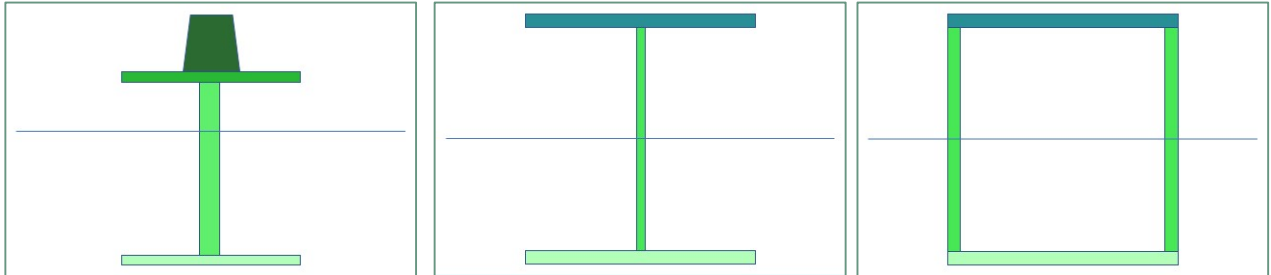


## Profiel opgebouwd met gestapelde blokken

In dit item bespreken we een voorbeeld van programmeren in VBA-code in Excel waarbij geometrische grootheden worden bepaald van een profiel dat is opgebouwd uit verschillende gestapelde geometrische figuren (vierkant, rechthoek, driehoek en trapezium).



Figuur 1: Voorbeelden van gestapelde profielen

In het bijbehorende Excel document (.xlsm) zijn verschillende technieken toegepast om de profielgrootheden te bepalen. Hiermee kunnen we een vergelijking tussen de verschillende technieken maken. In dit geval is een omvangrijke VBA-code geschreven om ook verschillende aspecten van het gebruik van invoerformulieren te demonstreren.

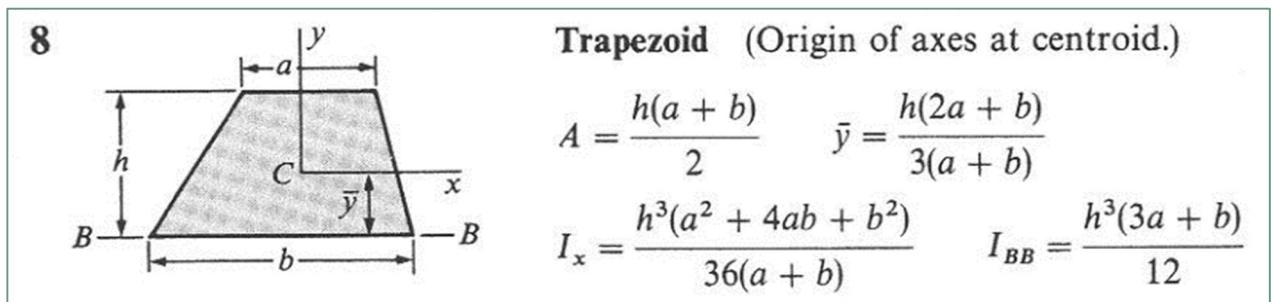
Eén van de belangrijkste onderdelen die we gaan zien is van het toepassen van grafische voorbeelden die automatisch aangepast worden afhankelijk van de gekozen invoer (figuur 1). De input van de gebruiker wordt direct omgezet naar een grafische voorstelling, zodat de gebruiker direct kan zien wat de gevolgen zijn van de gekozen input. Fouten door de gebruiker worden hiermee drastisch verminderd.

In deze toelichting op invoerformulieren in VBA worden verschillende aspecten benoemd om de grote voordelen van invoerformulieren te tonen. Uiteraard is het mogelijk om zelf wat vingeroefeningen doen met de Excel-sheet en de VBA-code.

Voor het gebruik van de Excel-sheet is het niet noodzakelijk deze toelichting te lezen. De toelichting zal echter wel helpen bij het begrip van invoerformulieren en de voordelen ervan en programmeren in VBA.

## De profielgrootheden

Het doel van de Excelsheet is de bepaling van enkele eenvoudige profielgrootheden van een profiel dat is opgebouwd uit een stapeling van geometrische vormen. Figuur 2 geeft de formules die zijn gebruikt.



Figuur 2: Toegepaste formules

Voor de eenvoud zijn deze formules gebruikt voor alle gebruikte geometrische vormen; driehoek, vierkant, rechthoek en trapezium. Het is in dit geval eenvoudig om een tabel op te stellen in Excel voor de invoer van de afmetingen van de verschillende vormen, blokken van het gestapelde profiel. In andere kolommen van



de tabel kunnen de berekeningen worden opgenomen voor de profielgrootheden op de gebruikelijke traditionele Excel methode. Dit principe is toegepast in één van de werkbladen in het Exceldocument (figuur 3).

