

# VISUAL BASIC FOR APPLICATIONS



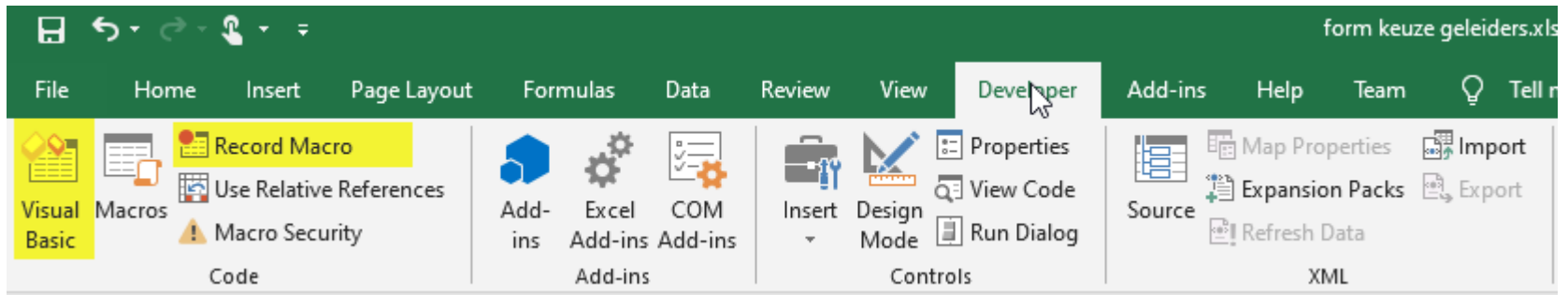
# DEVELOPER

The image shows the Microsoft Excel interface with the following elements highlighted:

- 1**: The **File** tab in the ribbon.
- 2**: The **Options** button in the left-hand navigation pane.
- 3**: The **Customize Ribbon** option in the left-hand pane of the Excel Options dialog.
- 4**: The **Developer** checkbox in the **Main Tabs** list of the Customize Ribbon dialog.

A blue arrow points from the **Developer** tab in the ribbon to the **Developer** checkbox in the dialog.

# DE MENUBALK



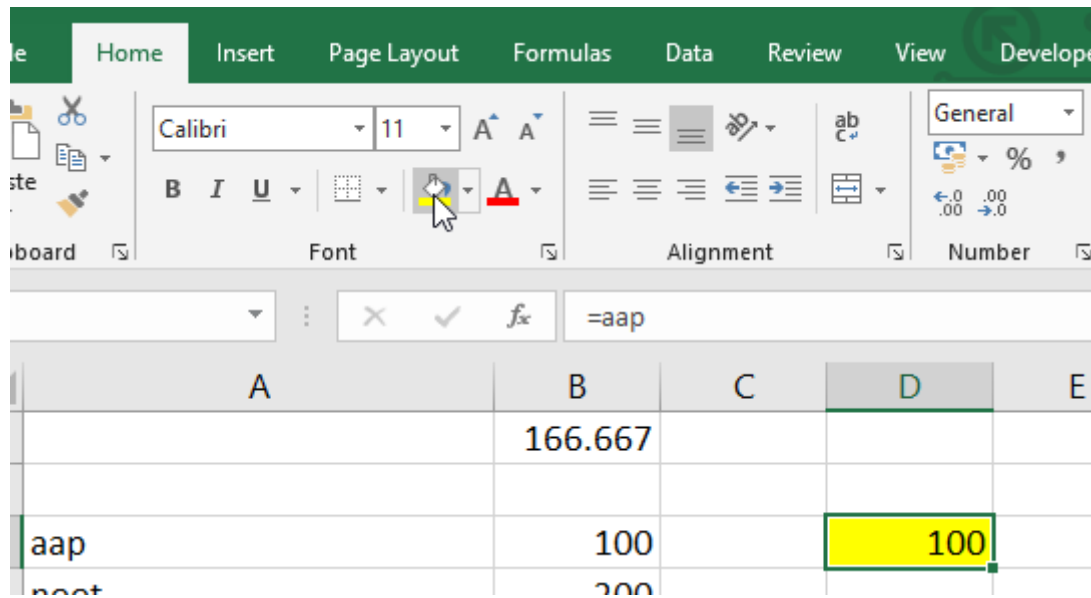
- De belangrijkste knoppen.
  - Developer
  - Visual Basic
  - Record Macro

