

How to create a Scalar Function in SQL

HELPING YOU ACHIEVE A BETTER DATABASE EXPERIENCE



WHAT THE HECK IS A SCALAR FUNCTION?

Scalar Functions are created to perform a certain action, usually related to data management and operations. They can return data with a certain type of information specified after performing desired operations or calculations.

WRITE A FUNCTION

CREATE FUNCTION -> Tell SQL to create a function

RETURNS TYPE -> SQL data type to return [e.g: INT]

AS -> Keyword to perform an action after the start condition is met

GO -> Execute any other queries that may be next

```
CREATE FUNCTION { fName } (
  @var1 TYPE
) RETURNS TYPE
AS BEGIN
  { query1 }
  { query2 }
END
GO
```

USING PARAMETERS

To add more parameters, separate them with a comma. Keep in mind you always need to indicate a name and data type for each parameter.

```
CREATE FUNCTION { fName } (
  @var1 TYPE, @var2 TYPE...
) RETURNS TYPE
AS BEGIN
  { query1 }
  { query2 }
END
GO
```

TIME FOR TESTING

Test your new function by executing a query. You may use the function's return statement as the function's returning type, and treat it as such in any query you'd like.

```
SELECT
  fName(var1, var2...)
  columnName
```

DECLARING AND RETURNING FUNCTION VARIABLES

It's a common practice to declare function variables to be manipulated inside the function. A typical "result" variable can be explicitly returned by the function as well.

```
CREATE FUNCTION { fName } (
  @var1 TYPE, @var2 TYPE...
) RETURNS TYPE
AS BEGIN
  DECLARE @result TYPE
  { do stuff }
  RETURN @result
END GO
```



schlafenhase

Source:

IBM Knowledge Center.

<https://www.ibm.com/support/knowledgecenter/SS6NHC/com.ibm.swg.im.dashdb.apdv.sqlpl.doc/doc/t0053767.html>