Voor alleen de dikte en breedte in		ok = onderkant profiel (lijn BB, zie figuur)													
Nr	Dikte (h)	Breedte (b)	Breedte (a)	O <sub>i</sub>	A <sub>i</sub>	z <sub>i</sub>	I <sub>i</sub>	Y <sub>ok</sub>	Z <sub>i</sub> <small>tot ok</small>	Z-Z <sub>i</sub>	z <sub>i</sub> x A <sub>i</sub>	(z-z <sub>i</sub> ) <sup>2</sup> x A <sub>i</sub>	I <sub>i</sub>	O <sub>ok</sub>	
		Onder [mm]	Boven [mm]	[mm]	[mm <sup>2</sup> ]	[mm]	[mm <sup>4</sup> ]	[mm]	[mm]	[mm]	[mm <sup>2</sup> ]	[mm <sup>3</sup> ]	[mm <sup>4</sup> ]	[mm]	
1	9	160	160	338	1440	4,50	9.720	0	4,50	103,70	6.480	15.485.100	15.494.820	0	
2	152	6	6	316	912	76,00	1.755.904	9	85,00	23,20	77.520	490.845	2.246.749	6	
3	9	160	160	338	1440	4,50	9.720	161	165,50	-57,30	238.320	4.728.056	4.737.776	6	
4	25	50	45	145	1188	12,28	61.792	170	182,28	-74,08	216.458	6.517.067	6.578.859	50	
195	160			1013	4980						538.778	27.221.067	29.058.203	62	108,2
		$b_{proef}$		$O_{proef}$	$A_{proef}$								$I_{proef}$	$O_{ok}$	$ZP_{proef}$

Figuur 3: Een traditionele tabel in Excel (input gelijk aan linker profiel figuur 1)

Figuur 3 laat het betreffende deel van de berekening zien op de “traditionele Excel wijze”. De profielafmetingen en -grootheden, zoals hoogte, breedte, oppervlak, zwaartepunt en traagheidsmoment worden keurig en correct bepaald en weergegeven. Of het profiel uit één blokje bestaat of uit 15, alles is mogelijk. De tabel werkt prima en is in dit geval goed controleerbaar. Het gebruik van verschillende kolommen met tussenresultaten maakt dit mogelijk. Hier is het aantal tussenresultaten gelukkig beperkt en overzichtelijk.

### Eigen functies voor de berekeningen

Hetzelfde voorbeeld is ook gebruikt met eigen functies en een invoerformulier in VBA-code. De VBA-code voor het invoerformulier is bewust een stuk omvangrijker dan strikt noodzakelijk. Dit is het gevolg van “extra” die zijn ingebouwd om verschillende aspecten van invoerformulieren duidelijk te maken. Echter door het gebruik van een eenvoudig probleem, de bepaling profielgrootheden van een willekeurig gestapeld profiel, is het eenvoudig om te zien welke items in het formulier en in de VBA-code voor de berekening zijn en welke de “extra’s” vormen. De focus kan zich meer richten op de VBA-code i.p.v. de berekening.

Onder de traditionele tabel in het Exceldocument is een tweede tabel opgenomen waarbij dezelfde berekeningen zijn uitgevoerd, maar dan met behulp van eigen gemaakte functies. Het grote verschil tussen traditioneel en eigen functies wordt direct zichtbaar. Er zijn geen extra kolommen met tussenresultaten noodzakelijk en de formules in de cellen zijn veel eenvoudiger. Met eigen functies wordt alles veel compacter én inzichtelijker.

