

基础：用于数据的XML:对XQuery的前瞻 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022__E5_9F_BA_E7_A1_80_EF_BC_9A_E7_c104_145470.htm 本专栏文章介绍了XML Query（也称为XQuery）工作草案的当前状态。Kevin Williams演示了如何使用FLWR（“flower”）子句，介绍了distinct-values函数（该函数被使用于透视数据关系）并就XQuery将如何影响数据文档操作给出了他的评价。XQuery和XML的样本代码演示了for、let、where和return（FLWR）子句的用法。如果您曾经尝试过使用XSL进行任何形式的复杂数据操作（例如连接两个元素集、透视关系或者甚至相对简单的数学计算），那么您就知道它在数据操作的特性方面稍微欠缺了一点。要解决这个问题，样式表作者们不得不使用书上的每种诀窍 将多个样式表链接在一起、对xsl:for-each运算符进行很深的嵌套以及编写令Perl专家都头痛的语法烦琐的代码。不用担心，马上就会得到帮助。即将于2002年夏季发布的XQuery规范将解决所有这些问题以及更多其它问题。那么什么是XML Query？XML Query，通常缩写为XQuery，是一种已经以这样或那样的方式存在几年的规范。XML Query工作组在1999年9月正式成立，任务是创建一种灵活的查询语言来从XML文档中抽取数据。最新的工作草案（请参阅参考资料）对实现这个目标大有帮助。XQuery构建在XPath规范之上。事实上，XQuery的一些特性已公认为是非常基本的，以致于它们已被合并入XPath 2.0规范中，而且这个规范目前为W3C的XML Query和XSL工作组共同拥有。这是个好消息，因为它意味着样式表作者们将很快就能利用

象序列、量化和更强有力的类型控制这样的特性。同样，已将条件表达式和迭代器添加到了 XPath 语言中，而在以前它们是 XSL 语言的一部分。这样就可以在样式表中编写更清晰的代码，并且为样式表创建者带来较少的麻烦。FLWR 表达式 XQuery 中最强大的新特性是 FLWR 表达式。FLWR（发音为 flower）是 For-Let-Where-Return 的首字母缩略词，这些子句都允许在这些表达式的任何一个中。FLWR 表达式可以完成很多您在 XSL 样式表中做梦都别想完成的任务。每个 FLWR 表达式都有一个或多个 for 子句、一个或多个 let 子句、一个可选的 where 子句以及一个 return 子句。for 子句如清单 1 所示，使用 for 子句来指定一组笛卡尔元组，表达式的其余部分将对该元组求值。通过为这些笛卡尔组选定次序来控制求值的次序。清单 1. 单个 for 子句 for \$exp1 in (,) 运行的程序将对清单 1 中的表达式进行二次求值，其中 \$exp 变量分别被设置为和。如果引入了另一个 for 表达式，那么这个程序将对笛卡尔积求值。看一下清单 2 中的示例，其中使用了多个 for 子句。清单 2. 多个 for 子句 for \$exp1 in (,) for \$exp2 in (,) 在清单 2 中，程序将对表达式进行四次求值，每次对一个元组求值：(,) (,) (,) (,) let 子句 let 子句为一个变量赋一个值或一个序列。在 where 或 return 子句中，这可能是非常有用的简写。where 和 return 子句 如果特定的元组不能满足特殊条件，那么 where 子句命令程序废弃这些元组。return 子句定义每个元组要返回的内容。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