

LibreOffice
The Document Foundation

Handleiding voor Calc

Hoofdstuk 13

Calc als een eenvoudige database

Een gids voor gebruikers en programmeurs van macro's

Auteursrechten

Dit document is onder auteursrecht © 2010 – 2013 van de bijdragers die onderstaand zijn genoemd. U mag het verspreiden en/of aanpassen met in acht neming van de condities van GNU General Public License (<http://www.gnu.org/licenses/gpl.html>), versie 3 of hoger of de Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), versie 3 of hoger.

Alle handelsmerken in deze gids zijn eigendom van de rechtmatige eigenaars

Medewerkers

Andrew Pitonyak

Reacties

De oorspronkelijke auteurs van de Engelstalige handleiding kunnen via de mailinglijst voor documentatie van LibreOffice (documentation@global.libreoffice.org) bereikt worden. Het is wel aan te raden om eerst in te schrijven op de mailinglijst van het "Documentation team" door een e-mail te sturen aan documentation@libreoffice.org. Vergeet niet om op het toegestuurd bericht te antwoorden.

Heeft u op- of aanmerkingen over de Nederlandstalige vertalingen dan kunt u terecht op de mailinglijst: documentatie@nl.libreoffice.org

Inschrijven kan via een e-mail aan documentatie+subscribe@nl.libreoffice.org

Vertalers

Dick Groskamp

Medewerkers

Aanpassingen van de lay-out voor LibreOffice

Kees Kriek

Leo Moons

Opmerking

De medewerkers hebben er voor gekozen om zo snel mogelijk een handleiding voor Calc te publiceren en gaven de voorkeur om de bestaande vertaling van Dick Groskamp te bewerken in plaats van te wachten op de "Calc Guide 4". Belangrijke wijzigingen tot versie 3.6 werden verwerkt, maar we hadden niet de ambitie om volledig te zijn.

Publicatiedatum en softwareversie

Gepubliceerd 22 januari 2011. Gebaseerd op LibreOffice 3.x.

Opmerking voor gebruikers van Mac

Sommige toetsaanslagen en menu-items zijn anders op een Mac dan die worden gebruikt in Windows en Linux. De tabel hieronder geeft enkele algemene vervangingen voor de instructies in dit hoofdstuk. Voor een meer gedetailleerde lijst, bekijk de toepassing Help.

Windows/Linux	Equivalent voor Mac	Effect
Menuselectie Extra > Opties	LibreOffice > Voorkeuren	Toegang tot instellingsopties
<i>Klik met rechts</i>	<i>Control+klik</i>	Contextmenu openen
<i>Ctrl (Control)</i>	<i>⌘ (Command)</i>	Gebruikt met andere toetsen
<i>F5</i>	<i>Shift+⌘+F5</i>	De Navigator openen
<i>F11</i>	<i>⌘+T</i>	Het venster Stijlen en opmaak openen

Inhoud

Auteursrechten.....	iii
Opmerking voor gebruikers van Mac.....	iv
Introductie.....	7
Associëren van een bereik met een naam.....	8
Benoemd bereik.....	8
Databasebereik.....	11
Sorteren.....	13
Filters.....	14
Autofilter.....	15
Standaardfilter.....	16
Speciaal filter.....	18
Manipuleren van gefilterde gegevens.....	20
Calc-functies soortgelijk aan functies voor databases.....	21
Cellen tellen en bij elkaar optellen die voldoen aan voorwaarden: AANTAL.ALS en SOM.ALS.....	22
Gefilterde cellen negeren met behulp van SUBTOTAAL.....	23
Formules gebruiken om gegevens te zoeken.....	24
In een blok gegevens zoeken met behulp van VERT.ZOEKEN.....	24
In een blok gegevens zoeken met behulp van HORIZ.ZOEKEN.....	25
In een rij of kolom zoeken met ZOEKEN.....	25
VERGELIJKEN gebruiken om de index van een waarde in een bereik te zoeken.....	26
Voorbeelden.....	26
ADRES geeft een tekenreeks terug met het adres van een cel.....	27
INDIRECT converteert een tekenreeks naar een cel of bereik.....	27
VERSCHUIVING geeft de verschuiving terug tussen een cel of bereik en een andere.....	28
INDEX geeft de cellen binnen een bepaald bereik terug.....	29
Database-specifieke functies.....	30
Conclusie.....	32

Introductie

Een Calcdocument is een zeer bruikbare database, die voldoende functionaliteit levert om te voldoen aan de behoeften van vele gebruikers. Dit hoofdstuk presenteert de capaciteiten van een Calcdocument die het geschikt maken als een programma voor een database. Waar van toepassing wordt de functionaliteit uitgelegd zowel met behulp van de GUI (Grafical User Interface) als met macro's.

Opmerking

Hoewel dit document initieel werd gemaakt voor programmeurs van macro's, zou de inhoud toegankelijk moeten zijn voor alle gebruikers. Als u geen macro's gebruikt, sla dan de gedeelten die macro's behandelen over. Aan de andere kant, als u meer over macro's wilt leren, bekijk dan zeker het boek *OpenOffice.org Macros Explained van Andrew Pitonyak en Hoofdstuk 12, Macro's in Calc.*

In een database is een record een groep gerelateerde data-items die worden behandeld als één enkele eenheid van informatie. Elk item in de record wordt een veld genoemd. Een tabel bestaat uit records. Elk record in een tabel heeft dezelfde structuur. Een tabel kan worden gevisualiseerd als een reeks rijen en kolommen. Elke rij in de tabel stemt overeen met één enkel record en elke kolom stemt overeen met de velden. Een werkblad in een Calcdocument is in structuur gelijk aan een tabel van een database. Elke cel correspondeert aan één enkel veld in een record van een database. Voor veel mensen heeft Calc voldoende functionaliteit als database zodat geen ander databaseprogramma of -functionaliteit is vereist.

Bij het onderwijzen kan een werkblad worden gebruikt als een programma met gradaties. Elke rij vertegenwoordigt één enkele student. De kolommen vertegenwoordigen de ontvangen quoteringen voor huiswerk, laboratorium en testen (zie [Tabel 1](#)). De sterke rekencapaciteit die een werkblad heeft maken dit een excellente keuze.

Tabel 1. Eenvoudig werkblad voor quoteringen


	A	B	C	D	E	F	G
1	Naam	Test 1	Test 2	Quiz 1	Quiz 2	Gemiddelde	Graad
2	André	95	93	93	92	93,25	
3	Betsy	87	92	65	73	79,25	
4	Bob	95	93	93	92	93,25	
5	Bea	45	65	92	85	71,75	
6	Frank	95	93	85	92	91,25	
7	Fred	87	92	65	73	79,25	
8	Ilse	70	85	97	79	82,75	
9	Jan	45	65	97	85	73	
10	Lisa	100	97	100	93	97,5	
11	Michelle	100	97	100	65	90,5	
12	Ravi	87	92	86	93	89,5	
13	Salvatore	45	65	100	92	75,5	
14	Ted	100	97	100	85	95,5	
15	Tom	70	85	93	65	78,25	
16	Wil	70	85	93	97	86,25	

Tip

Hoewel de keuze om een rij met een record te associëren in plaats van een kolom willekeurig is, is het bijna universeel. Met andere woorden: u zult niet snel iemand horen verwijzen naar een kolom met gegevens als één enkel record van een database.

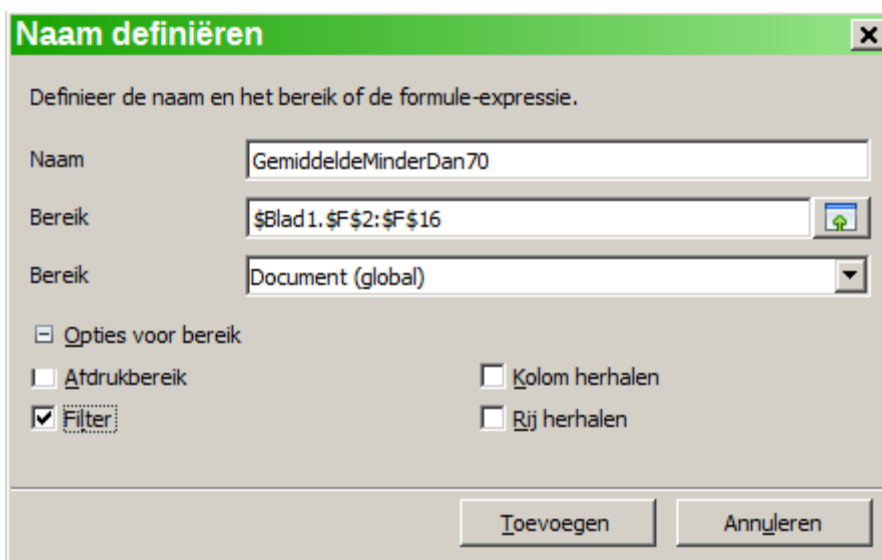
Associëren van een bereik met een naam

In een Calcdocument verwijst een bereik naar een aaneengesloten groep cellen die ten minste één cel bevat. U kunt een betekenisvolle naam associëren aan een bereik, die het u mogelijk maakt om naar het bereik te verwijzen met behulp van die betekenisvolle naam. U kunt of een *databasebereik* maken, dat een database-achtige functionaliteit heeft, of een *benoemd bereik*, wat dat niet heeft. Een naam wordt gewoonlijk om één van deze drie redenen met een bereik geassocieerd:

- 1) Associëren aan een bereik met een naam vergroot de leesbaarheid door het gebruiken van een betekenisvolle naam.
- 2) Als naar een bereik wordt verwezen met een naam op meerdere locaties, kunt u de naam aan een andere locatie toewijzen en alle verwijzingen verwijzen dan naar de nieuwe locatie.
- 3) Bereiken die geassocieerd zijn aan een naam worden weergegeven in de Navigator, die beschikbaar is door op de toets *F5* te drukken of door te klikken op het pictogram . De Navigator maakt het mogelijk snel naar de geassocieerde bereiken te navigeren.

Benoemd bereik

Het meest algemene gebruik van een benoemd bereik is, zoals de naam aangeeft, om een bereik van cellen te associëren met een betekenisvolle naam. Maak, bijvoorbeeld, een bereik genaamd *Scores*, en gebruik dan de volgende vergelijking: `=SOM(Scores)`. Selecteer het bereik dat moet worden gedefinieerd om een benoemd bereik te maken. Gebruik dan **Invoegen > Namen > Definieer** om het dialoogvenster **Namen definiëren** te openen. Gebruik het dialoogvenster **Namen definiëren** om één benoemd bereik per keer toe te voegen en aan te passen.



Afbeelding 1. Definiëren van een benoemd bereik.

