

Nr. 1

Excel tidsskriftet

Juli 2022

Bo Jönsson
[Dato]

Velkommen

Velkommen til Excel tidsskriftet. Jeg bruger fritiden på at grave i de mange bøger, der står på bogreolen, gennemgår hundredvis af artikler på nettet. Alt sammen med det formål at finde interessante emner inden for Excel.

Det er planen at udgive et eksemplar hver måned. Meget af indholdet i Excel tidsskriftet kan også findes som artikler på hjemmesiden webbojo.dk. Her er de samlet, så indholdet kan printes, hvis man har det bedst med papirudgaven.

Dette nummer

En af de vigtigste nyheder i mange år er de dynamiske array formler. I dette notesæt gennemgås de med flere eksempler.

Andre nyere tiltag er funktionen LAMBDA. Funktionen LAMBDA gør det muligt at oprette brugerdefinerede funktioner. Det har du muligvis tidligere brugt VBA til.

Vi skal også se på funktionen LET. Excel funktionen LET lader dig definere navngivne variable i en formel.

Indhold

| | | | |
|--|----|--|----|
| Excel LAMBDA funktion | 3 | Medfødt adfærd | 21 |
| Syntaks..... | 3 | Alle formler | 22 |
| Argumenter | 3 | Arrays bliver mainstream..... | 23 |
| Oprettelse af en LAMBDA-funktion..... | 3 | Array-operationer bliver vigtige | 23 |
| Eksempel 1 - grundlæggende eksempel..... | 4 | Nye og gamle matrixformler..... | 24 |
| Eksempel 2 - kuglevolumen..... | 6 | Tegnet @..... | 26 |
| Eksempel 3 - tæl ord..... | 8 | Opsummering | 26 |
| Løs problemet med tomme celler | 9 | | |
| Find rækketotaler med LAMBDA..... | 11 | | |
| Generisk formel | 11 | | |
| Resumé | 11 | | |
| Forklaring..... | 11 | | |
| Med funktionen BYROW..... | 11 | | |
| Med funktionen MPRODUKT..... | 12 | | |
| Alternativt kan bruges SEKVENS..... | 12 | | |
| Excel funktionen LET..... | 14 | | |
| Formål..... | 14 | | |
| Returværdi..... | 14 | | |
| Syntaks..... | 14 | | |
| Argumenter | 14 | | |
| Noter..... | 14 | | |
| Vigtige fordele | 15 | | |
| Eksempel 1..... | 15 | | |
| Eksempel 2..... | 15 | | |
| Dynamiske arrayformler i Excel..... | 17 | | |
| Tilgængelighed | 17 | | |
| Nye funktioner..... | 17 | | |
| Eksempel..... | 17 | | |
| Overløb - én formel, mange værdier | 18 | | |
| Reference for overløb område | 19 | | |
| Massiv forenkling..... | 19 | | |
| Den enes magt..... | 20 | | |
| Kædefunktioner | 20 | | |

Excel LAMBDA funktion

Excel funktionen LAMBDA giver mulighed for at oprette brugerdefinerede funktioner, der kan genbruges i en projektmappe uden VBA eller makroer.

Syntaks

=LAMBDA (parameter; ...; beregning)

Argumenter

parameter - En inputværdi for funktionen.

Beregning - Beregningen, der skal udføres som resultatet af funktionen. Skal være sidste argument.

Noter

LAMBDA-funktionen giver mulighed for at oprette en brugerdefineret funktion i Excel. Når den først er defineret og navngivet, kan en LAMBDA-funktion bruges hvor som helst i en projektmappe. LAMBDA-funktioner kan være meget enkle eller ret komplekse, idet de samler mange Excel-funktioner i én formel. En brugerdefineret LAMBDA-funktion kræver ikke VBA eller makroer.

I computerprogrammering refererer udtrykket LAMBDA til en anonym funktion eller udtryk. En anonym funktion er en funktion defineret uden navn. I Excel bruges LAMBDA-funktionen først til at oprette en generisk (unavngiven) formel. Når en generisk version er blevet oprettet og testet, overføres den til Navnstyringen, hvor den formelt defineres og navngives.

En af de vigtigste fordele ved en brugerdefineret LAMBDA-funktion er, at logikken i formlen kun findes ét sted. Det betyder, at der kun er én kopi af koden, der skal opdateres, når der løses problemer eller opdateres funktionalitet, og ændringer vil automatisk spredes til alle forekomster af funktionen LAMBDA i en projektmappe.

funktionen LET bruges ofte sammen med funktionen LAMBDA. LET giver mulighed for at erklære variabler og tildele værdier i en formel. Dette gør mere komplicerede formler lettere at læse ved at reducere redundant kode. LET-funktionen kan også forbedre ydeevnen ved at reducere antallet af beregninger udført af en formel.

Oprettelse af en LAMBDA-funktion

LAMBDA-funktioner oprettes og fejlsøges typisk i formellinjen i et regneark og flyttes derefter ind i navnehåndteringen for at tildele et navn, der kan bruges hvor som helst i en projektmappe.

Der er fire grundlæggende trin i oprettelsen og brugen af en brugerdefineret LAMBDA-funktion:

1. Verificér den logik, du vil bruge, med en standardformel
2. Opret og test en generisk (unavngivet) LAMBDA-version af formlen
3. Navngiv og definer LAMBDA-formlen med navneadministratoren
4. Kald den nye brugerdefinerede funktion med det definerede navn

Eksemplerne nedenfor gennemgår disse trin mere detaljeret.

Excel LAMBDA funktion

Eksempel 1 - grundlæggende eksempel

For at illustrere, hvordan LAMBDA virker, lad os da begynde med en meget simpel formel:

```
=x*y // vi ganger x og y
```

I Excel vil denne formel typisk bruge cellereferencer som dette:

```
=A4*B4 // med cellereferencer
```

| | A | B | C | D | E | F |
|----|----|----|---|-------------------|-----------------|------------------|
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | x | y | | Almindelig formel | LAMBDA generisk | LAMBDA navngivet |
| 4 | 2 | 6 | | 12 | | |
| 5 | 4 | 7 | | 28 | | |
| 6 | 6 | 8 | | 48 | | |
| 7 | 8 | 9 | | 72 | | |
| 8 | 10 | 10 | | 100 | | |
| 9 | 12 | 11 | | 132 | | |
| 10 | 14 | 12 | | 168 | | |
| 11 | 16 | 13 | | 208 | | |
| 12 | 18 | 14 | | 252 | | |
| 13 | 20 | 15 | | 300 | | |
| 14 | | | | | | |
| 15 | | | | | | |

Som du kan se, fungerer formlen fint, så vi er klar til at gå videre til at lave en generisk LAMBDA-formel (unavngivet version). Den første ting at overveje er, om formlen kræver input (parametre). I dette tilfælde er svaret "ja" – formlen kræver en værdi for x og en værdi for y. Når det er fastslået, starter vi med LAMBDA-funktionen og tilføjer de nødvendige parametre til brugerinput:

```
=LAMBDA(x;y // begynder med inputparametre
```

Dernæst skal vi tilføje den faktiske beregning, $x*y$:

```
=LAMBDA(x;y;x*y) (A4;B4)
```

Hvis du indtaster formlen på dette tidspunkt, får du en #BEREGN! fejl. Dette sker, fordi formlen ikke har nogen inputværdier at arbejde med, da der ikke længere er nogen cellereferencer. For at teste formlen skal vi bruge en speciel syntaks som denne:

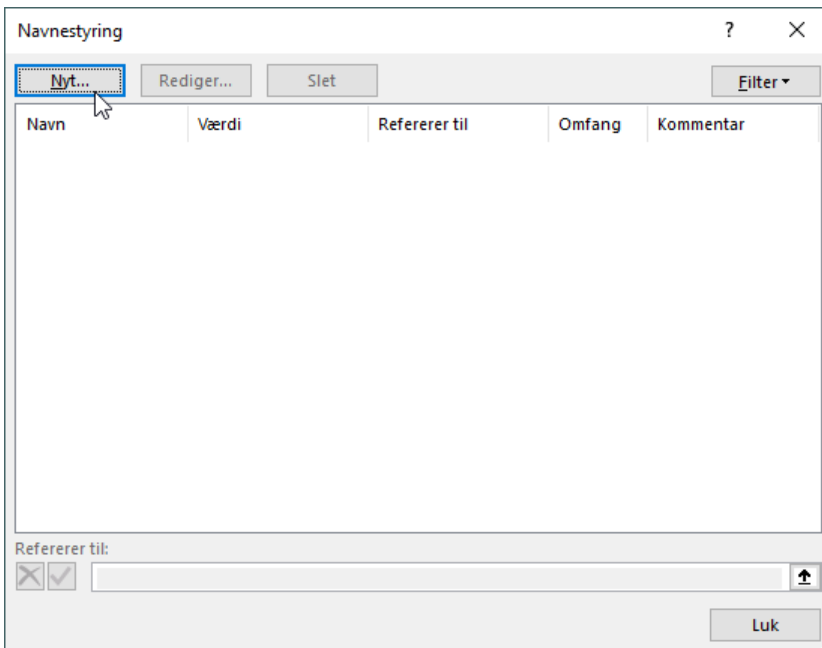
```
=LAMBDA(x;y;x*y) (A4;B4) // test syntaks
```