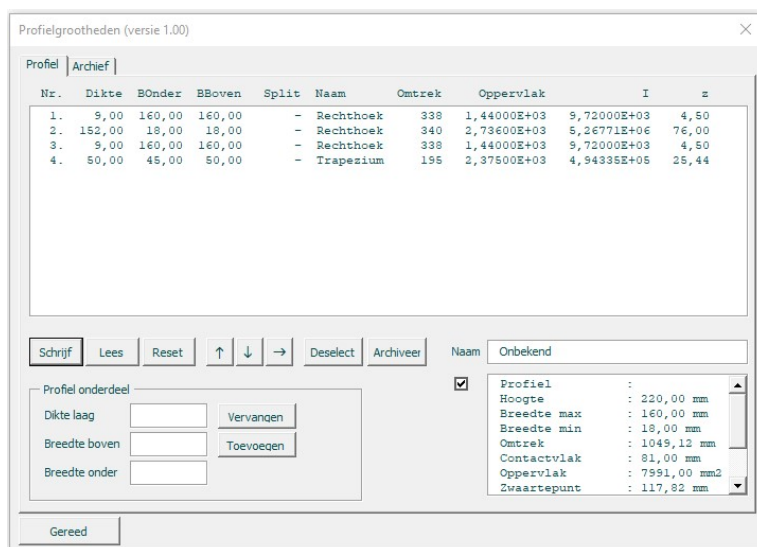
Uiteraard kan de traditionele tabel compacter worden gemaakt door de formules in de verschillende kolommen samen te voegen, maar dan wordt de kans op fouten groter en zijn de formules zeker niet meer leesbaar en/of oncontroleerbaar.

## Invoerformulier en VBA-code

We gaan de derde methode voor de bepaling van de profielgrootheden nader bekijken. Het gebruik van een invoerformulier in VBA. Er is geen gebruik gemaakt van berekeningen, of andere voorzieningen in Excel. In principe is het mogelijk om de VBA-code ook in Word of Powerpoint toe te passen. Het invoerformulier zou exact hetzelfde werken.

Er zijn echter paar kleine Excel eigenschappen gebruik in de VBA om het e.e.a. duidelijk te maken. Deze VBA onderdelen kunnen eenvoudig worden uitgeschakeld voor Word of Powerpoint. Voor de duidelijkheid, het is mogelijk om dit verschil in “software” direct in de VBA te verwerken. Er worden dan bepaalde delen van de VBA-code alleen gebruikt als het invoerformulier in Excel wordt toegepast en worden weer andere delen van de VBA-code gebruikt als hetzelfde invoerformulier in Word wordt toegepast. Delen van de VBA-code, zoals het “rekenhart” is voor beide gelijk.

In het formulier is de invoer van de losse blokken geregeld middels drie invoervelden: dikte, breedte onder en breedte boven. Evenals de drie kolommen in de traditionele Exceltabel. Na het invoeren van de blokgegevens, worden deze direct verwerkt in een overzicht (figuur 4). Daar worden automatisch alle profielgrootheden bepaald en zichtbaar gemaakt. Tevens wordt een grafische preview van het gestapelde profiel direct aangepast (op de Excelsheet). Naar wens kan allerlei informatie zichtbaar worden gemaakt die de gebruiker kan helpen bij zijn/haar taken.



Nr.	Dikte	B Onder	B Boven	Split	Naam	Omtrek	Oppervlak	I	z
1.	9,00	160,00	160,00	-	Rechthoek	338	1,44000E+03	9,72000E+03	4,50
2.	152,00	18,00	18,00	-	Rechthoek	340	2,73600E+03	5,26771E+06	76,00
3.	9,00	160,00	160,00	-	Rechthoek	338	1,44000E+03	9,72000E+03	4,50
4.	50,00	45,00	50,00	-	Trapezium	195	2,37500E+03	4,94335E+05	25,44

Figuur 4: Het invoerformulier

In dit geval is de optie ingebouwd dat bij het opstarten van het formulier automatisch de ingevoerde informatie uit de Excelsheet wordt overgenomen. Het is niet nodig om de invoergegevens opnieuw in te voeren in het formulier. De rekenresultaten en de preview worden direct verwerkt. De gebruiker kan beide invoermogelijkheden gebruiken, echter de berekening wordt hier volledig door VBA-code uitgevoerd. De VBA-code is zo ingesteld dat de gebruiker de informatie/invoer in de Excelsheet niet meer kan wijzigen zolang het formulier actief is. Alle verdere wijzigingen moeten in het formulier volgen, wat ook de essentie is van het formulier.

Door het formulier te starten worden de profielgrootheden direct berekend. De gebruiker kan allerlei acties uitvoeren met de gegevens en resultaten, waaronder de ingevoerde gegevens in de Excelsheet schrijven. Dit lijkt voor dit voorbeeld omslachtig, echter voor meer complexe berekeningen en formulieren geeft dit grote voordelen bieden, zie paragraaf “Plaatsing van gegevens in Excelsheet”.

Bij het opstellen van het formulier is ervoor gekozen om de grafische preview alleen zichtbaar te maken op de Excelsheet zelf, echter is het ook mogelijk om de preview zichtbaar te maken op het formulier, of bieden. In VBA zijn hiervoor verschillende technieken beschikbaar. De keuze is geheel aan de programmeur en zijn/haar creativiteit.