In een macro kan, met behulp van de eigenschap `NamedRanges` van een `Calcdocument`, toegang tot een benoemd bereik worden gekregen, een benoemd bereik worden gemaakt en verwijderd. Gebruik de methoden `hasByName(naam)` en `getByName(naam)` om een benoemd bereik te verifiëren en op te halen. De methode `getElementNames()` geeft een reeks terug die de namen van alle benoemde bereiken bevat. Het object `NamedRanges` ondersteunt de methode `addNewByname`, die vier argumenten accepteert; de naam, inhoud, positie en type. De macro in [Lijst 1](#) maakt een benoemd bereik, als het nog niet bestaat, dat verwijst naar een celbereik.

Lijst 1. Maak een benoemd bereik dat verwijst naar \$Blad1.\$B\$3:\$D\$6.

```
Sub BenoemdBereikToevoegen()
    Dim oBereik      ' Het gemaakte bereik.
    Dim oBereiken    ' Alle benoemde bereiken.
    Dim sNaam$       ' Naam van het te maken benoemde bereik.
    Dim oCel         ' Object Cel.
    Dim s$

    sNaam$ = "MijnBereik"
    oBereiken = ThisComponent.NamedRanges
    If NOT oBereiken.hasByName(sNaam$) Then
        REM Haal het celadres op door de cel op te halen
        REM en extract dan het adres van de cel.
        Dim oCelAdres As new com.sun.star.table.CellAddress
        oCelAdres.Sheet = 0      'Het eerste blad.
        oCelAdres.Column = 1    'Kolom B.
        oCelAdres.Row = 2      'Rij 3.

        REM Het eerste argument is de naam van het bereik.
        REM Het tweede argument is de formule of uitdrukking die
        REM gebruikt moet worden.
        REM Het tweede argument is gewoonlijk een tekenreeks die
        REM het bereik definieert.
        REM Het derde argument specificeert het basisadres voor
        REM relatieve celverwijzingen.
        REM Het vierde argument is een set vlaggen die definiëren
        REM hoe het bereik wordt gebruikt, maar de meeste bereiken
        REM gebruiken 0.
        REM Het vierde argument gebruikt waarden uit de
        REM constanten NamedRangeFlag (zie Tabel 2).
        s$ = "$Blad1.$B$3:$D$6"
        oBereiken.addNewByName(sNaam$, s$, oCelAdres, 0)
    End If
    REM Haal een bereik op met behulp van het gemaakte benoemde
    REM bereik.
    oBereik = ThisComponent.NamedRanges.getByname(sNaam$)

    REM Druk de tekenreeks af die is opgenomen in cel $Blad1.$B$3
    oCel = oBereik.getReferredCells().getCellByPosition(0,0)
    Print oCel.getString()
End Sub
```

De methode `addNewByname()` accepteert vier argumenten; de naam, inhoud, positie en type. Het vierde argument voor de methode `addNewByname()` is een combinatie van vlaggen die specificeren hoe het benoemde bereik zal worden gebruikt (zie [Tabel 2](#)). De meest algemene waarde is 0, wat geen gedefinieerde constante waarde is.

Tabel 2. com.sun.star.sheet.NamedRangeFlag constanten.

Waarde	Naam	Omschrijving
1	FILTER_CRITERIA	Het bereik dat de filtercriteria bevat.
2	PRINT_AREA	Het bereik dat kan worden gebruikt als afdrukgebied.
4	COLUMN_HEADER	Het bereik dat kan worden gebruikt als kolomkoppen voor afdrukken.
8	ROW_HEADER	Het bereik dat kan worden gebruikt als rijkoppen voor afdrukken.

Het derde argument, een celadres, figureert als het basisadres voor cellen waarnaar op een relatieve manier wordt verwezen. Als het celbereik niet gespecificeerd is als een absoluut adres, zal het bereik waarnaar wordt verwezen gebaseerd zijn op waar in het werkblad het bereik wordt gebruikt. Het relatieve gedrag wordt geïllustreerd in [Lijst 2](#), dat ook een ander gebruik van een benoemd bereik illustreert – definiëren van een vergelijking. De macro in [Lijst 2](#) maakt het benoemde bereik **LinksOptellen**, dat verwijst naar de vergelijking A3+B3 met C3 als de cel waarnaar wordt verwezen. De cellen A3 en B3 zijn de twee cellen direct links van C3, dus, de vergelijking =LinksOptellen() berekent de som van de twee cellen direct links van de cel die de vergelijking bevat. Wijzigen van de cel waarnaar wordt verwezen naar C4, die onder A3 en B3 staat, zorgt er voor dat de vergelijking LinksOptellen de som van de twee cellen berekent die links op de vorige rij staan.

Lijst 2. Maak het benoemd bereik LinksOptellen.

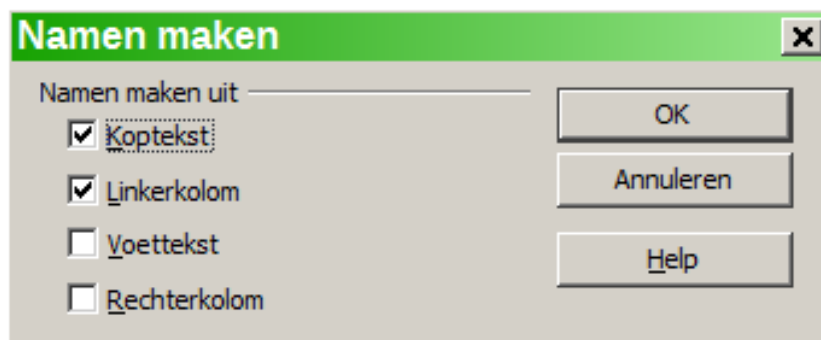
```
Sub BenoemdeFunctieToevoegen()
    Dim oBlad          'Blad dat het benoemde bereik bevat.
    Dim oCelAdres      'Adres voor relatieve verwijzingen.
    Dim oBereiken      'De eigenschap NamedRanges.
    Dim oBereik        'Bereik één enkele cel.
    Dim sNaam As String 'Naam van de te maken vergelijking.

    sNaam = "LinksOptellen"
    oBereiken = ThisComponent.NamedRanges
    If NOT oBereiken.hasByName(sNaam) Then
        oBlad = ThisComponent.getSheets().getByIndex(0)
        oBereik = oBlad.getCellRangeByName("C3")
        oCelAdres = oBereik.getCellAddress()
        oBereiken.addNewByName(sNaam, "A3+B3", oCelAdres, 0)
    End If
End Sub
```

Tip

[Lijst 2](#) illustreert twee mogelijkheden die niet breed bekend zijn. Een benoemd bereik kan een functie definiëren. Ook fungeert het derde argument als het basisadres voor cellen waarnaar op een relatieve manier wordt verwezen.

Selecteer het bereik dat de koppen en de gegevens bevat en gebruik dan **Invoegen > Namen > Maken** om het dialoogvenster **Naam maken** (zie [Afbeelding 2](#)) te openen, dat u in staat stelt om tegelijkertijd meerdere benoemde bereiken te maken gebaseerd op de bovenste rij, onderste rij, rechter kolom of linker kolom. Als u er voor kiest om bereiken te maken gebaseerd op de bovenste rij, wordt één benoemd bereik gemaakt voor elke kolomkop – de kop wordt niet opgenomen in het benoemde bereik, maar wordt gebruikt om het bereik een naam te geven.



Afbeelding 2. Definiëren van een benoemd bereik. (koptekst moet Bovenste rij en voettekst moet Onderste rij zijn – wordt aangepast in de volgende versie van LibreOffice)

De macro in [Lijst 3](#) maakt drie benoemde bereiken gebaseerd op de bovenste rij van een benoemd bereik.

Lijst 3. Maak meerdere benoemde bereiken.

```
Sub MeerdereBenoemdeBereikenToevoegen()
    Dim oBlad          'Blad dat het benoemde bereik bevat.
    Dim oAdres         'Adres van bereik.
    Dim oBereiken     'De eigenschap NamedRanges.
    Dim oBereik       'Bereik één enkele cel.

    oBereiken = ThisComponent.NamedRanges
    oBlad = ThisComponent.getSheets().getByIndex(0)
    oBereik = oBlad.getCellRangeByName("A1:C20")
    oAdres = oBereik.getRangeAddress()
    oBereiken.addNewFromTitles(oAdres, _
                               com.sun.star.sheet.Border.TOP)

End Sub
```

De constanten in [Tabel 3](#) bepalen de locatie van de koppen als meerdere bereiken worden gedefinieerd met behulp van de methode `addNewFromTitles()`.

Tabel 3. `com.sun.star.sheet.Border` constanten.

Waarde	Naam	Omschrijving
0	TOP	Selecteer de bovenste rand.
1	BOTTOM	Selecteer de onderste rand.
2	RIGHT	Selecteer de rechter rand.
3	LEFT	Selecteer de linker rand.

Waarschuwing



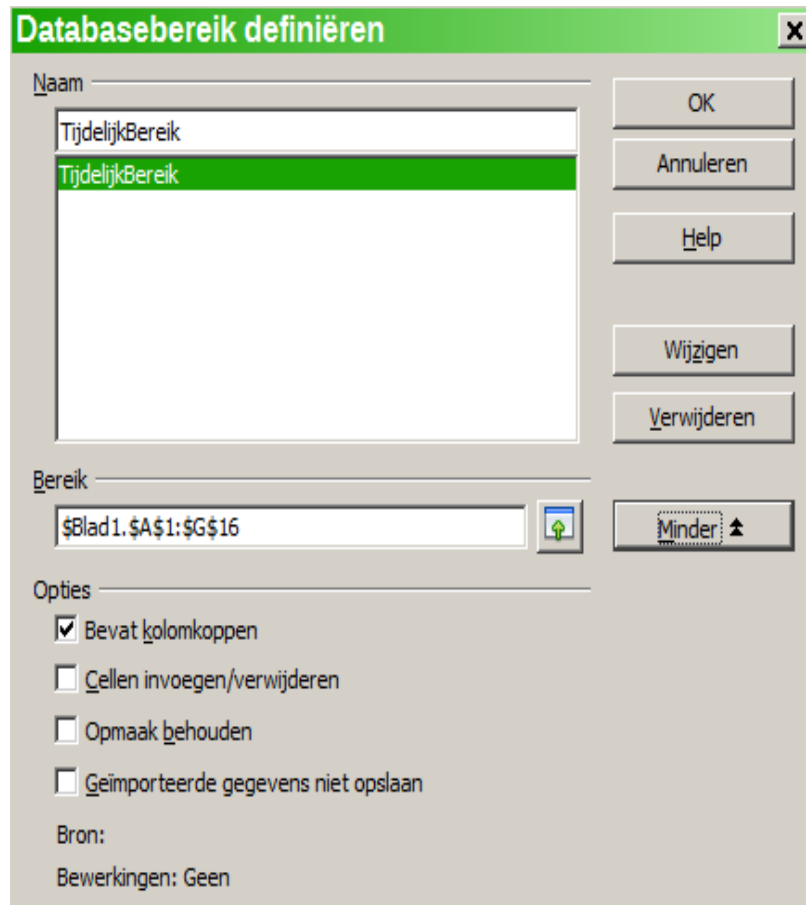
Het is mogelijk om meerdere benoemde bereiken met dezelfde naam te definiëren. Meerdere bereiken maken met één enkele opdracht vergroot de kans dat bereiken met dezelfde naam worden gemaakt – vermijd dit indien mogelijk.