Denne syntaks, hvor parametre angives i slutningen af en LAMBDA-funktion i et separat sæt parenteser, er unik for LAMBDA-funktioner. Dette gør det muligt at teste formlen direkte i regnearket, før LAMBDA'en navngives. I skærmbilledet nedenfor kan du se, at den generiske LAMBDA-funktion i F5 returnerer nøjagtigt det samme resultat som den oprindelige formel i E4:

Excel LAMBDA funktion

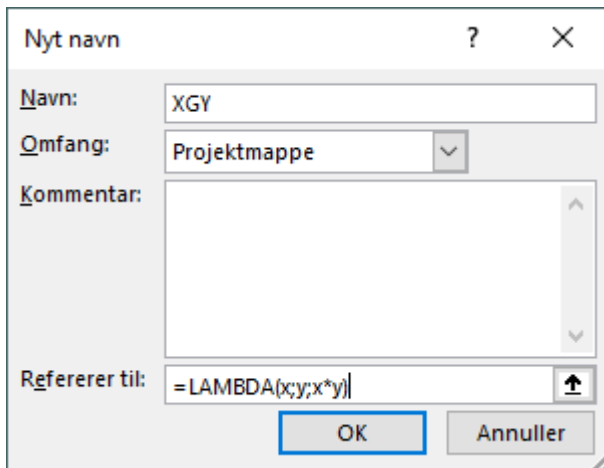
| | A | B | C | D | E | F |
|----|----|----|---|-------------------|-----------------|------------------|
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | x | y | | Almindelig formel | LAMBDA generisk | LAMBDA navngivet |
| 4 | 2 | 6 | | 12 | 12 | |
| 5 | 4 | 7 | | 28 | 28 | |
| 6 | 6 | 8 | | 48 | 48 | |
| 7 | 8 | 9 | | 72 | 72 | |
| 8 | 10 | 10 | | 100 | 100 | |
| 9 | 12 | 11 | | 132 | 132 | |
| 10 | 14 | 12 | | 168 | 168 | |
| 11 | 16 | 13 | | 208 | 208 | |
| 12 | 18 | 14 | | 252 | 252 | |
| 13 | 20 | 15 | | 300 | 300 | |
| 14 | | | | | | |
| 15 | | | | | | |

Vi er nu klar til at navngive LAMBDA-funktionen med Navnestyring. Først skal du kopiere formlen uden at inkludere testparametrene i slutningen. Åbn derefter Navnestyring med genvejen Ctrl + F3, og klik på Nyt.



Indtast navnet "XGY" i dialogboksen Nyt navn, lad Omfang være indstillet til Projektmappe, og indsæt den formel, du kopierede, i inputområdet "Refererer til". (Tip: Brug tabulatortasten til at navigere til feltet "Refererer til").

Excel LAMBDA funktion



Nyt navn

Navn: XGY

Omfang: Projektmappe

Kommentar:

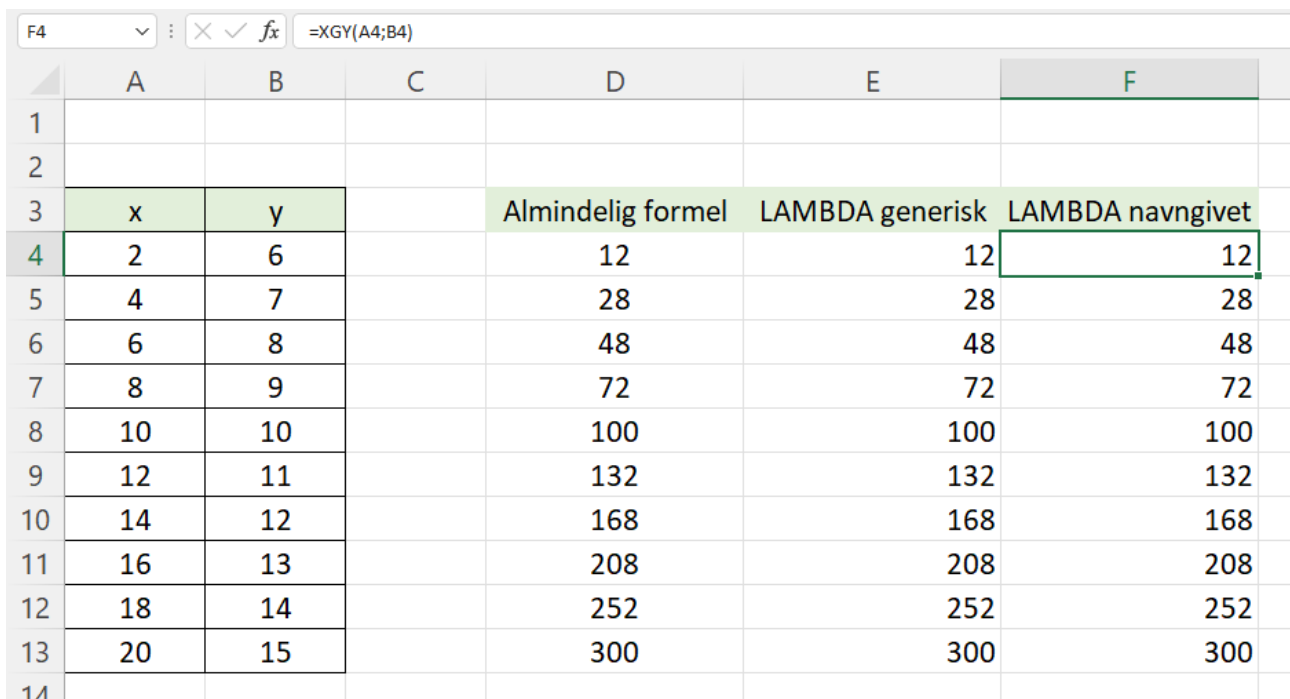
Refererer til: =LAMBDA(x;y;x*y)

OK Annuller

Sørg for, at formlen begynder med et lighedstegn (=). Nu hvor LAMBDA-formlen har et navn, kan den bruges i projektmappen som enhver anden funktion. I billedet nedenfor er formlen i F4, kopieret ned:

=XGY (A4 ; B4)

Skærbilledet nedenfor viser, hvordan det ser ud i projektmappen:



| | A | B | C | D | E | F |
|----|----|----|-------------------|-----------------|------------------|---|
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | x | y | Almindelig formel | LAMBDA generisk | LAMBDA navngivet | |
| 4 | 2 | 6 | 12 | 12 | 12 | |
| 5 | 4 | 7 | 28 | 28 | 28 | |
| 6 | 6 | 8 | 48 | 48 | 48 | |
| 7 | 8 | 9 | 72 | 72 | 72 | |
| 8 | 10 | 10 | 100 | 100 | 100 | |
| 9 | 12 | 11 | 132 | 132 | 132 | |
| 10 | 14 | 12 | 168 | 168 | 168 | |
| 11 | 16 | 13 | 208 | 208 | 208 | |
| 12 | 18 | 14 | 252 | 252 | 252 | |
| 13 | 20 | 15 | 300 | 300 | 300 | |
| 14 | | | | | | |

Den nye brugerdefinerede funktion returnerer det samme resultat som de to andre formler.

