如何巧妙利用SQLServer的EXISTS结构 PDF转换可能丢失图片或格式，建议阅读原文 https://www.100test.com/kao_ti2020/251/2021_2022__E5_A6_82_E4_BD_95_E5_B7_A7_E5_c97_251718.htm 有件东西我观察了很多年，那就是很少有开发者会去使用SQL Server中的一个非常有用的东西EXISTS结构。你可以在各种情况中用到EXISTS结构，在这篇文章中我会向大家介绍其中的一些。 EXISTS结构 EXISTS结构是一个TSQL函数，这个函数会就圆括号中的子查询返回一个布尔变量。如果子查询返回了任意行的话，这个结构就返回真；否则，它就返回假。这个结构特别适用于查询表中是否存在某些记录。它也非常适合来决定是否在表中运行更新或者插入语句（在这篇文章中，随后我将向大家介绍）。 Listing A中创建了一个表并插入了一些数据，这样你就可以利用这张表来看看EXISTS结构是如何工作的了。Listing B中的脚本使用EXISTS结构在SalesHistory表中检查每一行，并根据已存在的行得到一个变量。这个脚本将返回1（真），这意味着SalesHistory表包含你所要查询的行。 在这种情况下，一旦在SalesHistory表中找到了一条记录，EXISTS结构就不会继续查找，并马上返回1。使用EXISTS结构额外的好处就 是，一旦它找到满足条件的记录，就会立即返回。 Listing C与上面的脚本类似，但是它的返回值是0，因为在这个条件下，EXISTS条件不满足。 由于EXISTS结构的工作原理，所以下面的语句所实现的功能是一样的。 SELECT * FROM SalesHistory WHERE EXISTS(SELECT NULL) SELECT * FROM SalesHistory WHERE 1 = 1 SELECT * FROM SalesHistory WHERE EXISTS(SELECT 1) 在刚才的脚本中，你可能会对下

面的语句感到迷惑： SELECT * FROM SalesHistory WHERE EXISTS(SELECT NULL) 因为NULL在数据库中是一个值（它是一个未知的值），在出现NULL值的地方EXISTS结构都将返回真。 开发者很少使用EXISTS对子查询进行操作。下面的查询将返回同样的结果集，虽然它们的结构不同。第一个查询使用了EXISTS的相关子查询，第二查询在SalesHistory表中，将SalePrice字段与该字段的总平均值进行比较。 SELECT * FROM SalesHistory sh WHERE EXISTS ( SELECT AVG(SalePrice) FROM SalesHistory shi HAVING AVG(SalePrice) > sh.SalePrice ) SELECT * FROM SalesHistory sh WHERE SalePrice ( SELECT AVG(SalePrice) FROM SalesHistory shi ) 你可以使用EXISTS操作来决定，是否对某一个单独的记录使用UPDATE或者INSERT语句，就像下面的例子一样： IF EXISTS ( SELECT * FROM SalesHistory WHERE SaleDate = ’1/1/2001’ AND Product = ’Computer’ AND SalePrice = 1000 ) UPDATE TOP(1) SalesHistory SET SalePrice = 1100 WHERE SaleDate = ’1/1/2001’ AND Product = ’Computer’ AND SalePrice = 1000 ELSE INSERT INTO SalesHistory (Product, SaleDate, SalePrice) VALUES ( ’Computer’, ’1/1/2001’, 1100) 虽然在实际的环境中，你可能不会用到这里所介绍的例子，但是它们给你演示了如何利用EXISTS结构来决定使用哪种DML类型。 给它一个机会 下次，当你写查询语句的时候，请记住要使用EXISTS结构。它很容易掌握，使用也很方便，并且能在程序中给你提供很多便利。 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