Databasebereik

Hoewel een databasebereik kan worden gebruikt als een regulier benoemd bereik, kan een databasebereik ook een cellenbereik in een werkblad definiëren dat moet worden gebruikt als een

database. Elke rij in een bereik correspondeert met een record en elke cel correspondeert met een veld. U kunt sorteren, groeperen, zoeken en berekeningen uitvoeren op de bereiken alsof het een database was.

Een databasebereik creëert gedrag dat handig is voor het uitvoeren van activiteiten die gerelateerd zijn aan databases. U kunt, bijvoorbeeld, de eerste rij markeren als koppen. Gebruik **Gegevens > Bereik definiëren** om het dialoogvenster **Databasebereik definiëren** (zie [Afbeelding 3](#)) te openen, om een databasebereik te maken, aan te passen of te verwijderen. Wanneer u voor de eerste keer een bereik definieert, geeft de knop **Wijzigen**, in het voorbeeld, de tekst *Toevoegen* weer.



Afbeelding 3. Definiëren van een databasebereik.

In een macro wordt, via de eigenschap `DatabaseRanges`, toegang verkregen tot een databasebereik, er één gemaakt en/of verwijderd. De macro in [Lijst 4](#) maakt een databasebereik genaamd *MijnNaam* en stelt het bereik in dat moet worden gebruikt als een autofilter.

Lijst 4. Maak een databasebereik en een autofilter.

```
Sub NieuwDatabasebereikToevoegen()  
    Dim oBereik 'Object Databasebereik.  
    Dim oAdr    'Bereik van celadres voor het databasebereik.  
    Dim oBlad  'Eerste blad dat het bereik zal bevatten.  
    Dim oDoc   'Verwijzing naar ThisComponent met een kortere naam.  
  
    oDoc = ThisComponent  
    If NOT oDoc.DatabaseRanges.hasByName("MijnNaam") Then  
        oBlad = ThisComponent.getSheets().getByIndex(0)  
        oBereik = oBlad.getCellRangeByName("A1:F10")  
    End If  
End Sub
```

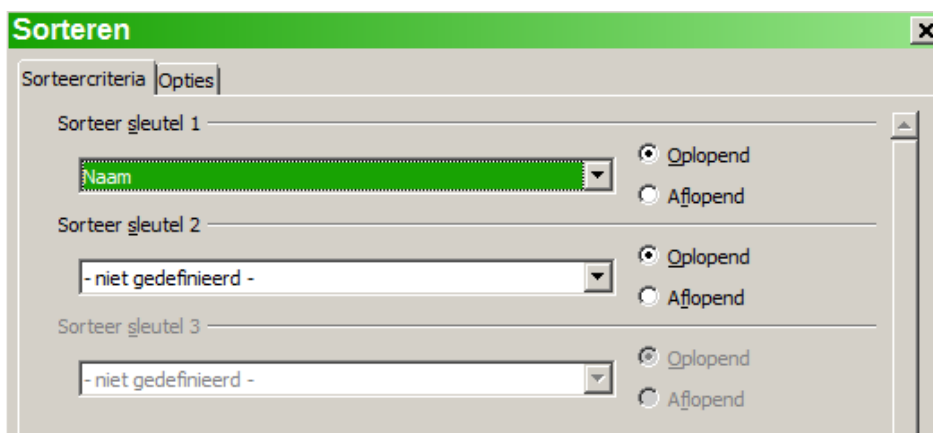
```

oAdr = oBereik.getRangeAddress()
oDoc.DatabaseRanges.addNewByName("MijnNaam", oAdr)
End If
oBereik = oDoc.DatabaseRanges.getByNamed("MijnNaam")
oBereik.AutoFilter = True
End Sub

```

Sorteren

Het mechanisme om in een Calcdocument te sorteren herschikt de gegevens in het blad. De eerste stap bij het sorteren van gegevens is het selecteren van de gegevens die u wilt sorteren. Selecteer de cellen van A1 tot en met G16 om de gegevens in [Tabel 1](#) te sorteren – indien u de kolomkoppen in de sortering wilt opnemen, geeft u dit aan in het dialoogvenster **Sorteren** (zie [Afbeelding 5](#)) Gebruik **Gegevens > Sorteren** om het dialoogvenster **Sorteren** te openen (zie [Afbeelding 4](#)). U kunt maximaal op drie kolommen of rijen tegelijkertijd sorteren.

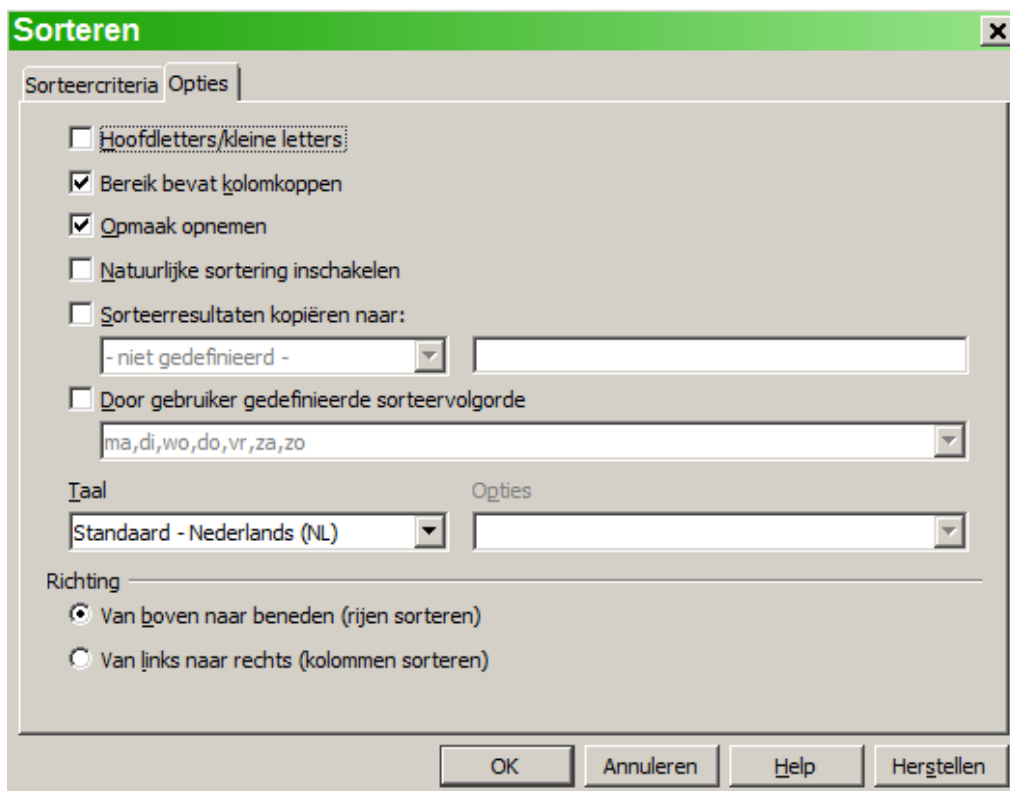


Afbeelding 4. Sorteren op de kolom Naam.

Open de tabpagina *Opties* (zie [Afbeelding 5](#)) om de opties voor het sorteren in te stellen. Vink het vak *Bereik bevat kolomkoppen* aan om te voorkomen dat de kolomkoppen tezamen met de rest van de gegevens worden gesorteerd. De keuzelijst *Sorteer sleutel* in [Afbeelding 4](#) geeft de kolommen weer die de kolomkoppen gebruiken als het vak *Bereik bevat kolomkoppen* in [Afbeelding 5](#) is aangevinkt. Als het vak *Bereik bevat kolomkoppen* niet is aangevinkt, dan worden de kolommen echter gedefinieerd door hun kolomnaam; Kolom A, bijvoorbeeld.

Normaal gesproken zorgt het sorteren er voor dat de bestaande gegevens worden vervangen door de nieuw gesorteerde gegevens. Het vak *Sorteerresultaten kopiëren naar*, zorgt er echter voor dat de geselecteerde gegevens ongewijzigd blijven en worden de gesorteerde gegevens gekopieerd naar de aangegeven locatie. U kunt direct een doeladres invoeren (bijvoorbeeld Blad3.A1) of een vooraf gedefinieerd bereik selecteren.

Vink het vak *Door gebruiker gedefinieerde sorteervolgorde aan*, om op een vooraf gedefinieerde lijst met waarden te sorteren. Gebruik **Extra > Opties > LibreOffice Calc > Sorteellijsten** om uw eigen vooraf gedefinieerde lijsten in te stellen en voer dan uw eigen sorteellijsten in. Vooraf gedefinieerde sorteellijsten zijn handig voor sorteellijsten van gegevens die niet alfabetisch of numeriek moeten worden gesorteerd. Bijvoorbeeld dagen op hun volgorde in de week sorteren.



Afbeelding 5. Opties voor sorteren instellen.

Waarschuwing



Wanneer een cel wordt verplaatst gedurende een sorteerbewerking, worden de externe verwijzingen naar die cel niet bijgewerkt. Indien een cel die een relatieve verwijzing bevat naar een andere cel wordt verplaatst, is de verwijzing relatief naar de nieuwe positie als het sorteren is voltooid. Ken het gedrag van verwijzingen bij sorteren en raak niet in paniek; dit is bijna altijd wat u wilt – omdat de verwijzing naar rechts of links is op dezelfde rij. Ook hebben we nog geen werkblad programma gevonden dat een ander gedrag vertoont voor verwijzingen bij het sorteren.

Filters

Gebruik filters om de zichtbare rijen in een werkblad te beperken. Generieke filters, algemeen voor alle soorten van bewerkingen voor gegevens, worden automatisch geleverd door de mogelijkheid Autofilter. U kunt ook uw eigen filters definiëren.

