DStDev, DStDevP

You can use the DStDev and DStDevP functions to determine the standard deviation of a set of values in a specified set of records (a domain).

tags:
Database Functions

Applies to ...

Object	Description
Application	Root class. When present, its name must be "`Application`" but the object name is optional.

or to ...

Object	Description
Database	A database object opened with the **OpenDatabase** or returned by the **CurrentDb** methods.

Syntax

```
[Application.]DStDev(expression, domain[, criteria])
[Application.]DStDevP(expression, domain[, criteria])
database.DStDev(expression, domain[, criteria])
database.DStDevP(expression, domain[, criteria])
```

Arguments

Argument	Type	Optional	Description
database	Object	Y	A database object opened with the *OpenDatabase* or returned by the *CurrentDb* methods.
expression	String		An expression that identifies the field whose value you want to return. It can be a string expression identifying a field in a table or query, or it can be a SQL expression that performs a calculation on data in that field. However the SQL expression must not include any SQL aggregate function.
domain	String		A string expression identifying the set of records that constitutes the domain. It can be a table name or a query name for a query that does not require a parameter.
criteria	String	Y	An optional string expression used to restrict the range of data on which the DAvg function is performed. For example, criteria is often equivalent to the WHERE clause in an SQL expression, without the word WHERE. If criteria is omitted, the DAvg function evaluates expr against the entire domain. Any field that is included in criteria must also be a field in domain.

Returned value

```
Variant
```

Remarks

- All *expression*, *domain* and *criteria* arguments may use database record- or fieldnames surrounded with square brackets [].
- If no record satisfies *criteria* or if *domain* contains no records, the DStDev (DStDevP) function returns a **Null**.
- Construct the *criteria* argument carefully to ensure that it will be evaluated correctly as a valid WHERE clause.

Error messages

DFunction execution failed SQL='...'

See also

Example

Database functions

```
Dim sLabel As String, sKey As String, sCategory As String
Dim dblAverage As Double, iCount As Integer, dblSum As Double
Dim dblMin As Double, dblMax As Double
Dim dblStdev As Double, dblStdevP As Double, dblVar As Double, dblVarP As Double
        sKey = "27165"
        sLabel = DLookup("[DESCRIPTION]", "[PRODUCTS]", "[PRODUCT CODE]='" & sKey & "'")
        sCategory = "METALLIC"
        dblAverage = DAvg("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [
        iCount = DCount("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [SA
        dblSum = DSum("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [SALE
        '        dblAverage should be = dblSum/iCount !!!
        dblMin = DMin("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [SALE
        dblMax= DMax("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [SALES
        dblStdev = DStdev("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [
        dblStdevP = DStdevP("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AN[
        dblVar = DVar("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [SALE
        dblVarP = DVarP("[SALES PRICE]", "[PRODUCTS]", "[CATEGORY]='" & sCategory & "' AND [SA
        '         Sqr(dblVar) should be = dblStdev !!!
```

Bookmark this page » » <u>DStDev, DStDevP</u>