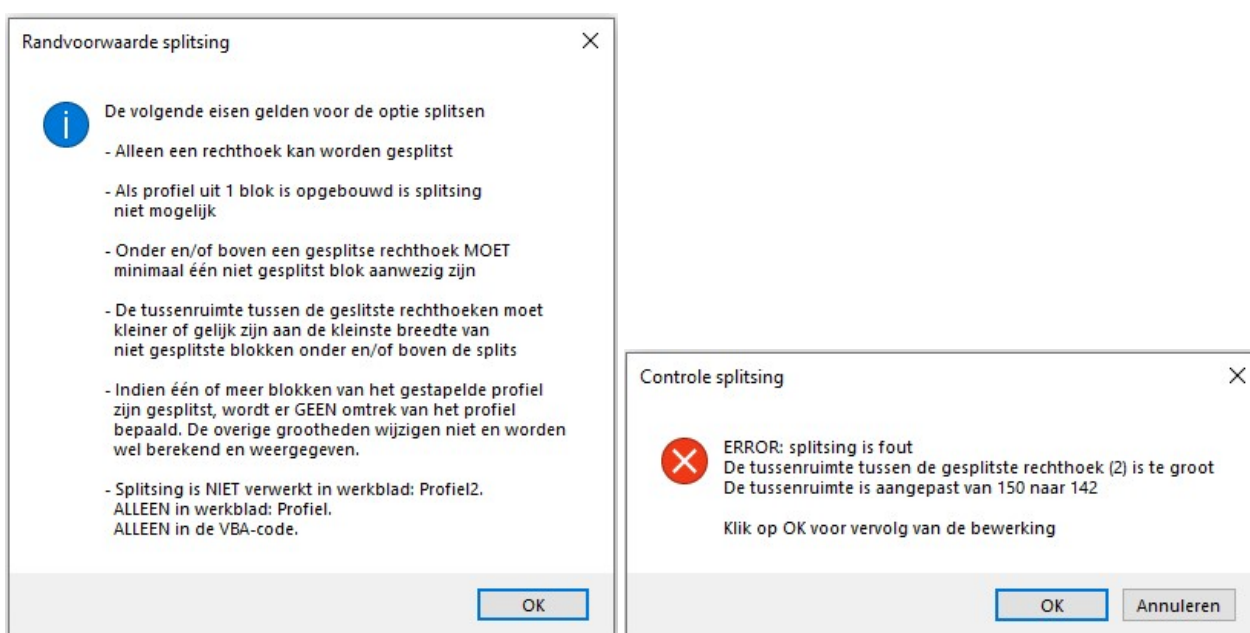
## Hulp bij de invoer van gegevens

In een formulier is het heel eenvoudig om “Tips” aan knoppen, invoervelden enz. toe te voegen. Deze “Tips” kunnen elke willekeurige vorm en inhoud hebben. De tips kunnen statische informatie weergeven aan de gebruiker over bijvoorbeeld de functie van de betreffende knop (figuur 5). Tips kunnen ook dynamisch zijn, ze kunnen de gebruiker informatie geven die gebaseerd is op de reeds ingevoerde invoer. Zo krijgt de gebruiker alleen hele specifieke informatie op die momenten dat dit nodig is; tijdens het klikken van een knop enz.



Figuur 5: Voorbeeld van het gebruik van “tips”

Nast de “Tips” kan in VBA ook gebruik worden gemaakt tekstvensters (pop-ups) zoals we die kennen uit allerlei Windows software. Deze vensters kunnen eenvoudig en op veel verschillende manieren worden uitgevoerd en weergegeven.



Figuur 6: Voorbeelden van pop-ups met specifieke informatie voor de gebruiker

Aan de inhoud van deze memo kunnen geen rechten worden ontleend. De gebruiker mag de inhoud vrij gebruiken en toepassen. Deze tekst wordt geleverd “as is”, zonder enige vorm van garantie. Geadviseerd wordt het gebruik van de inhoud van deze memo en de bepaling van de juistheid van de inhoud te bespreken met een zakenkundige binnen uw organisatie, zoals een systeembeheerders en/of leidinggevende.