Waarschuwing



Na het toepassen van een filter, zijn sommige rijen zichtbaar en sommige rijen niet. Als u meerdere rijen in één bewerking selecteert, zult u ook de onzichtbare rijen selecteren die zijn opgenomen tussen de geselecteerde zichtbare rijen. Bewerkingen, zoals verwijderen, hebben een invloed op alle geselecteerde rijen. U moet individueel elk van de gefilterde rijen selecteren met behulp van de Ctrl-toets, om dit probleem te vermijden.

Autofilter

Gebruik autofilters om snel en eenvoudig toegankelijke filters te vinden, die algemeen gebruikt worden in verschillende typen toepassingen. Na het maken van een autofilter voor een specifieke kolom, wordt een keuzelijst toegevoegd aan de kolom. De keuzelijst verschaft snelle toegang tot de typen van het autofilter.

- Het autofilter *Alles* zorgt er voor dat alle rijen zichtbaar zijn.
- Het autofilter *Standaardfilter* opent het dialoogvenster **Standaardfilter** en is hetzelfde als Standaardfilter.
- Het autofilter *Top 10* geeft de tien rijen met de hoogste waarden weer. Als de waarde 70 in de top tien-waarden staat, dan worden alle rijen die de waarde 70 bevatten weergegeven in de gefilterde kolom. Met andere woorden: er kunnen meer dan tien rijen worden weergegeven.
- Een item voor autofilter wordt voor elk uniek item in de kolom gemaakt.

Selecteer eerst de kolommen die moeten worden gefilterd, om een autofilter te maken. Bijvoorbeeld: Met behulp van de gegevens in [Tabel 1](#): selecteer de gegevens in de kolommen B en C. Indien u de titelrijen niet selecteert, vraagt Calc of de titelrij van de huidige rij moet worden gebruikt. Hoewel u het autofilter in elke rij kunt plaatsen, worden alleen de rijen onder het autofilter gefilterd. Gebruik **Gegevens > Filter > AutoFilter** om de keuzelijst van het autofilter in de toepasselijke cel in te voegen. Gebruik tenslotte de pijl naast de keuzelijst om een toepasselijk autofilter te kiezen (zie [Afbeelding 6](#)).

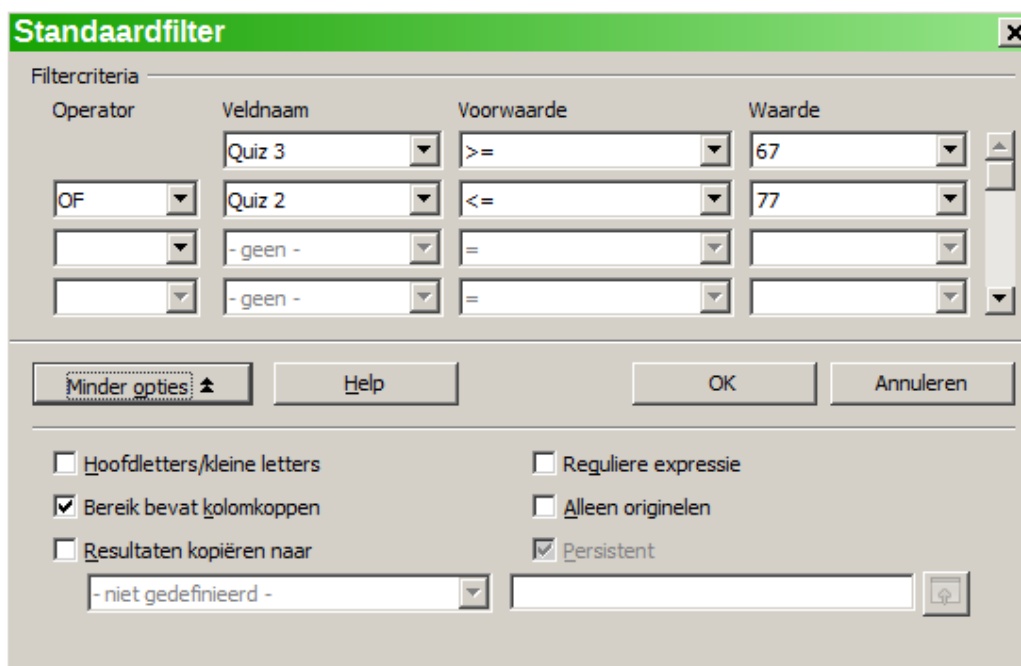
	A	B	C	D	E	F	G	H
1	Naam	Test 1	Test 2	Test 3	Quiz 1	Quiz 2	Quiz 3	Gemiddelde
2	André	95	93				55	93,25
3	Betsy	87	92				45	79,25
4	Bob	95	93				67	93,25
5	Bea	45	65				78	71,75
6	Frank	95	93				98	91,25
7	Fred	87	92				100	79,25
8	Ilse	70	85				100	82,75
9	Jan	45	65				34	73
10	Lisa	100	97				56	97,5
11	Michelle	100	97				76	90,5
12	Ravi	87	92				87	89,5
13	Salvatore	45	65				65	75,5
14	Ted	100	97				68	95,5
15	Tom	70	85				93	78,25
16	Wil	70	85				89	86,25
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								

Afbeelding 6: Een autofilter gebruiken met kolom D

Verwijder een autofilter door de stappen te herhalen voor het maken van het autofilter – met andere woorden, de menuoptie werkt als een schakelaar om het autofilter in en uit te schakelen. Wanneer een autofilter is verwijderd, wordt de keuzelijst uit de cel verwijderd. De macro in [Lijst 4](#) demonstreert het maken van een autofilter voor een bereik.

Standaardfilter

Gebruik **Gegevens > Filter > Standaardfilter** om het dialoogvenster **Standaardfilter** te openen (zie [Afbelding 7](#)) en de weergave te beperken gebaseerd op 1 tot 8 filtervoorwaarden. Gebruik **Gegevens > Filter > Filter verwijderen** om het filter uit te schakelen.



Afbelding 7: Het standaardfilter gebruiken

De macro in [Lijst 5](#) maakt een eenvoudig filter voor het eerste blad.

Lijst 5. Maak een eenvoudig filter voor een blad.

```
Sub EenvoudigFilterBlad()  
    Dim oBlad          ' Blad dat het filter zal bevatten.  
    Dim oFilterBeschr  ' Beschrijving van filter.  
    Dim oVelden(0) As New com.sun.star.sheet.TableFilterField  
  
    oBlad = ThisComponent.getSheets().getByIndex(0)  
  
    REM Als het argument True is, wordt een lege filterbeschrijving  
    REM gemaakt. Als het argument False is, maak een  
    REM beschrijving met de vorige instellingen.  
    oFilterBeschr = oBlad.createFilterDescriptor(True)  
  
    With oVelden(0)  
        REM U zou de eigenschap Connection kunnen gebruiken om aan te  
        REM geven hoe met het vorige veld moet worden verbonden. Dat is  
        REM het eerste veld, dus dit is niet vereist.  
        '.Connection = com.sun.star.sheet.FilterConnection.AND  
        '.Connection = com.sun.star.sheet.FilterConnection.OR  
  
        REM De eigenschap Field is het op nul gebaseerde kolom-  
        REM getal waarop moet worden gefilterd. Als u de cel bekend is,  
kan  
        REM .Field = oCel.CellAddress.Column gebruikt worden.  
        .Field = 5  
    End With  
End Sub
```



```

REM Vergelijken met behulp van een nummer of een tekenreeks?
.IsNumeric = True

REM De eigenschap NumericValue wordt gebruikt
REM omdat .IsNumeric = True er boven staat.
.NumericValue = 80

REM Indien IsNumeric False was, dan zou de
REM eigenschap StringValue worden gebruikt.
REM .StringValue = "wat dan ook"

REM geldige operatoren zijn onder meer EMPTY, NOT_EMPTY, EQUAL,
REM NOT_EQUAL, GREATER, GREATER_EQUAL, LESS,
REM LESS_EQUAL, TOP_VALUES, TOP_PERCENT,
REM BOTTOM_VALUES, en BOTTOM_PERCENT
.Operator = com.sun.star.sheet.FilterOperator.GREATER_EQUAL
End With

REM De beschrijving van het filter ondersteunt de volgende
REM eigenschappen: IsCaseSensitive, SkipDuplicates,
REM UseRegularExpressions,
REM SaveOutputPosition, Orientation, ContainsHeader,
REM CopyOutputData, OutputPosition en MaxFieldCount.
oFilterBeschr.setFilterFields(oVelden())
oFilterBeschr.ContainsHeader = True
oBlad.filter(oFilterBeschr)
End Sub

```

Indien een filter wordt toegepast op een blad, vervangt het elk bestaand filter voor het blad. Instellen van een leeg filter in een blad zal daarom alle filters voor dat blad verwijderen (zie [Lijst 6](#)).

Lijst 6. Verwijder het huidige filter voor het blad.

```

Sub VerwijderFilterBlad()
    Dim oBlad          ' Blad dat gefilterd moet worden.
    Dim oFilterBeschr  ' Beschrijving van filter.

    oBlad = ThisComponent.getSheets().getByIndex(0)
    oFilterBeschr = oBlad.createFilterDescriptor(True)
    oBlad.filter(oFilterBeschr)
End Sub

```

[Lijst 7](#) demonstreert een meer gevorderd filter dat filtert op twee kolommen en reguliere uitdrukkingen gebruikt. Er treedt enig onverwacht gedrag op tijdens het werken met [Lijst 7](#). Hoewel u een filterbeschrijving kunt maken met behulp van elk celbereik, is het filter van toepassing op het gehele blad.

Lijst 7. Een eenvoudig filter voor een blad met behulp van twee kolommen.

```

Sub EenvoudigFilterBlad_2()
    Dim oBlad          ' Blad dat gefilterd moet worden.
    Dim oBereik        ' Bereik dat gefilterd moet worden.
    Dim oFilterBeschr  ' Beschrijving van filter.
    Dim oVelden(1) As New com.sun.star.sheet.TableFilterField

    oBlad = ThisComponent.getSheets().getByIndex(0)
    oBereik = oBlad.getCellRangeByName("E12:G19")

    REM Als het argument True is, maakt het een

```

```

REM lege filterbeschrijving.
oFilterBeschr = oBereik.createFilterDescriptor(True)

REM Stel een veld in om cellen weer te geven die inhoud hebben
REM die begint met de letter b.
With oVelden(0)
    .Field = 0          ' Filter op kolom A.
    .IsNumeric = False ' Gebruik een tekenreeks, geen getal.
    .StringValue = "b.*" ' Alles dat begint met een b.
    .Operator = com.sun.star.sheet.FilterOperator.EQUAL
End With
REM Stel een veld in dat beide voorwaarden vereist en
REM deze nieuwe voorwaarde vereist een waarde die groter is dan
REM of gelijk is aan 70.
With oVelden(1)
    .Connection = com.sun.star.sheet.FilterConnection.AND
    .Field = 5          ' Filter op kolom F.
    .IsNumeric = True  ' Gebruik een getal
    .NumericValue = 70 ' Waarden groter dan 70
    .Operator = com.sun.star.sheet.FilterOperator.GREATER_EQUAL
End With

oFilterBeschr.setFilterFields(oVelden())
oFilterBeschr.ContainsHeader = False
oFilterBeschr.UseRegularExpressions = True
oBlad.filter(oFilterBeschr)
End Sub

```

Speciaal filter

Een speciaal filter ondersteunt maximaal acht filtervoorwaarden, in tegenstelling tot de drie die worden ondersteund door het standaard filter. De criteria voor een speciaal filter worden opgeslagen in een werkblad. De eerste stap in het maken van een speciaal filter is het invoeren van de filtercriteria in het werkblad.

