

Oracle数据库中COALESCE函数使用详解 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/166/2021\\_2022\\_Oracle\\_E6\\_95\\_B0\\_E6\\_c102\\_166975.htm](https://www.100test.com/kao_ti2020/166/2021_2022_Oracle_E6_95_B0_E6_c102_166975.htm)

数据库应用软件很多时候将多重的、相关的实体信息保存在同一个表格中。例如，购买的零件和本地生产的零件都是零件，经理和工人都是员工，尽管多重的实体类型在数据存储上略有不同，但是它们有太多的相同之处，因此通常使用一个表格而不是两个。处理这样的表格通常意味着对每一行进行条件测试来检查哪个实体类型正在被处理，然后返回每种类型的不同结果集。CASE语句可以用来完成这一工作。从Oracle 9i版开始，COALESCE函数在很多情况下就成为替代CASE语句的一条捷径，COALESCE的格式如下: COALESCE (expression\_1, expression\_2, ...,expression\_n)列表中第一个非空的表达式是函数的返回值，如果所有的表达式都是空值，最终将返回一个空值。使用COALESCE的秘密在于大部分包含空值的表达式最终将返回空值(连接操作符“||”是一个值得注意的例外)。例如，空值加任何值都是空值，空值乘任何值也都是空值，依此类推。这样您就可以构建一系列表达式，每个表达式的结果是空值或非空，这就像一个CASE语句，表达式按照顺序进行测试，第一个满足条件的值就确定了结果。列表A展示了名为PARTS的表格的一部分，该表格存储了购买的零件数据和生产的零件数据，如果是购买的零件，那么part\_type列的值为‘P’，如果是本地生产或组装的则是‘B’。此外，对于购买的零件，purchase\_cost列会显示购买成本，而本地生产的零件则是空值。而且，本地生产的零件还有material\_qty

和material\_cost两列的信息，对于购买的零件则是空值。

```
SQL> desc partsName Null? TypePART_ID NOT NULL  
NUMBER(6)PART_TYPE NOT NULL CHAR(1)(other  
columns)PURCHASE_COST NUMBER(8,2)MATERIAL_COST  
NUMBER(8,2)MATERIAL_QTY NUMBER(6)LAST_UPDATED  
TIMESTAMP(6)UPDATED_BY VARCHAR2(30)
```

列表A：您可以使用一个CASE语句来测试part\_type列的值并返回either purchase\_cost和material\_qty列与material\_cost列的乘积.不过COALESCE可以用一行语句解决这个问题:

COALESCE(purchase\_cost, material\_qty \* material\_cost)如果数据行中存储的是一个购买的零件，那么purchase\_cost就不是空值，将返回purchase\_cost的值.然而，对于本地生产的零件，purchase\_cost是空值，COALESCE会忽略它，然后将material\_qty和material\_cost相乘，并将乘积作为结果返回。

```
SELECT part_id "Part", part_type  
"Type",COALESCE(purchase_cost, material_qty * material_cost)  
"Cost"FROM parts.
```

您可以对任何数量的表达式重复使用这个模式，COALESCE是一个非常便捷的方法对统一表格中的多重实体求值。最后，还要说一点CASE语句的优点，就是CASE是自动进行文档记录的，这便于理解和解读正在发生的事情。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](http://www.100test.com)