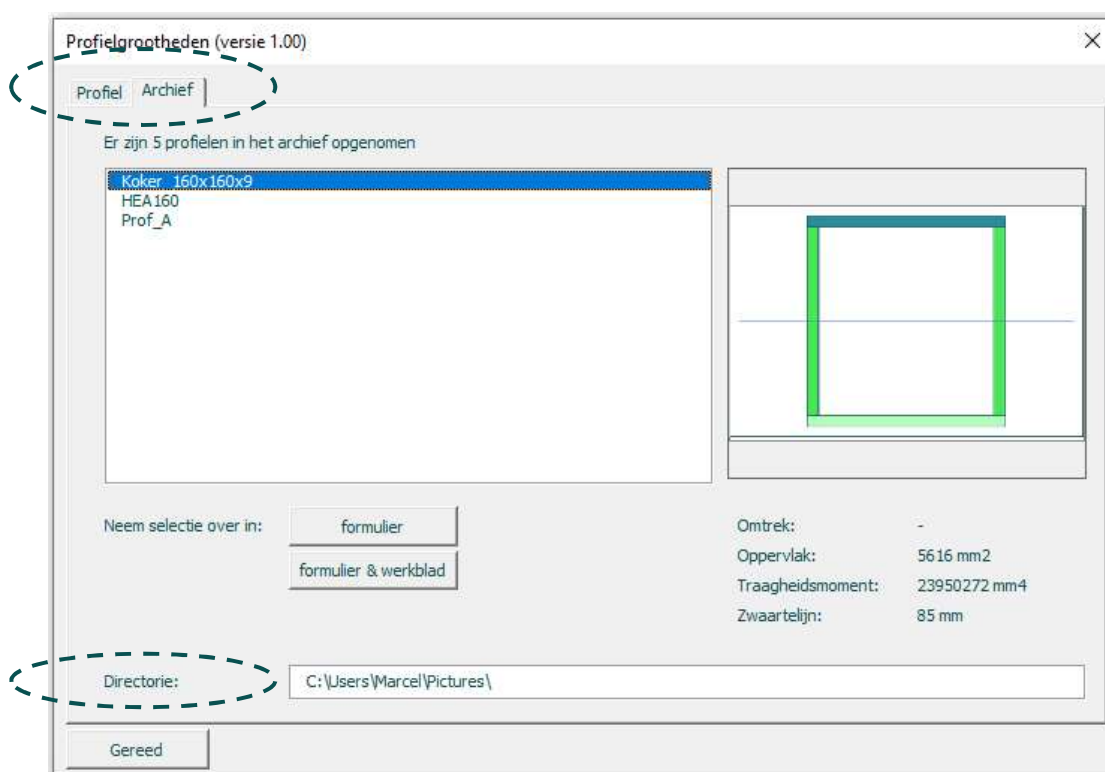
De inhoud kan geheel zelf worden bepaald. De vorm van de vensters kan geheel zelf worden bepaald. Het doel MOET altijd zijn, de gebruiker helpen bij het maken van beslissingen en het uitvoeren van taken. Deze vensters zijn alleen zichtbaar op de momenten dat het nodig is, alleen op die momenten dat de gebruiker dit wenst of automatisch als er fouten optreden.

Dergelijke extra informatie is in een traditionele Excelsheets ook mogelijk, echter zijn de mogelijkheden veel beperkter. In VBA is de creativiteit van de programmeur alleen de beperking.

### Default waarden

Naast het gebruik van “Tips” en “Vensters” is het ook mogelijk om gebruik te maken van default waarden in het formulier. Dit kunnen slimme/dynamische default waarden zijn, gebaseerd op allerlei ingevoerde informatie, gebaseerd op uitgevoerde berekeningen en/of analyses enz. Ook hier is de creativiteit van de programmeur de enige belemmering. Er zijn verschillende mogelijkheden om de default waarden te gebruiken. Dit kan ook worden gestuurd door de gebruiker zelf, zoals “Dubbelklik” op een invoervlak om de default waarde in te voeren (figuur 7: Directory). De gebruiker kan een eenvoudige actie uitvoeren om de default waarde over te nemen of naar wens een eigen waarde invoeren. De controle ligt volledig bij de gebruiker.