- 1) Selecteer een leeg gebied in het document van Calc. Het lege gebied mag op elk werkblad op elke locatie in het document van Calc liggen.
- 2) Dupliceer de kolomkoppen vanuit het gebied dat moet worden gefilterd naar het gebied dat de filtercriteria zal bevatten.
- 3) Voer de filtercriteria in onder de kolomkoppen (zie [Tabel 4](#)). Het criterium in elke kolom van een rij is verbonden met EN. De criteria voor elke rij zijn verbonden met OF.

Tabel 4. Voorbeeld criteria voor speciaal filter

Naam	Test 1	Test 2	Quiz 1	Quiz 2	Gemiddelde	Graad
= "André"		>80				
					<80	

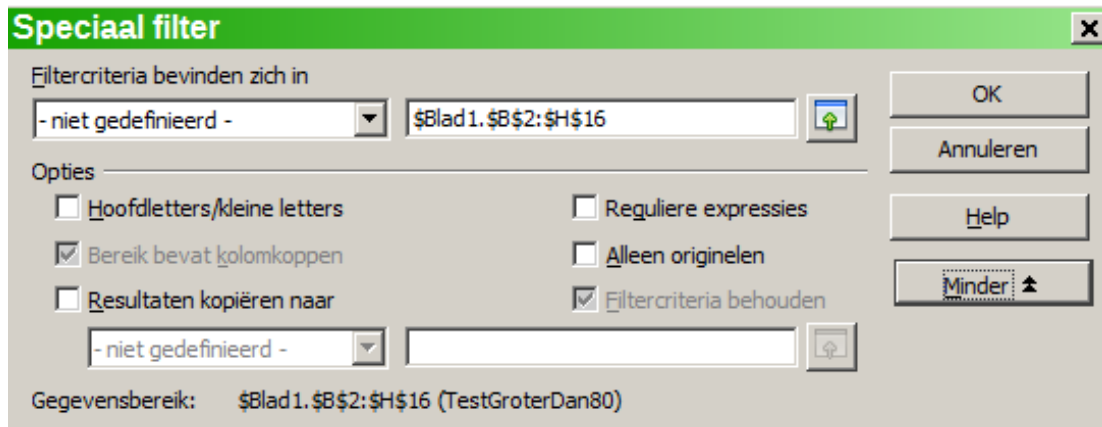
Tip

Definieer benoemde bereiken om te verwijzen naar uw speciale filtercriteria en elk bestemmingsbereik voor gefilterde gegevens (zie [Afbeelding 1](#)). Elk toepasselijk geconfigureerd benoemd bereik is beschikbaar in vakken met keuzelijsten in het dialoogvenster **Speciaal filter** (zie [Afbeelding 8](#)).

Na het maken van één of meerdere sets met filtercriteria, wordt een speciaal filter als volgt toegepast:

- 1) Selecteer de bereiken in het blad, die de gegevens bevatten die moeten worden gefilterd.
- 2) Gebruik **Gegevens > Filter > Speciaal filter** om het dialoogvenster **Speciaal filter** te openen (zie [Afbeelding 8](#)).
- 3) Selecteer het bereik dat de filtercriteria bevat en andere relevante opties.
- 4) Klik op **OK**.

Toepassen van een speciaal filter met behulp van een macro is eenvoudig (zie [Lijst 8](#)). Het celbereik dat de filtercriteria bevat, wordt gebruikt om een filterbeschrijving te maken, die dan wordt gebruikt om het bereik dat de gegevens bevat te filteren.



Afbeelding 8. Toepassen van een speciaal filter met behulp van een vooraf gedefinieerd benoemd bereik.

Lijst 8. Een speciaal filter gebruiken.

```
Sub GebruikenVanSpeciaalFilter()  
    Dim oBlad          'Een blad in het Calcdocument.  
    Dim oBereiken      'De eigenschap NamedRanges.  
    Dim oCritBereik    'Bereik dat de filtercriteria bevat.  
    Dim oGegBereik     'Bereik dat de gegevens bevat die moeten worden  
                        gefilterd.  
    Dim oFiltBeschr    'Filterbeschrijving.  
  
    REM Bereik dat de filtercriteria bevat  
    oBlad = ThisComponent.getSheets().getByIndex(1)  
    oCritBereik = oBlad.getCellRangeByName("A1:G3")  
  
    REM U kunt ook het bereik verkrijgen dat de  
    REM filtercriteria bevat vanuit een benoemd bereik.  
    REM oBereiken = ThisComponent.NamedRanges  
    REM oBereik = oRanges.getByNamed("GemiddeldeOnder80")  
    REM oCritBereik = oRange.getReferredCells()  
  
    REM De gegevens die u wilt filteren  
    oBlad = ThisComponent.getSheets().getByIndex(0)  
    oGegBereik = oBlad.getCellRangeByName("A1:G16")  
  
    oFiltBeschr =_  
                oCritBereik.createFilterDescriptorByObject(oGegBereik)  
                oGegBereik.filter(oFiltBeschr)
```

End Sub

Wijzig eigenschappen van de filterbeschrijving om het gedrag van het filter te wijzigen (zie [Tabel 5](#)).

Het filter dat is gemaakt in [Lijst 8](#) filtert de gegevens op hun plaats. Pas de eigenschap `OutputPosition` aan om een andere positie voor de uitvoer te specificeren (zie [Lijst 9](#)). De filterbeschrijving moet worden aangepast vóórdat het filter wordt toegepast.

Tabel 5. Eigenschappen voor speciaal filter.

Eigenschap	Opmerking
<code>ContainsHeader</code>	Booleaanse waarde (true of false) die specificeert als de eerste rij (of kolom) koppen bevat die niet gefilterd moeten worden.
<code>CopyOutputData</code>	Booleaanse waarde die specificeert of de gefilterde gegevens moeten worden gekopieerd naar een andere positie in het document.
<code>IsCaseSensitive</code>	Booleaanse waarde die specificeert of hoofd- of kleine letters belangrijk zijn bij het vergelijken van items.
<code>Orientation</code>	Specificeert of op kolommen (<code>com.sun.star.table.TableOrientation.COLUMNS</code>) of op rijen (<code>com.sun.star.table.TableOrientation.ROWS</code>) moet worden gefilterd.
<code>OutputPosition</code>	Specificeert, als <code>CopyOutputData True</code> is, de positie waar de gefilterde gegevens naar toe moeten worden gekopieerd.
<code>SaveOutputPosition</code>	Booleaanse waarde die specificeert of de positie <code>OutputPosition</code> wordt opgeslagen voor toekomstige aanroepen.
<code>SkipDuplicates</code>	Booleaanse waarde die specificeert of duplicaat-items buiten de resultaten moeten worden gelaten.
<code>UseRegularExpressions</code>	Booleaanse waarde die specificeert of de tekenreeksen voor het filter moeten worden geïnterpreteerd als reguliere uitdrukkingen.

Lijst 9. Kopieer gefilterde resultaten naar een andere locatie.

```
REM Kopieer de gegevens voor de uitvoer in plaats van te filteren  
REM op hun plaats.  
oFiltBeschr.CopyOutputData = True
```

```
REM Maak een CellAddress en stel het in voor Blad3,  
REM kolom B, rij 4 (onthoud: telling begint bij 0)  
Dim x As New com.sun.star.table.CellAddress  
x.Sheet = 2  
x.Column = 1  
x.Row = 3  
oFiltBeschr.OutputPosition = x
```

Manipuleren van gefilterde gegevens

Gefilterde gegevens, die zijn gekopieerd naar een nieuwe locatie, kunnen worden geselecteerd, aangepast en verwijderd naar believen. Gegevens, die echter niet zijn gekopieerd, vereisen speciale aandacht omdat rijen die niet voldoen aan de filtercriteria eenvoudigweg verborgen zijn. LibreOffice gedraagt zich anders, afhankelijk van hoe de cellen verborgen zijn en welke bewerking uitgevoerd wordt.

Cellen kunnen zijn verborgen met behulp van een overzicht, filteren van de gegevens of de opdracht Verbergen. Indien gegevens worden verplaatst door te slepen of met behulp van knippen en plakken, worden alle cellen verplaatst – inclusief de verborgen cellen. Bij het kopiëren van gegevens echter, bevatten gefilterde gegevens uitsluitend de zichtbare cellen en bij gegevens die zijn verborgen met een overzicht of de opdracht Verbergen worden alle gegevens gekopieerd.

Calc-functies soortgelijk aan functies voor databases

Hoewel elke functie van Calc kan worden gebruikt voor het manipuleren van een database, worden de functies in [Tabel 6](#) er in het algemeen meer voor gebruikt. Sommige namen van functies verschillen alleen door de letter A die aan het einde is toegevoegd; GEMIDDELDE en GEMIDDELDEA bijvoorbeeld. Functies die niet eindigen met de letter A werken alleen op numerieke waarden en cellen, die tekst bevatten of leeg zijn, worden genegeerd. De overeenkomstige functie, waarvan de naam eindigt met de letter A, behandelt tekstwaarden als een getal met waarde nul; lege cellen worden steeds genegeerd.

Tabel 6. Functies die regelmatig gebruikt worden als functies voor databases.

