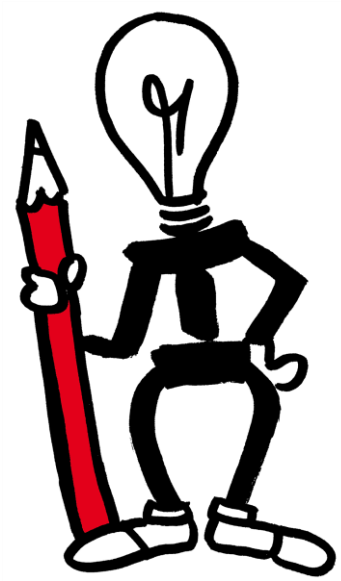
Calcscript Befehl

@CREATEBLOCK

Neu ab Version 11.1.2.3

Nutzen



Dieser Befehl legt einen oder mehrere Blöcke für ein oder mehrere Elemente einer Sparse-Dimension an und setzt die Werte in den Dense-Dimensionen in den neu angelegten Blöcken auf #Missing.

Manchmal sind neue Böcke allerdings nicht erwünscht, z.B. wenn keine Werte enthalten sind. In großen Datenbanken kann das Anlegen von nicht benötigten Blöcken zu Performance-Verlusten oder zur Erhöhung des benötigten Speicherplatzes führen.

Die Funktionen (@ALLOCATE und @MDALLOCATE) erzeugen ebenfalls neue Blöcke, sie sind jedoch speziell für die Verteilung von Werten gedacht. Im Gegensatz dazu erzeugt die Funktion @CREATEBLOCK nur Blöcke ohne Daten.

Funktionalität

Diese Fortgeschrittenen-Funktion kann zur Performance-Verbesserung durch die Verwendung von BOTTOM-UP-Kalkulationen behilflich sein. Dabei werden leere Blöcke angelegt, welche durch einen BOTTOM-UP-Prozess durchlaufen und mit Daten befüllt werden. Besonders hilfreich ist der Befehl in den Fällen, in denen Blöcke nicht automatisch erzeugt werden, z.B. beim Ausführen einer Formel in einer Dense-Dimension, wobei das Ziel Blöcke aus einer Sparse-Dimension sind.

Syntax (Auszug aus Technical Reference)

@CREATEBLOCK (mbrName | *mbrList*)

Parameter	Description
mbrList	Any single, sparse member name, a sparse member combination, or a function that returns a single member, member list or member combination. For example: • Single member name: ["200-20"] • Combination of sparse members: ["100-10"->"New York"] Member function returning *mbrName* or *mbrList*: @ANCESTORS("New York")

Info

Existiert der Block für die ausgewählte Element-Kombination hat die Funktion @CREATEBLOCK keine Auswirkung.

mbrName | mbrList können direkt eingegeben oder über eine Funktion definiert werden

Wenn mbrName eine Kombination aus mehreren Dimensionen ist, die über einen Cross-Dimension-Operator (z.B. "100-10"->"New York") definiert wurde, wird der Block auch nur für die jeweilige Kombination generiert.

Wird @CREATEBLOCK in einem Kalkulationsskript verwendet, sollte es in einem FIX-Statement verwendet werden, obgleich dies nicht unbedingt notwendig ist. Die Verwendung innerhalb eines FIX-Statements kann aber die Kalkulationszeit verbessern.

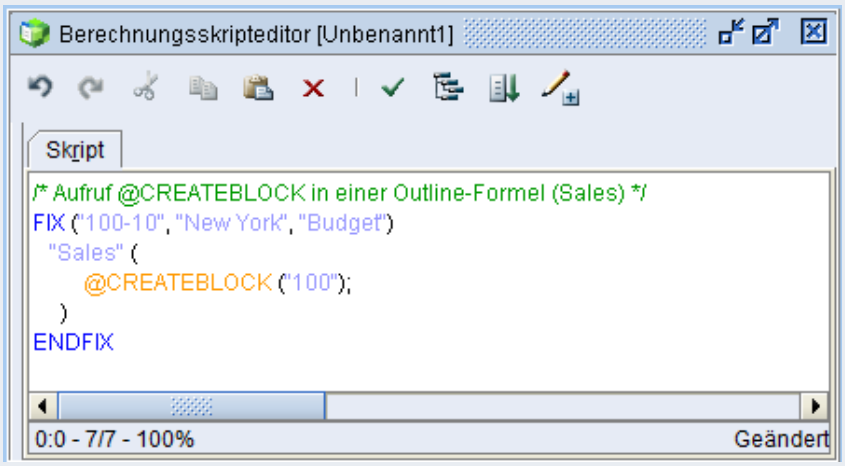
@CREATEBLOCK gibt keinen Wert zurück. Es werden nur Blöcke generiert und mit #Missing vorbelegt.

In Sparse-Dimensionen werden Formeln im TOP-DOWN-M odus ausgeführt und alle Blöcke generiert, während in Dense-Dimensionen Formeln im Bottom-up-Modus ausgeführt werden und neue Blöcke nur basierend auf bereits existierenden Blöcken angelegt werden.

@CREATEBLOCK erzeugt keine Blöcke in einer leeren Datenbank!

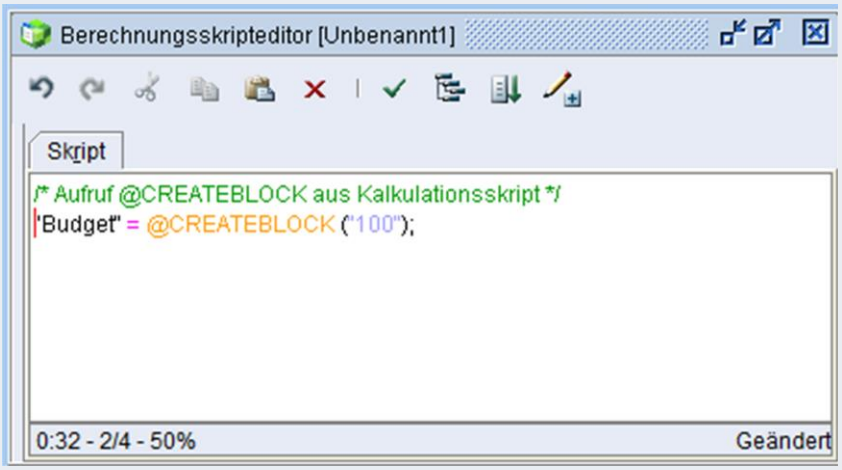
Beispiele auf der Datenbank Sample/Basic

Beispiel 1



In diesem Beispiel wird ein Block für "New York", "100" generiert.

Beispiel 2



In diesem Fall muss die Funktion direkt einem Element (“Budget“ =…) zugeordnet werden. Es werden keine neuen Blöcke generiert, da der Block bereits existiert.

cubus AG
Bahnhofstraße 29
71083 Herrenberg (Germany)
Tel +49 7032 9451-0
Fax +49 7032 9451-30

cubus Schweiz GmbH
Leutschenbachstrasse 95
CH - 8050 Zürich
Tel +41 (44) 308 3612
Fax +41 (44) 308 3500

consulting@cubus.eu
www.cubus.eu