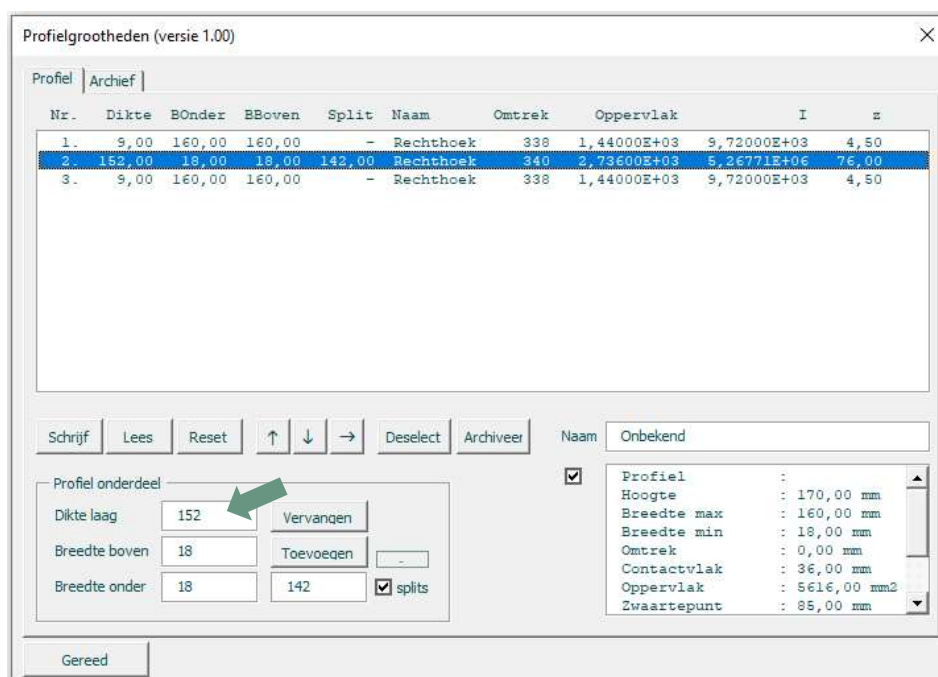
Uiteraard kunnen default waarden ook direct worden opgenomen in invoervakjes. Dit kan bij het opstarten van het formulier of nadat een bepaalde actie (klik) is uitgevoerd. In het geval van het formulier is dit duidelijk te zien als de gebruiker op één van de overzichten klikt, zoals in het Archief (figuur 7). Door op één van de regels in het overzicht te klikken worden allerlei gegevens van de betreffende selecties zichtbaar. Klikte de gebruiker op een andere regel worden de gegevens direct weer aangepast. In dit geval gaat het om een aantal profielgegevens en de preview van het betreffende profiel (figuur 7).



Figuur 7: Voorbeelden van het gebruik van default waarden in een formulier

In het invoervlak "Directory" wordt bij het opstarten een default directory gegeven (figuur 7). Deze is gekoppeld aan de persoon die op dat moment op de betreffende pc is ingelogd. Er wordt eenvoudig gebruik gemaakt van informatie die Windows beschikbaar stelt aan de programmeur. De programmeur heeft in VBA de beschikking over allerlei systeemgegevens. De systeemgegevens kunnen worden gebruikt voor een (specifieke) gebruiker en/of voor het gebruik van de applicatie zelf. Het doel is om alles zo eenvoudig mogelijk te maken. Eenvoud is het grote sleutelwoord bij het opstellen van invoerformulieren.

Nog een voorbeeld. In het onderdeel "Profiel" wordt hetzelfde gedaan met het overzicht. Door op één regel te klikken selecteer je de invoergegevens van één onderdeel van het profiel, een blok uit het profiel, en deze invoergegevens worden direct overgenomen in het formulier. Zie de invoervelden "Dikte" en "Breedte". Dit kan de gebruiker helpen bij het invoeren van nieuwe blokken, bijvoorbeeld als deze overeenkomsten hebben met andere, reeds ingevoerde blokken.



Figuur 8: Voorbeeld van de verwerking van gegevens op basis van een selectie

Stel dat de volgende blok een dikte heeft van 152 mm, een breedte boven van 18 mm en een breedte onder van 35 mm. Dan klik op regel 2, de informatie van 2 wordt automatisch overgenomen en de gebruiker hoeft alleen de waarde voor de Breedte onder van 18 naar 35 mm aan te passen en het blok toe te voegen. Dezelfde manier wordt gebruikt om fouten in de invoer te herstellen; bijvoorbeeld de 152 mm moet 240 mm zijn. Klik op de betreffende regel en pas de 152 mm aan.

## Controle van gegevens

Een van de belangrijkste voordelen van VBA-code in een invoerformulier is het controleren data en afvangen van fouten - kwaliteitsverbetering. Excel zelf heeft hiervoor enkele opties, maar dit is erg beperkt in verhouding tot de mogelijkheden van VBA. De enige beperking is de creativiteit van de programmeur.

Met VBA-code kun je alles controleren en kun je elke gewenste reactie geven, zoals een pop-up venster. Naast de invoergegevens is het ook mogelijk om rekenresultaten te controleren en acties uit te voeren als er een “afwijking” wordt “waargenomen”.

Uiteraard is het ook mogelijk om van de standaard Excel opties gebruikt te maken, zoals “voorwaardelijke opmaak”. Maar met VBA-code is het heel eenvoudig om naast een foutmelding ook een suggestie te geven die de gebruiker helpt bij het nemen van een beslissing of stuurt naar een mogelijke oplossing (figuur 9).