Functie	Omschrijving
GEMIDDELDE	Geeft het gemiddelde terug. Negeert lege cellen en cellen die tekst bevatten.
GEMIDDELDEA	Geeft het gemiddelde terug. De waarde van tekst is 0 en lege cellen worden genegeerd.
AANTAL	Telt het aantal numerieke items; tekstitems worden genegeerd.
AANTALARG	Telt het aantal niet-lege items.
AANTAL.LEGE.CELLEN	Geeft het aantal lege cellen terug.
AANTAL.ALS	Geeft het aantal cellen terug dat voldoet aan de zoekcriteria.
HORIZ.ZOEKEN	Zoekt naar een waarde en verwijzing naar de cellen in het geselecteerde gebied. Geeft de waarde in een rij van de matrix terug in dezelfde kolom.
INDEX	Geeft de inhoud van een cel terug, gespecificeerd door rij- en kolomnummer of een optionele bereiknaam.
INDIRECT	Geeft de verwijzing terug die wordt gespecificeerd door een tekenreeks van tekst.
ZOEKEN	Geeft de inhoud van een cel terug ofwel uit een bereik van één rij of één kolom óf uit een reeks.
VERGELIJKEN	Zoekt in een reeks en geeft de relatieve positie terug van het gevonden item.
MAX	Geeft de maximale numerieke waarde uit een lijst met argumenten.
MAXA	Geeft de maximale numerieke waarde uit een lijst met argumenten. De waarde van tekst is 0.
MIN	Geeft de minimale numerieke waarde uit een lijst met argumenten.
MINA	Geeft de minimale numerieke waarde uit een lijst met argumenten. De

Functie	Omschrijving
	waarde van tekst is 0.
MEDIAAN	Geeft de mediaan van een verzameling getallen.
MODUS	Geeft de meest voorkomende waarde in een verzameling gegevens. Indien er meerdere waarden zijn met dezelfde frequentie, geeft het de kleinste waarde terug. Er treedt een fout op als een waarde niet tweemaal verschijnt.
VERSCHUIVING	Geeft de waarde van een verschuiving van een cel terug met een bepaald aantal rijen en kolommen vanaf een opgegeven referentiepunt.
PRODUCT	Geeft het product van de cellen terug.
STDEV	Schat de standaard afwijking, gebaseerd op een monster.
STDEVA	Schat de standaard afwijking, gebaseerd op een monster. De waarde van tekst is 0.
STDEVP	Berekent de standaard afwijking, gebaseerd op de gehele populatie.
STDEVPA	Berekent de standaard afwijking, gebaseerd op de gehele populatie. De waarde van tekst is 0.
SUBTOTAAL	Berekent een gespecificeerde functie gebaseerd op een onderverzameling, die is gemaakt met behulp van AutoFilters.
SOM	Geeft de som van de cellen terug.
SOM.ALS	Berekent de som van de cellen, die voldoen aan de zoekcriteria.
VAR	Schat de variantie, gebaseerd op een steekproef
VARA	Schat de variantie, gebaseerd op een steekproef. De waarde van tekst is 0.
VARP	Schat de variantie, gebaseerd op de gehele populatie.
VARPA	Schat de variantie, gebaseerd op de gehele populatie. De waarde van een tekst is 0.
VERT.ZOEKEN	Zoekt naar een waarde en verwijzing naar de cellen in het geselecteerde gebied. Geeft de waarde in een kolom van de matrix terug in dezelfde rij.

De meeste functies in [Tabel 6](#) behoeven geen uitleg, ofwel omdat zij goed begrepen worden (SOM bijvoorbeeld) of omdat u weet wat zij zijn (STDEV bijvoorbeeld), als u ze moet gebruiken. Helaas worden enkele van de meest bruikbare functies niet vaak gebruikt omdat zij niet goed begrepen worden.

Cellen tellen en bij elkaar optellen die voldoen aan voorwaarden: AANTAL.ALS en SOM.ALS

De functies AANTAL.ALS en SOM.ALS berekenen hun waarden, gebaseerd op zoekcriteria. De zoekcriteria kunnen een getal, uitdrukking, tekenreeks van tekst of zelfs een reguliere uitdrukking zijn. De zoekcriteria kunnen worden opgenomen in een cel waarnaar verwezen wordt of zij kunnen direct in de aanroep voor de functie worden opgenomen.

De functie AANTAL.ALS telt het aantal cellen in een bereik dat overeenkomt met gespecificeerde criteria. Het eerste argument voor AANTAL.ALS specificeert het bereik waarin gezocht moet worden en het tweede argument zijn de zoekcriteria. [Tabel 7](#) illustreert verschillende zoekcriteria met behulp van de functie AANTAL.ALS met verwijzing naar de gegevens die worden weergegeven in [Tabel 1](#).

De eerste twee argumenten voor SOM.ALS dienen hetzelfde doel als de argumenten voor AANTAL.ALS; het bereik dat de cellen bevat die moeten worden doorzocht en de zoekcriteria. Het derde en laatste argument voor SOM.ALS specificeert het bereik dat moet worden opgeteld. Voor elke cel in het zoekbereik die overeenkomt met de zoekcriteria, wordt de overeenkomende cel in het sombereik opgeteld in de som.

Tabel 7. Voorbeelden van zoekcriteria voor de functies AANTAL.ALS en SOM.ALS.

Type criteria	Functie	Resultaat	Omschrijving
Getal	=AANTAL.ALS(B1:C16; 95)	3	Zoekt naar de numerieke waarden 95.
Tekst	=AANTAL.ALS(B1:C16; "95")	3	Zoekt naar de numerieke of tekstwaarden 95.
Uitdrukking	=AANTAL.ALS(B1:C16; ">95")	6	Zoekt numerieke waarden die groter zijn dan 95.
Uitdrukking	=AANTAL.ALS(B1:C16; 2*45+5)	3	Zoekt alleen numerieke waarden van 95.
Reguliere uitdrukking	=AANTAL.ALS(B1:C16; "9.*")	12	Zoekt getallen of tekst die beginnen met 9.
Verwijzing naar een cel	=AANTAL.ALS(B1:C16; B3)	3	Zoekt een getal of getal en tekst, afhankelijk van het gegevenstype in cel B3.
Reguliere uitdrukking	=SOM.ALS(A1:A16; "B.*"; B1:B16)	227	Telt de waarden in kolom B op voor namen in kolom A, die beginnen met de letter B.

Gefilterde cellen negeren met behulp van SUBTOTAAL

De functie SUBTOTAAL past een functie toe (zie [Tabel 8](#)) op een bereik van gegevens, maar het negeert cellen die zijn verborgen door een filter en cellen die al een SUBTOTAAL bevatten. Bijvoorbeeld: =SUBTOTAAL (2, B2 : B16) telt het aantal cellen in B2:B16, die niet verborgen zijn door een filter.

Tabel 8. Functie-index voor de functie SUBTOTAAL.

Functie-index	Functie
1	GEMIDDELDE
2	AANTAL
3	AANTALARG
4	MAX
5	MIN
6	PRODUCT
7	STDEV
8	STDEVP
9	SOM
10	VAR
11	VARP

Tip

Vergeet niet dat de functie SUBTOTAAL cellen negeert die de functie SUBTOTAAL gebruiken. Stel dat u een werkblad heeft dat investeringen bijhoudt. De investeringen voor het pensioen worden gegroepeerd met een subtotaal. Hetzelfde gebeurt voor reguliere investeringen. U kunt dus één enkel subtotaal gebruiken dat het gehele bereik bevat zonder dat u zich zorgen hoeft te maken over de cellen met de subtotalen.

Formules gebruiken om gegevens te zoeken

Calc biedt veel methoden om naar gegevens te zoeken in een blad. Bijvoorbeeld: **Bewerken > Zoeken & vervangen** biedt mogelijkheden om eenvoudig en gevorderd te zoeken. Gebruik **Gegevens > Filter** om het werkblad in omvang te beperken en alzo het zoeken te vergemakkelijken. Calc biedt ook functies voor opzoeken die kunnen worden gebruikt in formules, bijvoorbeeld in een formule om een graad van een student op te zoeken, gebaseerd op zijn test scores.

In een blok gegevens zoeken met behulp van VERT.ZOEKEN

Gebruik VERT.ZOEKEN om, in een blok met gegevens, op de eerste kolom te zoeken (kolommen zijn verticaal) en geef de waarde terug uit een andere kolom op dezelfde rij. Bijvoorbeeld: zoek in de eerste kolom op de naam 'Fred' en geef dan de waarde terug van de cel die twee plaatsen rechts daarvan staat. VERT.ZOEKEN heeft twee vormen:

```
VERT.ZOEKEN(zoek_waarde; zoek_bereik; teruggave_kolom_index)  
VERT.ZOEKEN(zoek_waarde; zoek_bereik; teruggave_kolom_index;  
sorteer_volgorde)
```

Het eerste argument, `zoek_waarde`, is de waarde die gezocht moet worden. De zoekwaarde kan tekst, een getal of een reguliere uitdrukking zijn. Bijvoorbeeld: **Fred** zoekt naar de tekst "Fred", **4** zoekt naar het getal 4, en **F.*** is de reguliere uitdrukking voor het zoeken naar iets dat begint met de letter F.

Het tweede argument, `zoek_bereik`, zijn de cellen waarin gezocht moet worden; alleen de eerste kolom wordt doorzocht. Bijvoorbeeld: **B3:G10** zoekt op hetzelfde blad dat de formule VERT.ZOEKEN bevat en **Blad2.B3:G10** zoekt in het bereik B3:G10 op het blad genaamd Blad2.

De `teruggave_kolom_index` bepaalt de kolom die moet worden teruggegeven; een waarde van 1 geeft de eerste kolom in het bereik terug. De uitdrukking `=VERT.ZOEKEN("Bob"; A1:G9; 1)` zoekt naar de eerste rij in A1:G9 die de tekst **Bob** bevat, en geeft de waarde in de eerste kolom terug. De eerste kolom is de kolom die doorzocht wordt, dus de tekst **Bob** wordt teruggegeven. Als de kolomindex 2 is dan wordt de waarde uit de cel rechts van Bob teruggegeven: kolom B.

De laatste kolom, `sorteer_volgorde`, is optioneel. De standaard waarde voor `sorteer_volgorde` is 1, wat betekent dat de eerste kolom wordt gesorteerd in oplopende volgorde; een waarde van 0 betekent dat de gegevens niet worden gesorteerd. Een niet-gesorteerde lijst wordt doorzocht door achtereenvolgens elke cel in de eerste kolom te controleren op een exacte overeenkomst. Indien een exacte overeenkomst niet wordt gevonden, wordt de tekst **#N/B** teruggegeven.

Een meer efficiëntere zoekroutine wordt gebruikt als de gegevens zijn gesorteerd in oplopende volgorde. Indien een exacte overeenkomst bestaat, is de teruggegeven waarde dezelfde als die voor een niet-gesorteerde lijst; maar het is sneller. Indien een overeenkomst niet bestaat, wordt de hoogste waarde in de kolom, die kleiner is dan of gelijk is aan de zoekwaarde, teruggegeven. Bijvoorbeeld: zoeken naar 7 in (3, 5, 10) geeft 5 terug omdat 7 tussen 5 en 10 ligt. Zoeken naar 27 geeft 10, en zoeken naar 2 geeft **#N/B** terug omdat er geen overeenkomst is en er geen waarde is die kleiner is dan 2.