Eksempel 2 - kuglevolumen

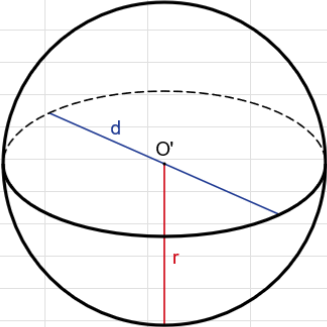
I dette eksempel konverterer vi en formel til at beregne rumfanget af en kugle til en brugerdefineret LAMBDA-funktion. Den generelle Excel-formel til beregning af rumfanget af en kugle er:

=4/3*PI()*A1^3 // rumfanget af en kugle

hvor A1 repræsenterer radius. Skærbilledet nedenfor viser denne formel i aktion:

Excel LAMBDA funktion

| Radius | Rumfang |
|--------|-----------|
| 10 | 4.189 |
| 20 | 33.510 |
| 30 | 113.097 |
| 40 | 268.083 |
| 50 | 523.599 |
| 60 | 904.779 |
| 70 | 1.436.755 |
| 80 | 2.144.661 |
| 90 | 3.053.628 |
| 100 | 4.188.790 |



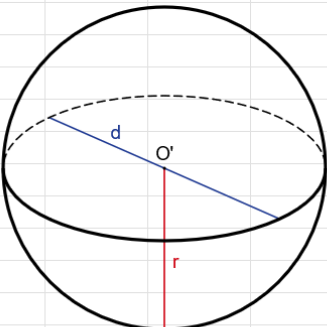
$$V = \frac{4}{3} \pi r^3$$

Bemærk, at denne formel kun kræver et input (radius) for at beregne rumfanget, så vores LAMBDA-funktion behøver kun én parameter (r), som vises som det første argument. Her er formelen konverteret til LAMBDA:

```
=LAMBDA(r;4/3*PI()*r^3) // generisk lambda
```

Tilbage i regnearket har vi erstattet den originale formel med den generiske LAMBDA-version. Bemærk, at vi bruger testsyntaksen, som giver os mulighed for at benytte A4 til radius:

| Radius | Rumfang |
|--------|-----------|
| 10 | 4.189 |
| 20 | 33.510 |
| 30 | 113.097 |
| 40 | 268.083 |
| 50 | 523.599 |
| 60 | 904.779 |
| 70 | 1.436.755 |
| 80 | 2.144.661 |
| 90 | 3.053.628 |
| 100 | 4.188.790 |



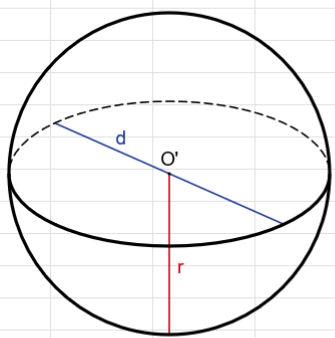
$$V = \frac{4}{3} \pi r^3$$

Resultaterne fra den generiske LAMBDA-formel er nøjagtig de samme som den originale formel, så det næste trin er at definere og navngive denne LAMBDA-formel med Navnestyringen, som forklaret ovenfor. Navnet, der bruges til en LAMBDA-funktion, kan være et hvilket som helst gyldigt Excel-navn. I dette tilfælde vil vi navngive formelen "SfæreRumfang".

Tilbage i regnearket har vi erstattet den generiske (unavngivne) LAMBDA-formel med den navngivne LAMBDA-version og indtastet A4 for r. Bemærk, at resultaterne returneret af den tilpassede SfæreRumfang-funktion er nøjagtig de samme som tidligere resultater.

Excel LAMBDA funktion

| Radius | Rumfang |
|--------|-----------|
| 10 | 4.189 |
| 20 | 33.510 |
| 30 | 113.097 |
| 40 | 268.083 |
| 50 | 523.599 |
| 60 | 904.779 |
| 70 | 1.436.755 |
| 80 | 2.144.661 |
| 90 | 3.053.628 |
| 100 | 4.188.790 |



$$V = \frac{4}{3}\pi r^3$$

Eksempel 3 - tæl ord

I dette eksempel vil vi oprette en LAMBDA-funktion til at tælle ord. Excel har ikke en funktion til dette formål, men du kan tælle ord i en celle med en brugerdefineret formel baseret på funktionerne LÆNGDE og ERSTAT som dette:

```
=LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) - LÆNGDE ( UDSKIFT ( A2 ; " " ; "" ) ) + 1
```

Her er formelen i aktion i et regneark:

| A | B |
|--|-----------|
| Tekst | Antal ord |
| Video er en kraftfuld metode til at understrege dine ideer. | 10 |
| Når du klikker på Onlinevideo, kan du indsætte integreringskoden for den video, du vil tilføje. | 15 |
| Du kan også skrive et nøgleord, hvis du vil søge online efter en video, der passer bedst til dit dokument | 20 |
| I Word finder du design med sidehoveder, sidefodder, følgebrev og tekstfelter, der passer til hinanden, så dine dokumenter kan se professionelle ud. | 22 |
| Du kan f.eks. tilføje matchende forside, overskrift eller margentekst. | 9 |
| Klik på Indsæt for at vælge mellem elementerne i de forskellige gallerier. | 12 |
| | 1 |
| Når du klikker på Design og vælger et nyt tema, ændres billeder, diagrammer og SmartArt-grafik, så de passer til dit nye tema. | 22 |
| Når du anvender typografier, ændres dine overskrifter til at matche det nye tema. | 13 |
| Spar tid i Word med nye knapper, der vises, når du har brug for dem. | 15 |

Bemærk, at vi får et forkert resultat på 1, når formelen møder en tom celle (A8). Vi behandler dette problem nedenfor.

Denne formel kræver kun ét input, som er den tekst, der indeholder ord. I vores LAMBDA-funktion vil vi navngive dette argument "tekst". Her er formelen konverteret til LAMBDA:

```
=LAMBDA ( tekst ; LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) - LÆNGDE ( UDSKIFT ( A2 ; " " ; "" ) ) + 1 )
```

Bemærk "tekst" vises som det første argument, og beregningen er det andet og sidste argument. I billedet nedenfor har vi erstattet den originale formel med den generiske LAMBDA-version. Bemærk, at vi bruger testsyntaksen, som giver os mulighed for at benytte B5 til tekst:

```
=LAMBDA ( tekst ; LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) - LÆNGDE ( UDSKIFT ( A2 ; " " ; "" ) ) + 1 ) ( A2 )
```

Excel LAMBDA funktion

| A | | B |
|----|--|-----------|
| 1 | Tekst | Antal ord |
| 2 | Video er en kraftfuld metode til at understrege dine ideer. | 10 |
| 3 | Når du klikker på Onlinevideo, kan du indsætte integreringskoden for den video, du vil tilføje. | 15 |
| 4 | Du kan også skrive et nøgleord, hvis du vil søge online efter en video, der passer bedst til dit dokument | 20 |
| 5 | I Word finder du design med sidehoveder, sidefodder, følgebrev og tekstfelter, der passer til hinanden, så dine dokumenter kan se professionelle ud. | 22 |
| 6 | Du kan f.eks. tilføje matchende forside, overskrift eller margentekst. | 9 |
| 7 | Klik på Indsæt for at vælge mellem elementerne i de forskellige gallerier. | 12 |
| 8 | | 1 |
| 9 | Når du klikker på Design og vælger et nyt tema, ændres billeder, diagrammer og SmartArt-grafik, så de passer til dit nye tema. | 22 |
| 10 | Når du anvender typografier, ændres dine overskrifter til at matche det nye tema. | 13 |
| 11 | Spar tid i Word med nye knapper, der vises, når du har brug for dem. | 15 |

Resultaterne fra den generiske LAMBDA-formel er de samme som den originale formel, så næste trin er at definere og navngive denne LAMBDA-formel med Navnstyringen, som forklaret tidligere. Vi vil navngive denne formel "OrdOptælling".

Nedenfor har vi erstattet den generiske (unavngivne) LAMBDA-formel med den navngivne LAMBDA-version og indtastet A2 for tekst. Bemærk, at vi får nøjagtig de samme resultater.

| A | | B |
|----|--|-----------|
| 1 | Tekst | Antal ord |
| 2 | Video er en kraftfuld metode til at understrege dine ideer. | 10 |
| 3 | Når du klikker på Onlinevideo, kan du indsætte integreringskoden for den video, du vil tilføje. | 15 |
| 4 | Du kan også skrive et nøgleord, hvis du vil søge online efter en video, der passer bedst til dit dokument | 20 |
| 5 | I Word finder du design med sidehoveder, sidefodder, følgebrev og tekstfelter, der passer til hinanden, så dine dokumenter kan se professionelle ud. | 22 |
| 6 | Du kan f.eks. tilføje matchende forside, overskrift eller margentekst. | 9 |
| 7 | Klik på Indsæt for at vælge mellem elementerne i de forskellige gallerier. | 12 |
| 8 | | 1 |
| 9 | Når du klikker på Design og vælger et nyt tema, ændres billeder, diagrammer og SmartArt-grafik, så de passer til dit nye tema. | 22 |
| 10 | Når du anvender typografier, ændres dine overskrifter til at matche det nye tema. | 13 |
| 11 | Spar tid i Word med nye knapper, der vises, når du har brug for dem. | 15 |

Formlen, der bruges i Navnstyringen til at definere OrdOptælling, er den samme som ovenfor uden testsyntaksen:

```
=LAMBDA (tekst; LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) - LÆNGDE ( UDSKIFT ( A2 ; " " ; "" ) ) + 1 )
```

Løs problemet med tomme celler

Som nævnt ovenfor returnerer formelen ovenfor et forkert resultat på 1, når en celle er tom. Dette problem kan løses ved at erstatte +1 med koden nedenfor:

```
=LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) - LÆNGDE ( UDSKIFT ( A2 ; " " ; "" ) ) + ( LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) > 0 )
```