Figuur 9: Een foutmelding met een toelichting en een suggestie voor herstel

## Selectie van gegevens

Met een invoerformulier is het eenvoudig om data en invoer te groeperen, te filteren, in delen zichtbaar te maken enz. Dit alles moet de gebruiker helpen bij de uitvoering van taken. Laat alleen die informatie zien die op dat moment zinvol is en een bijdragen kan leveren in het gebruik en het correct uitvoeren van acties/deeltaken. Meer informatie zal de gebruiker afleiden en de kans op fouten vergroten. Deze techniek wordt in alle software toegepast. Het geeft overzicht en eenvoud.

In het formulier is gebruik gemaakt van twee tabs; Profiel en Archief. Profiel laat alleen gegevens zien die van toepassing zijn op het samenstellen van het huidige profiel. Alle overige info is niet zichtbaar. Tab Archief laat alleen reeds ingevoerde profielen zien. Ook hier alleen die gegevens die hier noodzakelijk zijn; naam van het profiel, een preview die heel snel laat zien of het juiste profiel is geselecteerd en enkele profielgegevens die de gebruiker de mogelijkheid geven om snel een juiste selectie te maken. Stel de gebruiker zoekt een profiel met een bepaald oppervlak of traagheid. Simpelweg klik op de verschillende regels in het overzicht. Uiteraard is het mogelijk om een eenvoudige zoekfunctie in het formulier op te nemen.

Met formulieren kan informatie op een hele specifieke manier zichtbaar worden, zoals we dit kennen van allerlei Windows applicaties. Er kan gebruik worden gemaakt van teksten, getallen, overzichten, figuren, grafieken enz.

## Plaatsing van gegevens in Excelsheet

De VBA-code kan informatie (terug) op een Excelsheet plaatsen. In het geval van de tabel met profielgegevens zijn dit de afmetingen van de blokken (dikte en breedte), evenals de resultaten van de berekeningen. Dit kan op twee verschillende manieren:

- Met getallen en teksten. Er worden/zijn geen formules in de cellen opgenomen/verwerkt. Daarna kan Excel de getallen en teksten gebruiken op de traditionele wijze. Er zijn geen beperkingen.

- Met formules. Met VBA-code is het mogelijk om een Excelsheet naar wens te vullen met typische Excelformules. De werking en gebruik is 100% gelijk aan alle traditionele Excel formules. Er is geen enkel verschil of beperking.

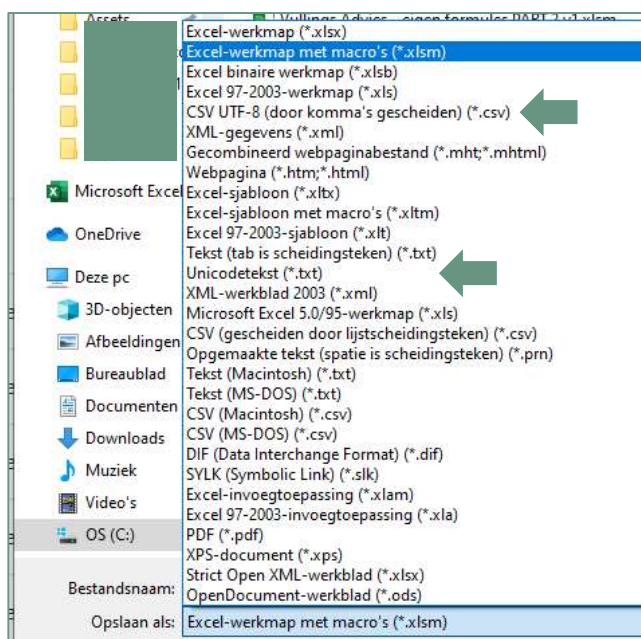
Een groot verschil tussen het gebruik van VBA-code versus een traditionele tabel is de wijze waarop de gebruiker gebruik maakt van traditionele tabellen. We gaan er even vanuit dat de gebruiker weet hoe de invoerdata (dikte en breedte van de blokken) kan invoeren in de sheet. Een probleem kan mogelijk ontstaan bij andere acties, zoals het verwijderen van data. Afhankelijk van de wijze waarop de data wordt verwijderd kunnen ook per ongeluk formules worden verwijderd. Of de bestaande formules in de sheet werken niet meer, de celverwijzingen zijn niet meer correct. Als men geluk heeft wordt dit direct duidelijk door foutmeldingen in één of meer cellen in de sheet. Erger is het als er geen meldingen worden gegeven. Dan zijn celverwijzingen mogelijk onjuist, maar bevatten de cellen wel data die wordt gebruikt, onjuiste data. Hetzelfde is mogelijk na het invoegen van regels in de tabel, in de Excelsheet. Dit kan gelijke problemen opleveren, formules die ontbreken en/of celverwijzingen die niet meer correct zijn.

