

Oracle中使用SQLMODEL定义行间计算（2）PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/143/2021_2022_Oracle_E4_B8_AD_E4_c102_143537.htm 4.现在，您可以创建 SALES_VIEW 视图。从 SQL*Plus 会话中，执行以下脚本:

@c:\wkdir\sample_data.sql sample_data.sql 脚本包含以下内容:

```
CREATE VIEW sales_view AS SELECT country_name country,
prod_name prod, calendar_year year, SUM(amount_sold) sale,
COUNT(amount_sold) cnt FROM sales, times, customers,
countries, products WHERE sales.time_id = times.time_id
AND sales.prod_id = products.prod_id AND sales.cust_id =
customers.cust_id AND customers.country_id =
```

```
countries.country_id GROUP BY country_name, prod_name,
calendar_year/5.验证视图创建正确，并具有 3219 行。从
```

SQL*Plus 会话中，执行以下脚本: @c:\wkdir\sel_sv.sql sel_sv.sql

脚本包含以下内容: SELECT COUNT(*) FROM sales_view.6.要使性能最佳化，系统应当已经具有基于以上视图使用的数据构建的物化视图。该物化视图是在 SH 模式数据的安装期间

创建的。Oracle 的摘要管理系统将使用上述视图自动重写所有查询，以便利用该物化视图。查看示例语法作为模型的初始示例，请考虑以下语句: SELECT SUBSTR(country,1,20)

```
country, SUBSTR(prod,1,15) prod, year, sales FROM
```

```
sales_view WHERE country IN (Italy,Japan) MODEL RETURN
```

```
UPDATED ROWS PARTITION BY (country) DIMENSION BY
```

```
(prod, year) MEASURES (sale sales) RULES ( sales[Bounce, 2002]
```

```
= sales[Bounce, 2001] sales[Bounce, 2000], sales[Y Box, 2002] =
```

sales[Y Box, 2001], sales[2_Products, 2002] = sales[Bounce, 2002]
sales[Y Box, 2002])ORDER BY country, prod, year.结果是:
COUNTRY PROD YEAR SALES-----

-----Italy 2_Products 2002
90387.54Italy Bounce 2002 9179.99Italy Y Box 2002 81207.55Japan
2_Products 2002 101071.96Japan Bounce 2002 11437.13Japan Y Box
2002 89634.83

因为该语句按照国家/地区划分，所以这些规则一次应用于一个国家/地区的数据。请注意，数据结束于2001年，因此为2002年或之后年份定义值的任何规则都将插入新单元格。第一个规则将2002年Bounce的销售定义为2000年和2001年的销售总和。第二个规则将2002年Y Box的销售定义为2001年的销售值。第三个规则定义了一个名为

2_Products的类别，它是2002年的Bounce与Y Box值相加所得的总和。请注意，2_Products的值派生自前两个规则的结果，因此这两个规则必须在2_Products规则之前执行。语法准则 请注意，MODEL关键字后面的RETURN UPDATED ROWS子句将结果限制为在该查询中创建或更新的那些行。使用该子句是使结果集只包含新计算的值的简便方法。在整个示例中，都将用到RETURN UPDATED ROWS子句。

100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com