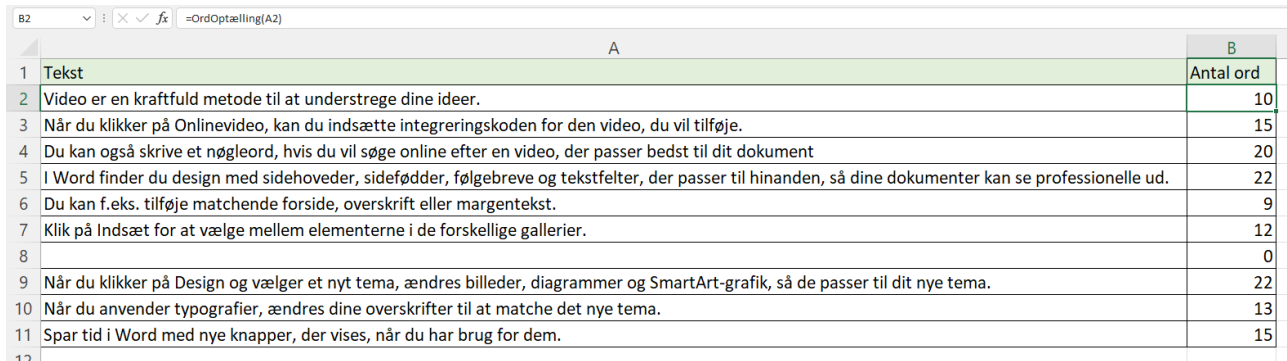
For at opdatere den eksisterende navngivne LAMBDA-formel skal vi igen bruge navneadministratoren:

1. Åbn Navnstyringen.
2. Vælg navnet " OrdOptælling" og klik på "Rediger".
3. Erstat koden "Refererer til" med denne formel:

```
=LAMBDA ( tekst ; LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) - LÆNGDE ( UDSKIFT ( A2 ; " " ; "" ) ) + ( LÆNGDE ( FJERN . OVERFLØDIGE . BLANKE ( A2 ) ) > 0 ) )
```

Når Navnstyringen er lukket, fungerer OrdOptælling korrekt på tomme celler, som det ses nedenfor:

Excel LAMBDA funktion



| | A | B |
|----|--|-----------|
| 1 | Tekst | Antal ord |
| 2 | Video er en kraftfuld metode til at understrege dine ideer. | 10 |
| 3 | Når du klikker på Onlinevideo, kan du indsætte integreringskoden for den video, du vil tilføje. | 15 |
| 4 | Du kan også skrive et nøgleord, hvis du vil søge online efter en video, der passer bedst til dit dokument | 20 |
| 5 | I Word finder du design med sidehoveder, sidefødder, følgebrev og tekstfelter, der passer til hinanden, så dine dokumenter kan se professionelle ud. | 22 |
| 6 | Du kan f.eks. tilføje matchende forside, overskrift eller margentekst. | 9 |
| 7 | Klik på Indsæt for at vælge mellem elementerne i de forskellige gallerier. | 12 |
| 8 | | 0 |
| 9 | Når du klikker på Design og vælger et nyt tema, ændres billeder, diagrammer og SmartArt-grafik, så de passer til dit nye tema. | 22 |
| 10 | Når du anvender typografier, ændres dine overskrifter til at matche det nye tema. | 13 |
| 11 | Spar tid i Word med nye knapper, der vises, når du har brug for dem. | 15 |

Bemærk: ved at opdatere koden én gang i Navnstyringen, opdateres alle forekomster af OrdOptælling-formlen på én gang. Dette er en vigtig fordel ved brugerdefinerede funktioner, der er oprettet med LAMBDA – formelopdateringer kan administreres ét sted.

Find rækketotaler med LAMBDA

| | A | B | C | D | E | F | G | H | I | J |
|---|-----------|--------|---------|--------|---------|--------|--------|--------|---|-------|
| 1 | By | Mandag | Tirsdag | Onsdag | Torsdag | Fredag | Lørdag | Søndag | | Total |
| 2 | København | 171 | 467 | 433 | 140 | 322 | 143 | 202 | | 1878 |
| 3 | Odense | 293 | 287 | 245 | 102 | 338 | 351 | 457 | | 2073 |
| 4 | Århus | 139 | 192 | 304 | 110 | 211 | 121 | 440 | | 1517 |
| 5 | Esbjerg | 257 | 280 | 151 | 457 | 446 | 184 | 455 | | 2230 |
| 6 | Ålborg | 362 | 382 | 103 | 314 | 258 | 488 | 397 | | 2304 |
| 7 | Randers | 495 | 149 | 478 | 175 | 263 | 291 | 235 | | 2086 |
| 8 | Silkeborg | 315 | 275 | 398 | 476 | 416 | 442 | 128 | | 2450 |

Generisk formel

```
=BYROW(område;LAMBDA(række;SUM(række)))
```

Resumé

For at få en matrix af rækketotaler baseret på række af numeriske værdier, kan du bruge en formel baseret BYROW-funktion sammen med funktionerne LAMBDA og SUM. I det viste eksempel er formlen i J2:

```
=BYROW(data;LAMBDA(rækken;SUM(rækken)))
```

hvor data er det navngivne område B2:H8. Resultatet er en matrix med syv summer, en for hver række i området, som det ses i kolonne J.

Bemærk: I ældre versioner af Excel kan du bruge MPRODUKT-funktionen, som forklaret nedenfor.

Forklaring

I dette eksempel er målet at returnere en matrix med syv subtotaler, én for hver af byerne nævnt i kolonne A. Tallene, der skal summeres, er indeholdt i data, som er det navngivne område B2:H8. Dette er et eksempel på en opgave, hvor målet er at skabe en matrix af summer i stedet for en enkelt sum. Vi kan ikke bruge en funktion som SUM alene, fordi SUM vil aggregere resultater og returnere en enkelt værdi. I artiklen nedenfor ser vi på to tilgange, en baseret på BYROW-funktionen og en baseret på MMULT-funktionen.

Med funktionen BYROW

I Excel 365 er den mest ligetil måde at generere subtotaler for hver række på med BYROW-funktionen. Formålet med BYROW er at behandle data på en "rækkevis" måde. For eksempel, hvis BYROW får et array med 7 rækker, vil BYROW returnere enkelt array med 7 resultater. I det viste eksempel er formlen i J2:

```
=BYROW(data;LAMBDA(rækken;SUM(rækken)))
```

Beregningen udført på hver række leveres af en brugerdefineret LAMBDA-funktion, som skal returnere et enkelt resultat for hver række. I dette eksempel summerer LAMBDA-funktionen, der bruges i BYROW, hver række således:

```
LAMBDA(rækken;SUM(rækken)) // summer hver række
```

Resultatet er en række af summer, én pr. række, der løber ind i området J2:J8. Dette resultat er fuldt ud dynamisk. Hvis dataværdierne ændres, eller hvis dataområdet udvides eller trækkes sammen, opdateres outputtet fra BYROW efter behov. Selvom dette eksempel omhandler totaler, kan det samme mønster bruges til at beregne andre oplysninger om rækker, herunder max, min, gennemsnit osv. som dette:

Find rækketotaler med LAMBDA

```
=BYROW(data;LAMBDA(rækken;MAKS(rækken))) // maks
```

```
=BYROW(data;LAMBDA(række;MIN(række))) // min
```

```
=BYROW(data;LAMBDA(række;MIDDEL(række))) // gennemsnit
```

Med funktionen MPRODUKT

En anden måde at løse dette problem på er med MPRODUKT-funktionen, som udfører matrixmultiplikation. MPRODUKT tager to arrays, array1 og array2, her skal gælde, at antallet af kolonner i array1 er det samme som antallet af rækker i array2. Den resulterende matrix vil have samme antal rækker som den første matrix og det samme antal kolonner som den anden matrix. MPRODUKT-formlen ser sådan ud:

```
=MPRODUKT(--data;TRANSPONER((KOLONNE(data)^0)))
```

Det første array er simpelthen alle værdier i området data, det navngivne område B2:H8:

```
=MPRODUKT(--data
```

For at beskytte mod tomme celler, som vil få MPRODUKT til at vise #VÆRDI! fejl, bruger vi en dobbelt negativ (--) for at tvinge eventuelle tomme celler til at være nul.

