

Oracle11新特性虚拟列 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/472/2021_2022_Oracle11_E6_96_c67_472313.htm

对于虚拟列又有一点新的研究。虽然虚拟列不能参考其他的虚拟列，但是可以通过其他的方法来变相实现：
SQL> CREATE TABLE T_V_COL_P 2 (3 ID NUMBER PRIMARY KEY, 4 NAME VARCHAR2(30), 5 V_NAME AS (LOWER(NAME)), 6 V_COL AS (LENGTH(V_NAME)) 7).

V_NAME AS (LOWER(NAME)), *第 5 行出现错误:

ORA-54012: 在列表表达式中引用了虚拟列 由于虚拟列并不存储数据，而且数据的生成是在查询的时候，因此可以先建立表，然后将表的主键作为参数传递给虚拟列参考的函数：

SQL> CREATE OR REPLACE FUNCTION F_TEST_VIRTUAL (P_IN NUMBER) RETURN NUMBER DETERMINISTIC AS 2 BEGIN 3 FOR I IN (SELECT LENGTH(V_NAME) LEN FROM

T_V_COL_P WHERE ID = P_IN) LOOP 4 RETURN I.LEN. 5

END LOOP. 6 END. 7 / 函数已创建。 SQL> INSERT INTO

T_V_COL_P (ID, NAME) VALUES (1, ' TABLE '). 已创建 1

行。 SQL> INSERT INTO T_V_COL_P (ID, NAME) VALUES

(2, ' INDEX '). 已创建 1 行。 SQL> SELECT * FROM

T_V_COL_P. ID NAME V_NAME V_COL -----

----- 1 TABLE table 5 2 INDEX index 5 通过这个方法

，就可以实现虚拟列参考其他的虚拟列。当前，由于虚拟列的数值本身就来自其他的实际列，因此虚拟列参考虚拟列的意义不大。但是上面给出的方法还是很有意义的，可以利用

这个方法实现很多的功能。 举个简单的例子，刚才建立的是主表，有一个子表引用主表：SQL> CREATE TABLE T_V_COL_F 2 (3 ID NUMBER, 4 FID NUMBER, 5 NAME VARCHAR2(30), 6 FOREIGN KEY (FID) REFERENCES T_V_COL_P 7). 表已创建。 SQL> INSERT INTO T_V_COL_F SELECT 100000 ROWNUM, 1, TABLE_NAME FROM DBA_TABLES. 已创建2493行。 SQL> INSERT INTO T_V_COL_F SELECT 200000 ROWNUM, 2, INDEX_NAME FROM DBA_INDEXES. 已创建3945行。 如果想查询主表记录的同时查询参考当前主表ID的子表记录数：SQL> SELECT ID, NAME, V_NAME, (SELECT COUNT(*) FROM T_V_COL_F WHERE FID = A.ID) NUM 2 FROM T_V_COL_P A. ID NAME V_NAME NUM -----

----- 1 TABLE table 2493 2 INDEX index 3945 这是常规的写法，而使用虚拟列可以在一张表上实现这个功能：SQL> CREATE OR REPLACE FUNCTION F_TEST_VIRTUAL (P_IN NUMBER) RETURN NUMBER DETERMINISTIC AS 2 BEGIN 3 FOR I IN (SELECT COUNT(*) NUM FROM T_V_COL_F WHERE FID = P_IN) LOOP 4 RETURN I.NUM. 5 END LOOP. 6 END. 7 / 函数已创建。 SQL> SELECT * FROM T_V_COL_P. ID NAME V_NAME V_COL -----

----- 1 TABLE table 2493 2 INDEX index 3945 采用这种方法可以简化很多的问题，而且如果不访问虚拟列，并不会引发对子表的访问。这种方法唯一需要注意一点，不要造成循环引用：SQL> CREATE OR