Gebruik `VERT.ZOEKEN` als:

- De data in rijen zijn geschikt en u gegevens van dezelfde rij wilt weergeven. Bijvoorbeeld: namen van studenten met hun test- en quizscores, rechts van die namen van de studenten.
- Zoeken in de eerste kolom van een bereik met gegevens.

In een blok gegevens zoeken met behulp van `HORIZ.ZOEKEN`

Gebruik `HORIZ.ZOEKEN` om op de eerste rij te zoeken (rijen zijn horizontaal), in een blok met gegevens en geef de waarde terug uit een andere rij in dezelfde kolom. `HORIZ.ZOEKEN` werkt met dezelfde vormen en argumenten als `VERT.ZOEKEN`:

```
HORIZ.ZOEKEN(zoek_waarde; zoek_bereik; teruggave_kolom_index)
HORIZ.ZOEKEN(zoek_waarde; zoek_bereik; teruggave_kolom_index;
sorteer_volgorde)
```

Gebruik `HORIZ.ZOEKEN` als:

- De data in kolommen zijn geschikt en u gegevens van dezelfde kolom wilt weergeven. Bijvoorbeeld: namen van studenten met hun test- en quizscores onder die namen van de studenten.
- Zoeken in de eerste rij van een bereik met gegevens.

In een rij of kolom zoeken met `ZOEKEN`

`ZOEKEN` is soortgelijk aan `HORIZ.ZOEKEN` en `VERT.ZOEKEN`. Het zoekbereik voor de functie `ZOEKEN` is één enkele gesorteerde rij of kolom. `ZOEKEN` heeft twee vormen:

```
ZOEKEN(zoek_waarde; zoek_bereik)
ZOEKEN(zoek_waarde; zoek_bereik; teruggave_bereik)
```

De zoekwaarde is hetzelfde als voor `HORIZ.ZOEKEN` en `VERT.ZOEKEN`. Het zoekbereik moet echter één enkele rij of één enkele kolom zijn; bijvoorbeeld A7:A12 (waarden in kolom A) of C5:Q5 (waarden in rij 5). Indien `teruggave_bereik` wordt weggelaten, wordt de overeenkomende waarde teruggegeven. `ZOEKEN` gebruiken zonder een bereik voor teruggave is hetzelfde als `HORIZ.ZOEKEN` en `VERT.ZOEKEN` gebruiken met een kolomindex van 1.

Het bereik voor teruggave moet één enkele rij of kolom zijn die hetzelfde aantal elementen bevat als het zoekbereik. Indien de zoekwaarde wordt gevonden in de vierde cel in het zoekbereik, dan

wordt de waarde in de vierde cel van het bereik voor teruggave teruggegeven. Het bereik voor teruggave mag een andere oriëntatie hebben dan het zoekbereik. Met andere woorden: het zoekbereik kan een rij zijn en het bereik voor teruggave mag een kolom zijn.

Gebruik ZOEKEN als:

- De gegevens waarin gezocht moet worden oplopend zijn gesorteerd.
- De gegevens waarin gezocht moet worden niet zijn opgeslagen in dezelfde rij, kolom of oriëntatie als de terug te geven gegevens.

VERGELIJKEN gebruiken om de index van een waarde in een bereik te zoeken

Gebruik VERGELIJKEN om in één enkele rij of kolom te zoeken en de positie terug te geven die overeenkomt met de zoekwaarde. Gebruik VERGELIJKEN om de index van een waarde in een bereik te zoeken. De gebruikte vormen voor VERGELIJKEN zijn de volgende:

```
=VERGELIJKEN(zoek_waarde; zoek_bereik)  
=VERGELIJKEN(zoek_waarde; zoek_bereik; zoek_type)
```

De zoekwaarde en het zoekbereik zijn hetzelfde als voor ZOEKEN. Het laatste argument, zoektype, beslist hoe de zoekactie wordt uitgevoerd. Een zoektype van 1, gesorteerd in oplopende volgorde, is de standaard. Een zoektype van -1 geeft aan dat de lijst is gesorteerd in aflopende volgorde. Een zoektype van 0 geeft aan dat de lijst niet is gesorteerd. Reguliere uitdrukkingen kunnen alleen worden gebruikt op een lijst, die niet gesorteerd is.

Gebruik VERGELIJKEN als:

- u een index nodig hebt uit een bereik in plaats van de waarde.
- de gegevens waarin gezocht moet worden in aflopende volgorde staan en het aantal gegevens groot genoeg is dat de gegevens moeten worden doorzocht er van uitgaande dat die zijn gesorteerd; omdat het sneller gaat om een gesorteerde lijst te doorzoeken.

Voorbeelden

Bekijk de gegevens in [Tabel 1](#). Alle informatie over een student is opgeslagen in één enkele rij. Schrijf een formule om de gemiddelde graad voor 'Fred' te berekenen. Het probleem kan worden verwoord als 'Zoek in kolom A in het bereik A1:G16 naar Fred en geef de waarde in kolom F terug' (kolom F is de zesde kolom). De voor de hand liggende oplossing is
`=VERT.ZOEKEN("Fred"; A2:G16; 6)`. Net zo voor de hand liggend is
`=ZOEKEN("Fred"; A2:A16; F2:F16)`.

Over het algemeen bevat de eerste rij in een bereik, de kolomkoppen. Alle zoekfuncties controleren de eerste rij om te zien of er een overeenkomst is en negeren die dan als er geen overeenkomst is, voor het geval de eerste rij een kolomkop is.

Wat als de kolomkop **Gemiddelde** bekend is, maar niet de kolom die het gemiddelde bevat? Zoek de kolom die Gemiddelde bevat in plaats van de waarde 6 op te geven. Een kleine aanpassing met behulp van VERGELIJKEN om de kolom te zoeken levert op
`=VERT.ZOEKEN("Fred"; A2:G16; VERGELIJKEN("Gemiddelde"; A1:G1; 0))`; merk op dat de kop niet is gesorteerd. Gebruik, als een oefening, HORIZ.ZOEKEN om Gemiddelde te vinden en dan VERGELIJKEN om de rij te vinden die Fred bevat.

Als laatste voorbeeld: schrijf een formule om quoteringen toe te wijzen, gebaseerd op de gemiddelde score van een student. Laten we aannemen dat een score minder dan 51 een F is, minder dan 61 is een E, minder dan 71 is een D, minder dan 81 is een C, minder dan 91 is een B, en 91 tot en met 100 is een A. Ga er van uit dat de waarden in [Tabel 9](#) op Blad 2 staan.

Tabel 9. Scores toewijzen aan quoteringen.

	A	B
1	Score	Quotering
2	0	F
3	51	E
4	61	D
5	71	C
6	81	B
7	91	A

De formule =VERT.ZOEKEN(83; \$B1ad2.\$A\$2:\$B\$7; 2) is een voor de hand liggende oplossing. Dollartekens worden gebruikt zodat de formule kan worden gekopieerd en geplakt naar een andere locatie en nog steeds naar dezelfde zal verwijzen in [Tabel 9](#).

ADRES geeft een tekenreeks terug met het adres van een cel

Gebruik ADRES om een tekstweergave terug te geven van een celadres gebaseerd op de rij, kolom en blad; ADRES wordt regelmatig gebruikt met VERGELIJKEN. De ondersteunde vormen voor ADRES zijn de volgende:

ADRES(rij; kolom)
ADRES(rij; kolom; abs)
ADRES(rij; kolom; abs; blad)

De rij en kolom zijn waarden als gehele getallen waarbij ADRES(1; 1) **\$A\$1** teruggeeft. Het argument 'abs' specificeert welke gedeelte als absoluut wordt beschouwd en welk gedeelte als relatief wordt beschouwd (zie [Tabel 10](#)); een absoluut adres wordt gespecificeerd met behulp van het teken \$. Het blad wordt alleen opgenomen als deel van het adres als het argument 'blad' wordt gebruikt. Het argument 'blad' wordt behandeld als een tekenreeks. Door ADRES(VERGELIJKEN("Bob";A1:A5 ; 0); 2) te gebruiken met de gegevens in [Tabel 9](#) wordt **\$B\$2** teruggegeven.

Tip

Calc ondersteunt vele krachtige functies die hier niet worden besproken. Bijvoorbeeld: de uitdrukkingen RIJ, KOLOM, RIJEN en KOLOMMEN zijn niet besproken; iemand die nieuwsgierig is zou deze functies nader bekijken.

Tabel 10. Waarden ondersteunt door het argument 'abs' voor ADRES.

Waarde	Omschrijving
1	Absolute adressering gebruiken. Dit is de standaard waarde als het argument ontbreekt of als een ongeldige waarde wordt gebruikt. ADRES(2; 5; 1) geeft \$E\$2 terug.
2	Gebruik een absolute verwijzing naar de rij en een relatieve verwijzing naar de kolom. ADRES(2; 5; 2; "Blah") geeft Blah.E\$2 terug.
3	Gebruik een relatieve verwijzing naar de rij en een absolute verwijzing naar de kolom. ADRES(2; 5; 3) geeft \$E2 terug.
4	Relatieve adressering gebruiken. ADRES(2; 5; 4) geeft E2 terug.

INDIRECT converteert een tekenreeks naar een cel of bereik

Gebruik INDIRECT om een weergave als tekenreeks voor een celadres of adres voor een bereik te converteren naar een verwijzing naar de cel of het bereik. [Tabel 11](#) bevat voorbeelden voor het benaderen van gegevens zoals weergegeven in [Tabel 9](#).

Tabel 11. Voorbeelden die INDIRECT gebruiken.

Voorbeeld	Opmerking
INDIRECT("A2")	Geeft cel A2 terug, die Bob bevat.
INDIRECT(G1)	Indien cel G1 de tekst A2 bevat, dan geeft dit Bob terug.
SOM(INDIRECT("B1:B5"))	Geeft de som van het bereik B1:B5 terug, wat 194 is.
INDIRECT(ADRES(2; 1))	Geeft de inhoud terug van cel \$A\$2, wat Bob is.

VERSCHUIVING geeft de verschuiving terug tussen een cel of bereik en een andere

Gebruik VERSCHUIVING om de verschuiving tussen een cel of bereik terug te geven, gespecificeerd door een aantal rijen en kolommen vanaf een opgegeven referentiepunt. Het eerste argument bepaalt het referentiepunt. De tweede en derde argumenten specificeren het aantal rijen en kolommen om te verplaatsen vanaf het referentiepunt; met andere woorden: waar het nieuwe bereik begint. De functie VERSCHUIVING heeft de volgende syntaxis:

```
VERSCHUIVING(referentie; rijen; kolommen)
VERSCHUIVING(referentie; rijen; kolommen; hoogte)
VERSCHUIVING(referentie; rijen; kolommen; hoogte; breedte)
```

Tip

Indien de breedte of de hoogte is opgenomen geeft de functie VERSCHUIVING een bereik terug. Als zowel de breedte als de hoogte ontbreken, wordt een celverwijzing teruggegeven.