Dernæst skal vi oprette array2. Det første array indeholder 7 kolonner, så vi skal bruge det andet array, der skal indeholde 7 rækker. Vi ønsker kun en enkelt kolonne med resultater, så det andet array skal være 7 rækker gange 1 kolonne (7 x 1). Fordi vi ikke ønsker at ændre nogen værdier, bør arrayet kun indeholde tallet 1 (dvs. multiplisering med 1 ændrer ikke den oprindelige værdi). Array2 genereres med TRANSPONER-funktionen og KOLONNE-funktionen som dette:

```
TRANSPONER((KOLONNE(data)^0))
```

Selvom den er lidt kryptisk, er denne syntaks ovenfor en smart måde at udføre opgaven på. Funktionen KOLONNE returnerer en 1 x 7 matrix af kolonnenumre:

```
KOLONNE(data) // returnerer {2,3,4,5,6,7,8}
```

Dernæst opløftes disse tal til nul'te potens med eksponentoperatoren (^), hvilket skaber en 1 x 7 matrix af 1ere:

```
KOLONNE(data)^0 // returnerer {1,1,1,1,1,1,1}
```

Og TRANSPONER-funktionen vender arrayet fra 1 x 7 til 7 x 1:

```
TRANSPONER({1;1;1;1;1;1;1}) // returnerer {1;1;1;1;1;1;1}
```

Resultatet afleveres til MPRODUKT-funktionen som array2. MPRODUKT-funktionen udfører derefter matrixmultiplikation med de to arrays og returnerer en subtotal for hver række:

```
=MPRODUKT(--data;{1;1;1;1;1;1;1})
```

returnerer arrayet:

```
{1878;2073;1517;2230;2304;2086;2450}
```

Disse værdier returneres til celle J2 og overføres til området J2:J8.

Alternativt kan bruges SEKvens

En anden måde at konstruere array2 inde i MPRODUKT er med SEKvens-funktionen som denne:

Find rækketotaler med LAMBDA

```
=MPRODUKT(--data;SEKVENS(KOLONNER(data);1;1;0))
```

Denne formel fungerer på samme måde, men array2 oprettes med funktionen SEQUENCE direkte:

```
SEKVENS(KOLONNER(data);1;1;0) // returnerer {1;1;1;1;1;1;1}
```

Bemærk, at vi bruger funktionen KOLONNER til at fortælle SEKVENS, hvor mange rækker der skal oprettes (7).

Excel funktionen LET

| | A | B | C | D | E | F |
|----|-------|------------|---|------------|---|---|
| 1 | Start | 01-02-2022 | | Datoer | | |
| 2 | Slut | 18-02-2022 | | 01-02-2022 | | |
| 3 | | | | 02-02-2022 | | |
| 4 | | | | 03-02-2022 | | |
| 5 | | | | 04-02-2022 | | |
| 6 | | | | 07-02-2022 | | |
| 7 | | | | 08-02-2022 | | |
| 8 | | | | 09-02-2022 | | |
| 9 | | | | 10-02-2022 | | |
| 10 | | | | 11-02-2022 | | |
| 11 | | | | 14-02-2022 | | |
| 12 | | | | 15-02-2022 | | |
| 13 | | | | 16-02-2022 | | |
| 14 | | | | 17-02-2022 | | |
| 15 | | | | 18-02-2022 | | |
| 16 | | | | | | |

Excel LET-funktionen lader dig definere navngivne variable i en formel. Der er to primære grunde til, at du måske ønsker at gøre dette:

- 1) For at forbedre ydeevnen ved at eliminere overflødige beregninger og
- 2) For at gøre mere komplekse formler nemmere at læse og skrive.

Formål

Tildel variable i formlen

Returværdi

Normalt formelresultat

Syntaks

```
=LET(navn1, værdi1, [navn2/værdi2], ..., resultat)
```

Argumenter

navn1 – Første navn, der skal tildeles. Skal begynde med et bogstav.

værdi1 – Værdien eller beregningen, der skal tildeles til navn 1.

navn2/værdi2 – [valgfrit] Andet navn og værdi. Indtastet som et par argumenter.

resultat – En beregning eller en variabel, der tidligere er beregnet.

Noter

funktionen LET lader dig definere navngivne variable i en formel. Der er to primære grunde til, at du måske ønsker at gøre dette:

- 1) for at forbedre ydeevnen ved at eliminere overflødige beregninger og

- 2) for at gøre mere komplekse formler nemmere at læse og skrive.

Når en variabel er navngivet, kan den tildeles en statisk værdi eller en værdi baseret på en beregning. Formlen kan derefter referere til en variabel ved navn så mange gange som nødvendigt, mens værdien af variabelen kun er defineret ét sted.

Variabler navngives og tildeles værdier i par, adskilt af semikolon (navn1;værdi1;navn2;værdi2 osv.). LET kan håndtere op til 126 navn/værdi-par, men kun første navn/værdi-par er påkrævet. Omfanget af hver variabel er den aktuelle LET-funktion og indlejrede funktioner nedenfor. Det endelige resultat er en beregning eller en variabel, der tidligere er beregnet. Resultatet fra LET vises altid som det sidste argument til funktionen.

De navne, der bruges i LET, skal begynde med et bogstav og skelner ikke mellem store og små bogstaver. Du kan bruge navne, der indeholder tal som "værdi1", "værdi2" osv., men navne som "AB1" og "AB2" vil mislykkes, fordi Excel vil fortolke navnene som en cellerreference. Mellemrumstegn og tegnsætningssymboler er ikke tilladt i navne, men understregningstegnet (_) kan bruges.

LET-funktionen kombineres ofte med LAMBDA-funktionen som en måde at gøre en kompleks formel nemmere at bruge. LAMBDA giver mulighed for at navngive en formel og genbruge den i et regneark som en brugerdefineret funktion.

Vigtige fordele

LET-funktionen giver tre vigtige fordele:

1. **Klarhed** - navngivning af variabler brugt i en formel kan gøre en kompleks formel meget lettere at læse og forstå.
2. **Forenkling** - at navngive og definere variabler kun ét sted hjælper med at eliminere redundans og de fejl, der opstår ved at have den samme kode mere end ét sted.
3. **Ydeevne** - eliminering af redundant kode betyder samlet set mindre beregningstid, da dyre beregninger kun behøver at forekomme én gang.

Eksempel 1

Nedenfor er den generelle formel for funktionen LET med en variabel:

```
=LET(x;100;x+1) // returnerer 101
```

Med en anden variabel:

```
=LET(x;100;y;50;x+y) // returnerer 150
```

Efter at x og y er blevet erklæret og tildelt værdier, returnerer beregningen i det 5. argument 150.

Eksempel 2

En hovedfordel ved LET-funktionen er forenkling ved at eliminere redundans. For eksempel viser skærmbilledet ovenfor en formel, der bruger SEQUENCE-funktionen til at generere alle datoer mellem 1. maj 2020 og 15. maj 2020, som derefter filtreres af FILTER-funktionen til kun at inkludere ugedage. Formlen i E5 er:

```
=LET(datoer;SEKvens(B2-B1+1;1;B1;1);FILTRER(datoer;UGEDAG(datoer;2)<6))
```


Excel LET-funktionen

Det første argument erklærer variabelen datoer, og det andet argument tildeler output fra SEKVENS til datoer:

```
=LET(datoer;SEKVENS(B2-B1+1;1;B1;1))
```

Bemærk, at start- og slutdatoerne kommer fra henholdsvis cellerne B1 og B2. Når **datoer** er blevet tildelt en værdi, kan den bruges i den endelige beregning, som er baseret på funktionen FILTRER:

```
FILTRER(datoer;UGEDAG(datoer;2)<6) // filtrer weekender fra
```

Bemærk **datoer** bruges to gange: én gang med FILTRER og en gang med funktionen UGEDAG. I første omgang overføres de rå datoer fra SEKVENS til funktionen FILTRER som det array, der skal filtreres. I andet tilfælde overføres datoerne fra SEKVENS til funktionen UGEDAG, som kontrollerer for ugedage (dvs. ikke lørdag eller søndag). Resultatet fra UGEDAG er den logik, der bruges til at filtrere de originale datoer.

Uden funktionen LET ville SEKVENS skulle vises to gange i formlen, begge gange med den samme (redundante) konfiguration. Funktionen LET tillader, at funktionen SEKVENS kun vises og konfigureres én gang i formlen.

Dynamiske arrayformler i Excel

Dynamiske arrays er den største ændring af Excel-formler i årevis. Måske den største forandring nogensinde. Dette skyldes, at dynamiske arrays nemt lader dig arbejde med flere værdier på samme tid i en formel. Denne artikel giver et overblik med flere eksempler.

Den vigtigste fordel ved dynamiske arrays er evnen til at arbejde med flere værdier på samme tid i en formel. Dette er en stor opgradering og velkommen forandring. Dynamiske arrays løser nogle virkelig svære problemer i Excel og vil fundamentalt ændre den måde, regneark er designet på. Når du først har set, hvordan de fungerer, vil du aldrig vende tilbage.

Tilgængelighed

Dynamiske arrays og de nye funktioner nedenfor er kun tilgængelige Excel 365 og Excel 2021. Excel 2019 og tidligere tilbyder ikke dynamiske array-formler. For nemheds skyld bruger jeg "dynamiske arrays" (Excel 365) og "tidligere Excel" (2019 eller tidligere) til at skelne mellem versioner nedenfor.