De programmeur kan rekening houden met de mogelijkheden van Excel (zoals hierboven omschreven) en VBA-code schrijven die dergelijke problemen opvangen/oplossen. Dit kan bijvoorbeeld door de VBA-code onafhankelijk te maken van de inhoud van een sheet. Of de VBA-code kan zoeken naar de gewenste data in de sheet (niet op basis van celverwijzingen). De kans op fouten wordt zeker een heel stuk minder.

## Exporteren van gegevens

Excel kent verschillende manieren om gegevens op te slaan op een harde schijf. Deze gegevens kunnen door andere software ook worden gelezen en verwerkt. Daarmee is het mogelijk om data uit te wisselen tussen verschillende software pakketten. Het beste voorbeeld is het CSV formaat (comma separated value). Excel kan dergelijke bestanden lezen en schrijven, evenals andere software.

De formaat mogelijkheden worden zichtbaar als een bestand (of selectie) wordt opgeslagen "als". Bij "Opslaan als" wordt het menu zichtbaar, evenals de opties (figuur 10). Extra mogelijkheden zijn gegeven onder "Export" onder het item "Bestand".



Figuur 10: Overzicht van "Opslaan als"

Aan de inhoud van deze memo kunnen geen rechten worden ontleend. De gebruiker mag de inhoud vrij gebruiken en toepassen. Deze tekst wordt geleverd "as is", zonder enige vorm van garantie. Geadviseerd wordt het gebruik van de inhoud van deze memo en de bepaling van de juistheid van de inhoud te bespreken met een zakenkundige binnen uw organisatie, zoals een systeembeheerder en/of leidinggevende.





Rubriek Constructeurs programmeren

Item Kwaliteitsverbetering met eigen formulieren

Datum 19-02-2020

Pagina 9 | 9

Al deze mogelijkheden die Excel (en Word, Access, Outlook, Powerpoint, Project) kent kunnen ook via VBA-code worden toegepast. Er is geen enkel verschil, echter kan VBA nog veel meer andere formaten genereren. Alle formaten die gebaseerd zijn op ASCII (tekst bestanden) kunnen eenvoudig worden opgesteld met VBA. Daarmee kan de data in Excel worden verwerkt tot een extern bestand in een groot scala van formaten, waaronder ook XML, DXF, IFC enz. Er zijn ook formaten mogelijk die direct kunnen worden gelezen door (3D) printers, productiemachines enz. Ook hier geldt dat met VBA er heel veel mogelijkheden zijn dan met Excel.

## Vervolg

Om de grote voordelen van VBA (programmeren) duidelijk te maken zullen er binnenkort weer nieuwe voorbeelden op de website worden geplaatst ([www.VullingsAdvies.nl](http://www.VullingsAdvies.nl)). Daar zullen verschillende aspecten nog eens op een andere manier worden gebruikt en worden weer andere mogelijkheden zichtbaar gemaakt. Er worden voorbeelden gegeven voor het koppelen van software waardoor optimaal gebruik kan worden gemaakt van de specifieke pakketten. Zo kunnen we vanuit Excel resultaten direct verwerken in Word. Vanuit Word kunnen we gebruik maken van functies van Outlook zonder Outlook te openen. Vanuit Excel kunnen we naar Outlook en MS Project. Enz. Koppelingen zijn ook mogelijk buiten de Microsoft familie, zoals Autocad, AxisVM enz. Hou de website in de gaten voor nieuwe voorbeelden.

Momenteel wordt er gewerkt aan de “Bouwblok generator”, waarmee het mogelijk is voor grondgebonden woningen een willekeurig bouwblok samen te stellen op basis van een paar klikken. Er wordt automatisch allerlei informatie genereerd met behulp van VBA. Denk daarbij aan tekstbestanden met een overzicht van alle beschikbare data (geometrie), XML bestanden voor archivering en DXF bestanden voor Autocad. De gewichtsberekening wordt direct automatisch opgesteld. Ideaal voor zogenaamde “rijtjes woningen” enz. Uiteraard wordt het een simpele en vereenvoudigde versie, maar het geeft een heel goed beeld van de mogelijkheden én het voorbeeld kan worden gebruikt voor (eigen) verdere ontwikkelingen. Hou de website in de gaten.

*Marcel Vullings*