

Access: using operators and functions in queries

Reference document

Aims and Learning Objectives

This document aims to cover all the query language elements (expressions, functions etc) that are available for you to use in your queries, and on your forms and reports.

Document information

This document is available on the web. To find this, go to www.bristol.ac.uk/is/learning/resources and in the **Keyword** box, type the document code given in brackets at the top of this page.

Related documentation

Other related documents are available from the web at:

<http://www.bristol.ac.uk/is/learning/resources>

Contents

Document information

Null values	1
Operators	1
Arithmetic operators	1
Character operator	1
Logical operators	1
Relational operators	2
Character matching	3
Functions	4
Aggregate functions	4
Domain aggregate functions used for subset of records ...	5
Number functions	6
Character functions	7
Date functions	8
Conversion functions	9
Other functions	10
Special values	10
Handling dates and times	11
Other format elements	13
Financial functions	14

Introduction

Unless specified, the following query elements are available in all versions of Access. Where a version is mentioned, then that element is only available from that version.

Null values

Null values are excluded from calculations. For example, if an average is requested, the value returned is the average of the **non-null** values only.

The reserved word **null** matches null values and is used with the 'is' operator, not '='

Note that null values come first in ascending order, last in descending order.

Operators

Arithmetic operators

* , + , - , /	standard operators For example, 5/2 returns 2.5
 	round to integer For example, 5\2 returns 2
^	power of For example, 2^3 returns 8
mod	remainder of result For example, 5 mod 2 returns 1

Character operator

&	concatenation For example, [initials]&" "&[surname]
--------------	--

Logical operators

and, or, not	(join two criterion together) For example town = "Bristol" and sex = "m"
eqv	if both expressions true or both expressions false then return true else return false
imp	if the first expressions implies the second expression then return true else return false eg a>b imp b>c
xor	if both expressions true or both false then return true else return false

[Click here to download full PDF material](#)