Nye funktioner

Som en del af den dynamiske array-opdatering indeholder Excel nu 8 nye funktioner, som direkte udnytter dynamiske arrays til at løse problemer, der traditionelt er svære at løse med konventionelle formler:

| Funktion | Formål |
|---------------------|--|
| FILTRER | Filtrer data og returner matchende poster |
| SLUMPMATRIX | Generer matrix af tilfældige tal |
| SEKVEN | Generer matrix af sekventielle numre |
| SORTER | Sortér område efter kolonne |
| SORTER.EFTER | Sorter området efter et andet område eller array |
| ENTYDIGE | Udtræk unikke værdier fra en liste eller et område |
| XOPSLAG | Moderne erstatning for VLOOKUP |
| XSAMMENLIGN | Moderne erstatning for funktionen SAMMENLIGN |

Bemærkninger:

1. XOPSLAG og XSAMMENLIGN var ikke med i den oprindelige gruppe af nye dynamiske array-funktioner, men de kører godt sammen med den nye dynamiske array-motor. XOPSLAG erstatter LOPSLAG og tilbyder en moderne, fleksibel tilgang, der udnytter arrays. XSAMMENLIGN er en opgradering af SAMMENLIGN-funktionen, der giver nye muligheder til INDEX og SAMMENLIGN formler.
2. Funktionen LET er nu også en del af de nye dynamiske array-funktioner.
3. Flere funktioner er på vej.

Eksempel

Før vi går i detaljer, lad os så først se på et simpelt eksempel. Nedenfor bruger vi den nye funktion ENTYDIGE til at udtrække unikke værdier fra området B5:B15 med en enkelt formel indtastet i D2:

Dynamiske arrayformler i Excel

=ENTYDIGE(A2:A10) // returnerer unikke værdier i A2:A10

| | A | B | C | D |
|----|--------|------------|---|--------|
| 1 | Farve | Dato | | Farver |
| 2 | Rød | 01-08-2022 | | Rød |
| 3 | Grøn | 06-08-2022 | | Grøn |
| 4 | Blå | 10-08-2022 | | Blå |
| 5 | Gul | 11-08-2022 | | Gul |
| 6 | Grøn | 13-08-2022 | | Orange |
| 7 | Blå | 18-08-2022 | | |
| 8 | Rød | 19-08-2022 | | |
| 9 | Orange | 24-08-2022 | | |
| 10 | Gul | 29-08-2022 | | |
| 11 | | | | |

Resultatet er en liste over de fem unikke farver, som optræder i A2:A10.

Som alle formler opdateres ENTYDIGE automatisk, når data ændres. Nedenfor er Gul i A10 erstattet Violet. Resultatet fra ENTYDIGE inkluderer nu Violet:

| | A | B | C | D |
|----|--------|------------|---|--------|
| 1 | Farve | Dato | | Farver |
| 2 | Rød | 01-08-2022 | | Rød |
| 3 | Grøn | 06-08-2022 | | Grøn |
| 4 | Blå | 10-08-2022 | | Blå |
| 5 | Gul | 11-08-2022 | | Gul |
| 6 | Grøn | 13-08-2022 | | Orange |
| 7 | Blå | 18-08-2022 | | Violet |
| 8 | Rød | 19-08-2022 | | |
| 9 | Orange | 24-08-2022 | | |
| 10 | Violet | 29-08-2022 | | |
| 11 | | | | |

Overløb - én formel, mange værdier

I dynamisk Excel vil formler, der returnerer flere værdier, "overløber" disse værdier direkte i regnearket. Dette vil umiddelbart være mere logisk for formelbrugere. Det er også en fuldt dynamisk adfærd - når kildedata ændres, vil overløb resultater straks opdateres.

Det rektangel, der omslutter værdierne, kaldes "overløbs området". Du vil bemærke, at overløbs området har en særlig fremhævelse. I eksemplet med ENTYDIGE ovenfor er overløbs området D2:D7.

Dynamiske arrayformler i Excel

Når data ændres, vil overløbs området udvides eller trække sig sammen efter behov. Du kan muligvis se nye værdier tilføjet, eller eksisterende værdier forsvinde. På denne måde er et overløbs området et slags nyt dynamisk område.

I tidligere Excel kan du derimod se flere resultater returneret af matrixformlen i formellinjen, hvis du bruger F9 til at inspicere formelen. Medmindre formelen er indtastet som en flercellet matrixformel, vises kun én værdi i regnearket. Denne adfærd har altid gjort matrixformler svære at forstå. Overløb gør matrixformler mere intuitive.

Bemærk: når overløb er blokeret af andre data, vil du se en #OVERLØB-fejl. Når du har gjort plads til overløbs området, overløbes formelen automatisk.

Reference for overløb område

For at henvise til et overløbs område skal du bruge et hash-tegn (#) efter den første celle i området. For at referere til resultaterne fra ENTYDIGE-funktionen ovenfor, skal du f.eks. bruge:

```
=D2# // refererer ENTYDIGE resultater
```

Dette er det samme som at henvise til hele overløbs området, og du vil se denne syntaks, når du skriver en formel, der refererer til et komplet overløbs område.

Du kan føre en reference for et overløbs område direkte ind i andre formler. For at tælle antallet af farver returneret af ENTYDIGE kan du f.eks. bruge:

```
=TÆLV(D2#) // antal unikke farver
```

| | A | B | C | D | E | F |
|----|--------|------------|---|--------|---|-------|
| 1 | Farve | Dato | | Farver | | Antal |
| 2 | Rød | 01-08-2022 | | Rød | | 6 |
| 3 | Grøn | 06-08-2022 | | Grøn | | |
| 4 | Blå | 10-08-2022 | | Blå | | |
| 5 | Gul | 11-08-2022 | | Gul | | |
| 6 | Grøn | 13-08-2022 | | Orange | | |
| 7 | Blå | 18-08-2022 | | Violet | | |
| 8 | Rød | 19-08-2022 | | | | |
| 9 | Orange | 24-08-2022 | | | | |
| 10 | Violet | 29-08-2022 | | | | |
| 11 | | | | | | |

Når overløbs området ændres, vil formelen afspejle de seneste data.

Massiv forenkling

Tilføjjelsen af nye dynamiske matrixformler betyder, at visse formler kan forenkles drastisk. Her er et par eksempler:

- Udtræk og angiv unikke værdier

Dynamiske arrayformler i Excel

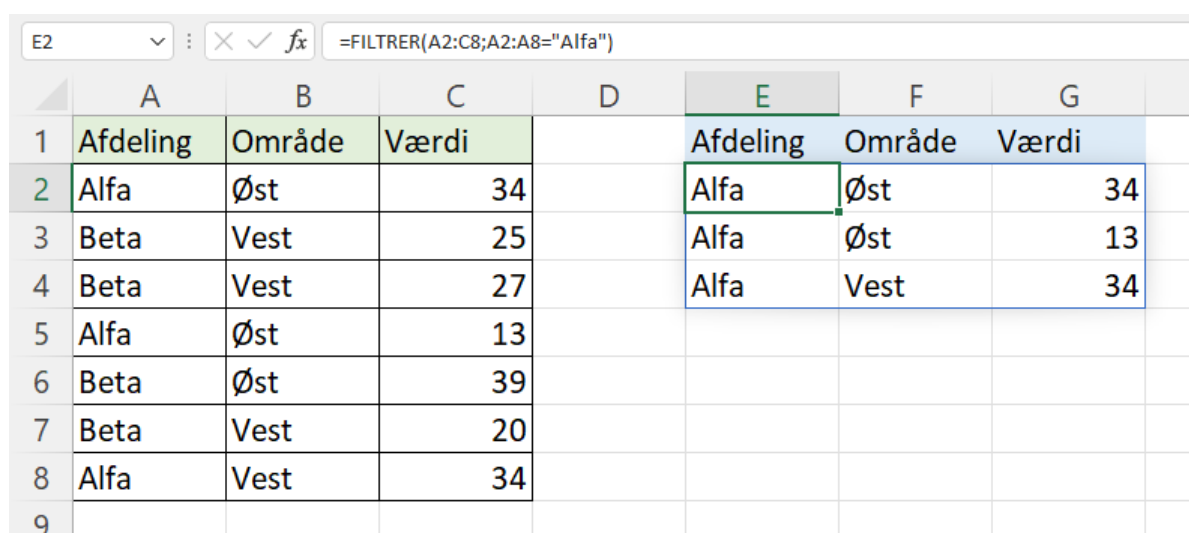
- Tæl unikke værdier
- Filtrer og udtræk poster
- Uddrag delvise sammenligninger

Den enes magt

En af de mest kraftfulde fordele ved tilgangen "én formel, mange værdier" er mindre afhængighed af absolutte eller blandede referencer. Da en dynamisk matrixformel overløber resultater i regnearket, forbliver referencer uændrede, men formlen genererer korrekte resultater.

