式化(F9)　执行计划(F6)　我的SQL

```
1 select * FROM 美国各州县确诊与死亡数统计表
2 INSERT INTO 美国各州县确诊与死亡数统计表 VALUES ('2021-10-19','US','AAA','BBB',123,1);
3
```

SQL执行记录　消息

```
----------------开始执行----------------

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：(1)】
INSERT INTO 美国各州县确诊与死亡数统计表 VALUES ('2021-10-19','US','AAA','BBB',123,1);
执行失败，失败原因：ERROR: 未通过数据一致性验证!
```

管理触发器

❖ 实验步骤

5）删除INSERT触发器、DELETE触发器、UPDATE触发器。

执行SQL(F8)　格式化(F9)　执行计划(F6)　我的SQL

```
1 select * FROM 美国各州县确诊与死亡数统计表
2 INSERT INTO 美国各州县确诊与死亡数统计表 VALUES ('2021-10-19','US','AAA','BBB',123,1);
3
```

SQL执行记录　消息

----------开始执行----------

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：(1)】
INSERT INTO 美国各州县确诊与死亡数统计表 VALUES ('2021-10-19','US','AAA','BBB',123,1);
执行失败，失败原因：ERROR: 未通过数据一致性验证!

谢谢！！！