Indien de hoogte of breedte ontbreken, zijn zij standaard 1. Indien de hoogte aanwezig is, dan wordt een verwijzing naar een celbereik teruggegeven in plaats van een celverwijzing. Met behulp van de waarden uit [Tabel 1](#), gebruikt [Lijst 10](#) VERSCHUIVING om de quizscores voor de student Bob te verkrijgen.

Lijst 10. Complex voorbeeld van VERSCHUIVING.

```
=SOM(VERSCHUIVING(INDIRECT(ADRES(VERGELIJKEN("Bob";A1:A16; 0); 4)); 0; 0; 1; 2))
```

In zijn geheel is [Lijst 10](#) complex en moeilijk te begrijpen. [Tabel 12](#) isoleert elke functie in [Lijst 10](#), en geeft een eenvoudig te begrijpen uitleg over hoe het voorbeeld werkt.

Tabel 12. Opgebroken [Lijst 10](#).

Functie	Omschrijving
VERGELIJKEN("Bob";A1:A16; 0)	Geeft 4 terug omdat Bob het vierde item is in kolom A.
ADRES(4; 4)	Geeft \$D\$4 terug.
INDIRECT("\$D\$4")	Converteert \$D\$4 in een verwijzing naar de cel D4.
VERSCHUIVING(\$D\$4; 0; 0; 1; 2)	Geeft het bereik D4:E4 terug.
SOM(D4:E4)	Geeft de som terug van de quizscores van Bob.

Hoewel [Lijst 10](#) werkt zoals bedoeld, gaat het gemakkelijk en onverwacht kapot. Overweeg bijvoorbeeld wat er gebeurt als het bereik wordt gewijzigd naar A2:A16. VERGELIJKEN geeft een verschuiving naar het opgegeven bereik, dus VERGELIJKEN("Bob"; A2:A16 ; 0) geeft 3 in plaats van 4 terug. ADRES(3; 4) geeft \$D\$3 in plaats van \$D\$4 en Betsy's quizscores worden teruggegeven in plaats van die van Bob. [Lijst 11](#) gebruikt een enigszins andere methode om de quizscores van Bob te verkrijgen.

Lijst 11. Beter gebruik van VERSCHUIVING.

```
=SOM(VERSCHUIVING(A1; VERGELIJKEN("Bob"; A1:A16; 0)-1; 3; 1; 2))
```

[Tabel 13](#) bevat een beschrijving van elke functie die gebruikt is in [Lijst 11](#). Vervang A1 door A2 in zowel [Lijst 11](#) als in [Tabel 13](#) en merk op dat u nog steeds Bob's quizscores kunt verkrijgen om u te helpen uzelf te overtuigen dat [Lijst 11](#) beter is dan [Lijst 10](#).

Tabel 13. Opgebroken [Lijst 11](#).

Functie	Omschrijving
VERGELIJKEN("Bob";A1:A16; 0)-1	Geeft 3 terug omdat Bob het vierde item is in kolom A.
VERSCHUIVING(A1; 3; 3; 1; 2)	Geeft het bereik D4:E4 terug.
SOM(D4:E4)	Geeft de som terug van de quizscores van Bob.

Tip

Het eerste argument voor VERSCHUIVING mag een bereik zijn, zodat u een gedefinieerde bereiknaam kunt gebruiken.

INDEX geeft de cellen binnen een bepaald bereik terug

INDEX geeft de cellen terug die worden aangeduid door de rij- en kolomnummer(s). De rij- en kolomnummer(s) zijn relatief ten opzichte van de linker bovenhoek van het gespecificeerde verwijzingsbereik. Bijvoorbeeld: gebruik van =INDEX(B2:D3; 1; 1) geeft de cel B2 terug. [Tabel 14](#) geeft de syntaxis voor het gebruiken van de functie INDEX weer.

Tabel 14. Syntaxis voor INDEX.

Syntaxis	Omschrijving
INDEX(verwijzing)	Geeft het gehele bereik terug.
INDEX(verwijzing; rij)	Geeft de gespecificeerde rij in het bereik terug.
INDEX(verwijzing; rij; kolom)	Geeft de cel terug die wordt gespecificeerd door de rij en kolom. Een rij en kolom van 1 geeft de cel in de linker bovenhoek van het bereik weer.
INDEX(verwijzing; rij; kolom; bereik)	Een verwijzingsbereik kan meerdere bereiken bevatten. Het argument 'bereik' specificeert welk bereik moet worden gebruikt.

De functie INDEX kan een geheel bereik teruggeven, een rij of één enkele kolom (zie [Tabel 14](#)). De mogelijkheid om te indexeren, gebaseerd op het begin van het verwijzingsbereik verschaft enkele interessante gebruiksmogelijkheden. Met behulp van de waarden [Tabel 1](#), zoekt [Lijst 12](#) Bob's quizscores en geeft die terug. [Tabel 15](#) bevat een lijst van elke functie die wordt gebruikt [Lijst 12](#).

Lijst 12. Geef Bob's quizscores terug.

```
=SOM(VERSCHUIVING( INDEX(A2:G16; VERGELIJKEN("Bob"; A2:A16; 0)); 0; 3; 1; 2))
```

Tabel 15. Opgebroken [Lijst 12](#).

Functie	Omschrijving
VERGELIJKEN("Bob";A2:A16; 0)	Geeft 3 terug omdat Bob het derde item is in kolom A2:A16.
INDEX(A2:A16; 3)	Geeft A4:G4 terug – de rij die Bob's quizscores bevat.
VERSCHUIVING(A4:G4; 0; 3; 1; 2)	Geeft het bereik D4:E4 terug.
SOM(D4:E4)	Geeft de som terug van de quizscores van Bob.

Tip

Een eenvoudig bereik bevat één aaneengesloten rechthoekig gebied van cellen. Het is mogelijk om een meervoudig bereik te definiëren dat meerdere eenvoudige bereiken bevat. Als de verwijzing bestaat uit meerdere bereiken, moet u de verwijzing of bereiknaam tussen haakjes plaatsen.

Indien het argument 'verwijzing' voor de functie INDEX een meervoudig bereik is, dan specificeert het argument 'bereik' welk eenvoudig bereik moet worden gebruikt (zie [Tabel 16](#)).

Tabel 16. Gebruiken van INDEX meet een meervoudig bereik.

Functie	Geeft terug
=INDEX(B2:G2; 1; 2)	93
=INDEX(B5:G5; 1; 2)	65
=INDEX((B2:G2;B5:G5); 1; 2)	93
=INDEX((B2:G2;B5:G5); 1; 2; 1)	93
=INDEX((B2:G2;B5:G5); 1; 2; 2)	65

Database-specifieke functies

Hoewel elke functie van Calc kan worden gebruikt voor het manipuleren van een database, zijn de functies in [Tabel 17](#) speciaal ontworpen voor het gebruik als een database. De beschrijvingen in [Tabel 17](#) gebruiken de volgende uitwisselbare termen: rij en record, cel en veld, en database en alle rijen.

Tabel 17. Databasefuncties in een Calcdocument.

Functie	Omschrijving
DBGEMIDDELDE	Geeft het gemiddelde terug van alle velden die overeenkomen met de zoekcriteria.
DBAANTAL	Telt het aantal records die numerieke gegevens bevatten en die overeenkomen met de zoekcriteria.
DBAANTALC	Telt het aantal records die tekstgegevens bevatten en die overeenkomen met de zoekcriteria.
DBLEZEN	Geeft de inhoud van een veld terug dat overeenkomt met de zoekcriteria.
DBMAX	Geeft de maximale waarde terug van een veld dat overeenkomt met de zoekcriteria.
DBMIN	Geeft de minimale waarde terug van een veld dat overeenkomt met de zoekcriteria.
DBPRODUCT	Geeft het product terug van de velden die overeenkomen met de zoekcriteria.
DBSTDEV	Berekent de standaardafwijking met behulp van de velden die overeenkomen met de zoekcriteria. De velden worden behandeld als een steekproef.
DBSTDEVP	Berekent de standaardafwijking met behulp van de velden die overeenkomen met de zoekcriteria. De velden worden behandeld als de gehele populatie.
DBSOM	Geeft de som terug van alle velden die overeenkomen met de zoekcriteria.
DBVAR	Berekent de variantie met behulp van de velden die overeenkomen met de zoekcriteria. De velden worden behandeld als een steekproef.
DBVARP	Berekent de variantie met behulp van de velden die overeenkomen met de zoekcriteria. De velden worden behandeld als de gehele populatie.

De syntaxis voor de databasefuncties zijn identiek.

`DBAANTAL(database; databaseveld; zoekcriteria)`

Het argument 'database' is het celbereik dat de database definieert. Het celbereik moet de kolomlabels bevatten (zie [Lijst 13](#)). De volgende voorbeelden gaan er van uit dat de gegevens uit [Tabel 1](#) zijn geplaatst op Blad 1 en de filtercriteria in [Tabel 4](#) zijn geplaatst op Blad 2.

Lijst 13. Het argument 'database' is inclusief de koppen.

`=DBAANTAL(A1:G16; "Test 2"; Blad2.A1:G3)`

Het databaseveld specificeert de kolom waarop de functie wordt uitgevoerd nadat de zoekcriteria zijn toegepast en de rijen met gegevens zijn geselecteerd. Het databaseveld kan worden gespecificeerd met behulp van de kolomkop of als een geheel getal. Indien de kolom als een geheel getal is gespecificeerd, specificeert 0 het gehele bereik met gegevens, 1 specificeert de eerste kolom, 2 specificeert de tweede kolom, enzovoort. [Lijst 14](#) berekent de gemiddelde testscore voor de rijen die overeenkomen met de zoekcriteria.

Lijst 14. "Test 2" is kolom 3.

```
=DBGEMIDDELDE(A1:G16; "Test 2"; Blad2.A1:G3)
```

```
=DBGEMIDDELDE(A1:G16; 3; Blad2.A1:G3)
```

Zoekcriteria is het celbereik dat de zoekcriteria bevat. De zoekcriteria zijn identiek aan het speciaal filter; criteria in dezelfde rij worden verbonden door EN en criteria in verschillende rijen worden verbonden door OF.

Conclusie

Een Calcdocument verschaft voldoende functionaliteit als database om aan de behoeften van de meeste mensen te voldoen. De niet frequent gebruikte databasefuncties, zoals VERSCHUIVING en INDEX, zijn de tijd waard om ze te leren gebruiken en kunnen u op termijn tijd besparen.