For eksempel, nedenfor bruger vi FILTERER-funktionen til at udtrække poster i gruppen "Afdeling". I celle E2 indtastes en enkelt formel:

```
=FILTERER(A2:C8;A2:A8="Alfa") // referencerne er relative
```



| | A | B | C | D | E | F | G |
|---|----------|--------|-------|---|----------|--------|-------|
| 1 | Afdeling | Område | Værdi | | Afdeling | Område | Værdi |
| 2 | Alfa | Øst | 34 | | Alfa | Øst | 34 |
| 3 | Beta | Vest | 25 | | Alfa | Øst | 13 |
| 4 | Beta | Vest | 27 | | Alfa | Vest | 34 |
| 5 | Alfa | Øst | 13 | | | | |
| 6 | Beta | Øst | 39 | | | | |
| 7 | Beta | Vest | 20 | | | | |
| 8 | Alfa | Vest | 34 | | | | |
| 9 | | | | | | | |

Bemærk, at begge områder er ulåste relative referencer, men formlen fungerer perfekt.

Dette er en stor fordel for mange brugere, fordi det gør processen med at skrive formler meget enklere. Et andet godt eksempel, kan du se i multiplikationstabellen længere nede.

Kædefunktioner

Ting bliver virkelig interessante, når du kæder mere end én dynamisk array-funktion sammen. Måske vil du sortere de returnerede resultater efter ENTYDIGE? Nemt. Indsæt bare funktionen ENTYDIGE i funktionen SORTER på denne måde:

Dynamiske arrayformler i Excel

| | A | B | C | D |
|----|--------|------------|---|--------|
| 1 | Farve | Dato | | Farver |
| 2 | Rød | 01-08-2022 | | Blå |
| 3 | Grøn | 06-08-2022 | | Grøn |
| 4 | Blå | 10-08-2022 | | Gul |
| 5 | Gul | 11-08-2022 | | Orange |
| 6 | Grøn | 13-08-2022 | | Rød |
| 7 | Blå | 18-08-2022 | | Violet |
| 8 | Rød | 19-08-2022 | | |
| 9 | Orange | 24-08-2022 | | |
| 10 | Violet | 29-08-2022 | | |
| 11 | | | | |

Som før, når kildedata ændres, vises der automatisk nye unikke resultater, pænt sorteret.

Medfødt adfærd

Det er vigtigt at forstå, at dynamisk array-adfærd er medfødt og dybt integreret. Når en formel returnerer flere resultater, overføres disse resultater til flere celler i regnearket. Dette inkluderer ældre funktioner, der ikke oprindeligt er designet til at arbejde med dynamiske arrays.

For eksempel, i tidligere Excel, hvis vi giver funktionen LÆNGDE en række tekstværdier, vil vi se et enkelt resultat. I dynamisk Excel, hvis vi giver LÆNGDE-funktionen en række værdier, vil vi se flere resultater. Billedet nedenfor viser den gamle adfærd til venstre og den nye adfærd til højre:

| | A | B | C | D | E | F | G |
|---|-----------------|---|------------|---|----------------|---|------------|
| 1 | Afdeling | | Antal tegn | | Afdeling | | Antal tegn |
| 2 | Alfa | | 4 | | Alfa | | 4 |
| 3 | Beta | | | | Beta | | 4 |
| 4 | Gamma | | | | Gamma | | 5 |
| 5 | | | | | | | |
| 6 | Tidligere Excel | | | | Dynamisk Excel | | |
| 7 | | | | | | | |

Dette er en meget stor ændring, der kan påvirke alle slags formler. For eksempel er funktionen LOPSLAG designet til at hente en enkelt værdi fra en tabel ved hjælp af et kolonneindeks. Men i dynamisk Excel, hvis vi giver LOPSLAG mere end ét kolonneindeks ved hjælp af en arraykonstant som denne:

```
=LOPSLAG("København";F5:H7;{1\2\3};0)
```

Dynamiske arrayformler i Excel

| | A | B | C | D | E | F | G | H |
|---|-----------|----------|-------|---|---|-----------|----------|-------|
| 1 | | | | | | | | |
| 2 | København | | | | | København | A | 500 |
| 3 | | | | | | | | |
| 4 | By | Afdeling | Værdi | | | By | Afdeling | Værdi |
| 5 | København | A | 500 | | | København | A | 500 |
| 6 | Odense | B | 750 | | | Odense | B | 750 |
| 7 | Århus | A | 200 | | | Århus | A | 200 |
| 8 | | | | | | | | |

Med andre ord, selvom LOPSLAG aldrig blev designet til at returnere flere værdier, kan det nu gøre det takket være den nye formelmotor i den dynamiske Excel.

Alle formler

Bemærk endelig, at dynamiske arrays fungerer med alle formler, ikke kun funktioner. I eksemplet nedenfor indeholder celle C5 en enkelt formel:

```
=A2:A11*B1:K1
```

Resultatet løber ind i et 10 gange 10 område, der omfatter 100 celler:

| | A | B | C | D | E | F | G | H | I | J | K | L |
|----|----|----|----|----|----|----|----|----|----|----|-----|---|
| 1 | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| 2 | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| 3 | 2 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | |
| 4 | 3 | 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 | |
| 5 | 4 | 4 | 8 | 12 | 16 | 20 | 24 | 28 | 32 | 36 | 40 | |
| 6 | 5 | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | |
| 7 | 6 | 6 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 | 60 | |
| 8 | 7 | 7 | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63 | 70 | |
| 9 | 8 | 8 | 16 | 24 | 32 | 40 | 48 | 56 | 64 | 72 | 80 | |
| 10 | 9 | 9 | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 | 90 | |
| 11 | 10 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 | |
| 12 | | | | | | | | | | | | |
| 13 | | | | | | | | | | | | |

Hvis tallene i intervallerne A2:A11 og A2: B1:K1 i sig selv er dynamiske arrays (dvs. oprettet med SEKVENS-funktionen), kan overløbs referenceoperatoren bruges på denne måde:

```
=A2#*B1# // returnerer samme 10 x 10 matrix
```

Dynamiske arrayformler i Excel

Arrays bliver mainstream

Med udrulningen af dynamiske arrays vil ordet "array" dukke op meget oftere. Faktisk kan du se "array" og "område" bruges næsten i flæng. Du vil se arrays i Excel omgivet af krøllede parenteser som dette:

```
{1\2\3} // vandret array
```

```
{1;2;3} // lodret array
```

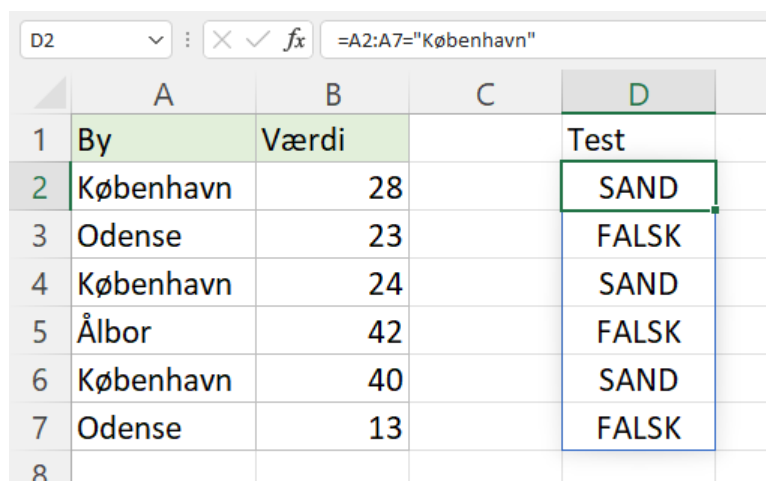
Array er et programmeringsudtryk, der refererer til en liste af elementer, der vises i en bestemt rækkefølge. Grunden til, at arrays dukker op så ofte i Excel-formler, er, at arrays kan udtrykke værdierne i en række celler.

Array-operationer bliver vigtige

Da dynamiske Excel-formler nemt kan arbejde med flere værdier, bliver matrixoperationer vigtigere.

Udtrykket "array-operation" refererer til et udtryk, der kører en logisk test eller matematisk operation på et array. For eksempel tester udtrykket nedenfor, om værdier i A2:A7 er lig med "København".

```
=A2:A7="København" // By = "København"
```



| | A | B | C | D |
|---|-----------|-------|---|-------|
| 1 | By | Værdi | | Test |
| 2 | København | 28 | | SAND |
| 3 | Odense | 23 | | FALSK |
| 4 | København | 24 | | SAND |
| 5 | Ålbør | 42 | | FALSK |
| 6 | København | 40 | | SAND |
| 7 | Odense | 13 | | FALSK |
| 8 | | | | |

fordi der er 6 celler i A2:A7, er resultatet 6 SAND/FALSK-værdier i en matrix:

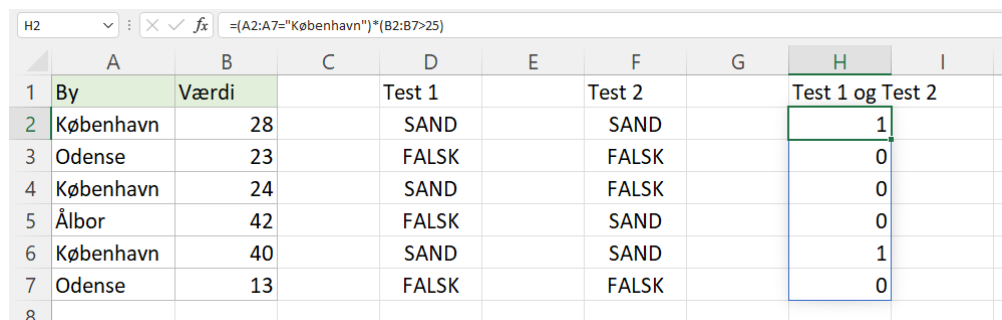
```
{SAND;FALSK;SAND;FALSK;SAND;FALSK}
```

Matrixoperationen nedenfor kontrollerer for værdier større end 25:

```
=B2:B7>25 // værdier > 25
```

Den endelige array-operation kombinerer første opgave og anden opgave i et enkelt udtryk:

```
=(A2:A7="København")*(B2:B7>25)
```



| | A | B | C | D | E | F | G | H | I |
|---|-----------|-------|---|--------|---|--------|---|------------------|---|
| 1 | By | Værdi | | Test 1 | | Test 2 | | Test 1 og Test 2 | |
| 2 | København | 28 | | SAND | | SAND | | 1 | |
| 3 | Odense | 23 | | FALSK | | FALSK | | 0 | |
| 4 | København | 24 | | SAND | | FALSK | | 0 | |
| 5 | Ålbør | 42 | | FALSK | | SAND | | 0 | |
| 6 | København | 40 | | SAND | | SAND | | 1 | |
| 7 | Odense | 13 | | FALSK | | FALSK | | 0 | |
| 8 | | | | | | | | | |

Dynamiske arrayformler i Excel

Bemærk: Excel tvinger automatisk værdierne SAND og FALSK til 1 og 0 ved en matematikoperation.

For at bringe dette tilbage til dynamiske matrixformler i Excel, viser eksemplet nedenfor, hvordan vi kan bruge nøjagtig den samme matrixoperation inde i FILTERER-funktionen som inkluderingsargumentet:

| D2 : <input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> fx =FILTERER(A2:B7;(A2:A7="København")*(B2:B7>25)) | | | | | |
|--|-----------|-------|---|-----------|-------|
| | A | B | C | D | E |
| 1 | By | Værdi | | By | Værdi |
| 2 | København | 28 | | København | 28 |
| 3 | Odense | 23 | | København | 40 |
| 4 | København | 24 | | | |
| 5 | Ålbør | 42 | | | |
| 6 | København | 40 | | | |
| 7 | Odense | 13 | | | |

FILTERER returnerer de to poster, hvor By = "København" og værdi > 25.

Nye og gamle matrixformler

I den dynamiske Excel er der ingen grund til at indtaste matrixformler med Ctrl + Shift + Enter. Når en formel oprettes, kontrollerer Excel, om formelen muligvis returnerer flere værdier. Hvis det er tilfældet, vil den automatisk blive gemt som en dynamisk matrixformel, men du vil ikke se krøllede parenteser.

Eksemplet nedenfor viser en typisk matrixformel indtastet i dynamisk Excel:

| F1 : <input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> fx =MAKS(B2:B8-C2:C8) | | | | | | |
|---|---------|------------|------------|---|------|-----|
| | A | B | C | D | E | F |
| 1 | Dag | Afdeling 1 | Afdeling 2 | | Maks | 118 |
| 2 | Mandag | 54 | 77 | | Min | -74 |
| 3 | Tirsdag | 75 | 90 | | | |
| 4 | Onsdag | 131 | 171 | | | |
| 5 | Torsdag | 83 | 157 | | | |
| 6 | Fredag | 200 | 82 | | | |
| 7 | Lørdag | 76 | 94 | | | |
| 8 | Søndag | 136 | 71 | | | |

Hvis du åbner den samme formel i en tidligere Excel, vil du se krøllede parenteser:

Dynamiske arrayformler i Excel

| | A | B | C | D | E | F |
|---|---------|------------|------------|---|------|-----|
| 1 | Dag | Afdeling 1 | Afdeling 2 | | Maks | 118 |
| 2 | Mandag | 54 | 77 | | Min | -74 |
| 3 | Tirsdag | 75 | 90 | | | |
| 4 | Onsdag | 131 | 171 | | | |
| 5 | Torsdag | 83 | 157 | | | |
| 6 | Fredag | 200 | 82 | | | |
| 7 | Lørdag | 76 | 94 | | | |
| 8 | Søndag | 136 | 71 | | | |
| 9 | | | | | | |

Hvis du går den anden vej, når en "traditionel" matrixformel åbnes i dynamisk Excel, vil du se de krøllede parenteser i formellinjen. For eksempel viser skærmen nedenfor en simpel matrixformel i en tidligere Excel:

| | A | B | C |
|----|-------|---|----------|
| 1 | Værdi | | Resultat |
| 2 | 1 | | SAND |
| 3 | 2 | | |
| 4 | 3 | | |
| 5 | 4 | | |
| 6 | 5 | | |
| 7 | 6 | | |
| 8 | 7 | | |
| 9 | 8 | | |
| 10 | 9 | | |
| 11 | 10 | | |
| 12 | | | |

Men hvis du indtaster formlen igen uden ændringer, fjernes de krøllede parenteser, og formlen returnerer det samme resultat:

| | A | B | C |
|----|-------|---|----------|
| 1 | Værdi | | Resultat |
| 2 | 1 | | SAND |
| 3 | 2 | | |
| 4 | 3 | | |
| 5 | 4 | | |
| 6 | 5 | | |
| 7 | 6 | | |
| 8 | 7 | | |
| 9 | 8 | | |
| 10 | 9 | | |
| 11 | 10 | | |
| 12 | | | |

Konklusionen er, at matrixformler indtastet med Ctrl + Shift+ Enter (CSE) stadig fungerer for at opretholde kompatibilitet, men du bør ikke have behov for at indtaste arrayformler med CSE i dynamisk Excel.

Tegnet @

Med introduktionen af dynamiske arrays vil du se @-tegnet forekomme oftere i formler. Tegnet @ muliggør en adfærd kendt som "implicit skæringspunkt". Implicit skæring er en logisk proces, hvor mange værdier reduceres til én værdi.

I tidligere Excel er implicit skæring en tavs adfærd, der bruges (når det er nødvendigt) til at reducere flere værdier til et enkelt resultat i én celle. I dynamisk Excel er det typisk ikke nødvendigt, da flere resultater kan vises i regnearket. Når det er nødvendigt, aktiveres implicit skæringspunkt manuelt med tegnet @.

Når du åbner regneark, der er oprettet i en ældre version af Excel, kan du muligvis se @-tegnet tilføjet automatisk til eksisterende formler, der har potentialet til at returnere mange værdier. I tidligere Excel vil en formel, der returnerer flere værdier, ikke overløbes i regnearket. Tegnet @ fremtvinger den samme adfærd i dynamisk Excel, så formlen opfører sig på samme måde og returnerer det samme resultat, som den gjorde i den originale Excel-version.

Med andre ord tilføjes @ for at forhindre, at en ældre formel overløber med flere resultater i regnearket. Afhængigt af formlen kan du muligvis fjerne @-tegnet, og formlens opførsel ændres ikke.

Opsummering

- Dynamiske arrays vil gøre visse formler meget nemmere at skrive.
- Du kan nu let filtrere matchende data, sortere og udtrække unikke værdier med formler.
- Dynamiske array-formler kan kædes (indlejres) for at udføre ting som filtrering og sortering.
- Formler, der returnerer mere end én værdi, overløbes automatisk.
- Det er ikke nødvendigt at bruge Ctrl + Shift + Enter ved indtastningen af en matrixformel.

Dynamiske arrayformler i Excel

- Dynamiske matrixformler er kun tilgængelige i Excel 365.

Stikord

| | | | |
|--------------------------------|----|----------------------|----|
| #BEREGN! | 4 | LÆNGDE | 8 |
| #VÆRDI! | 12 | MPRODUKT | 12 |
| BYROW | 11 | Navnestyringen | 9 |
| Dynamiske arrayformler | 17 | Overløb | 19 |
| ERSTAT | 8 | parameter | 7 |
| FALSK | 24 | SAND | 24 |
| FJERN.OVERFLØDIGE.BLANKE | 8 | SEKVENSS | 12 |
| KOLONNE | 12 | TRANSPONER | 12 |
| kuglevolumen | 6 | TÆLV | 19 |
| Kædefunktioner | 20 | UDSKIFT | 8 |
| LAMBDA | 3 | UGEDAG | 16 |
| LET | 3 | VBA | 3 |
| LOPSLAG | 21 | | |