# MACRO OPNEMEN

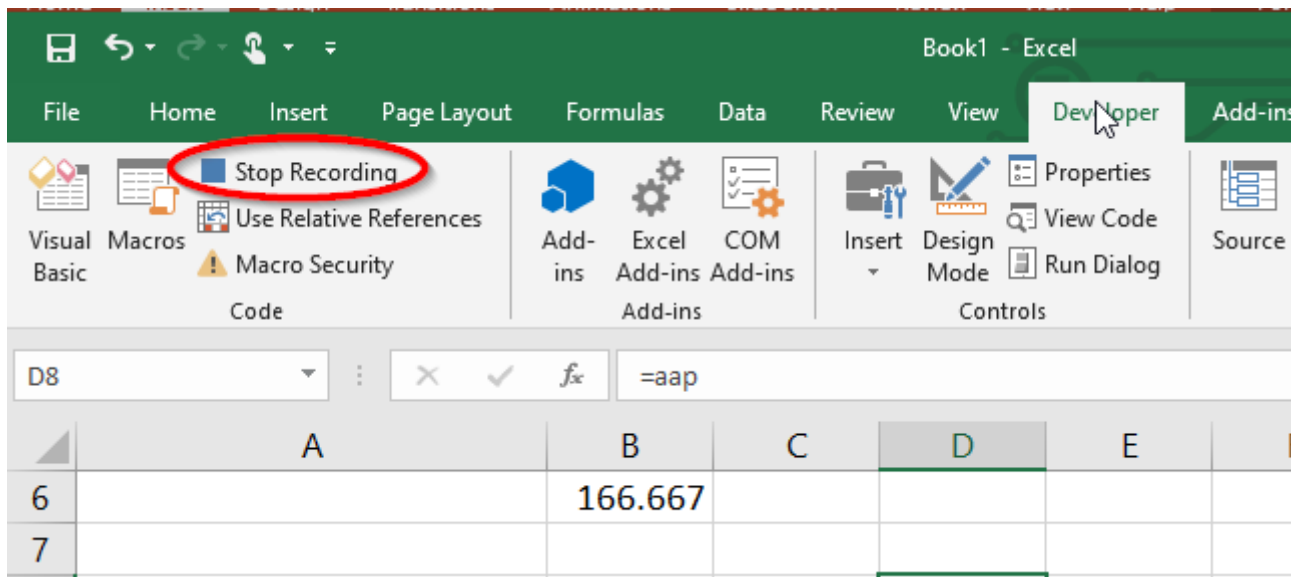
The image shows a screenshot of the Microsoft Excel interface. The 'Developer' tab is selected in the ribbon, and the 'Record Macro' button is circled in red. A blue arrow points from this button to the 'Record Macro' dialog box. In the dialog box, the 'Macro name' field contains the text 'Macro1', which is also circled in red. A yellow speech bubble with a red border points to the 'Shortcut key' field, containing the text 'Optioneel een letter om de macro te starten'. The background spreadsheet shows a grid with columns A through G and rows 6 through 18. The formula bar shows '=aap'.

	A	B	C	D	E	F	G
6		166.667					
7							
8	aap	100					
9	noot	200					
10							
11							
12							
13							
14							
15							
16							
17							
18							

# MAAK DE CEL GEEL



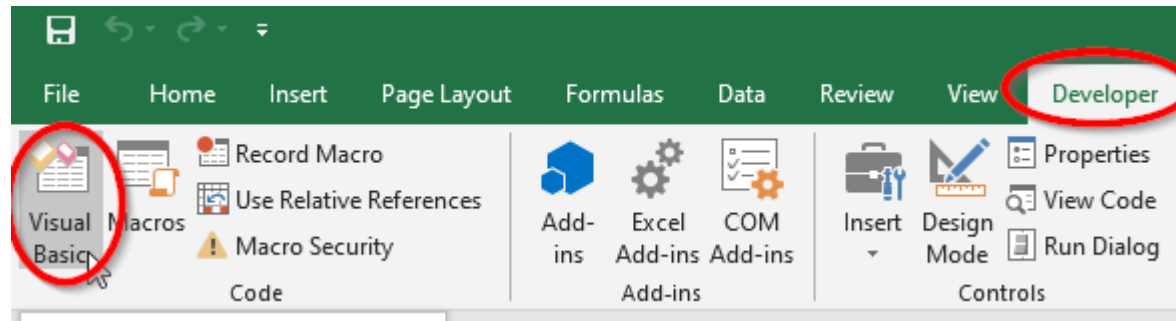
# EN STOP DE MACRO



# DOEL VAN DE MACRO

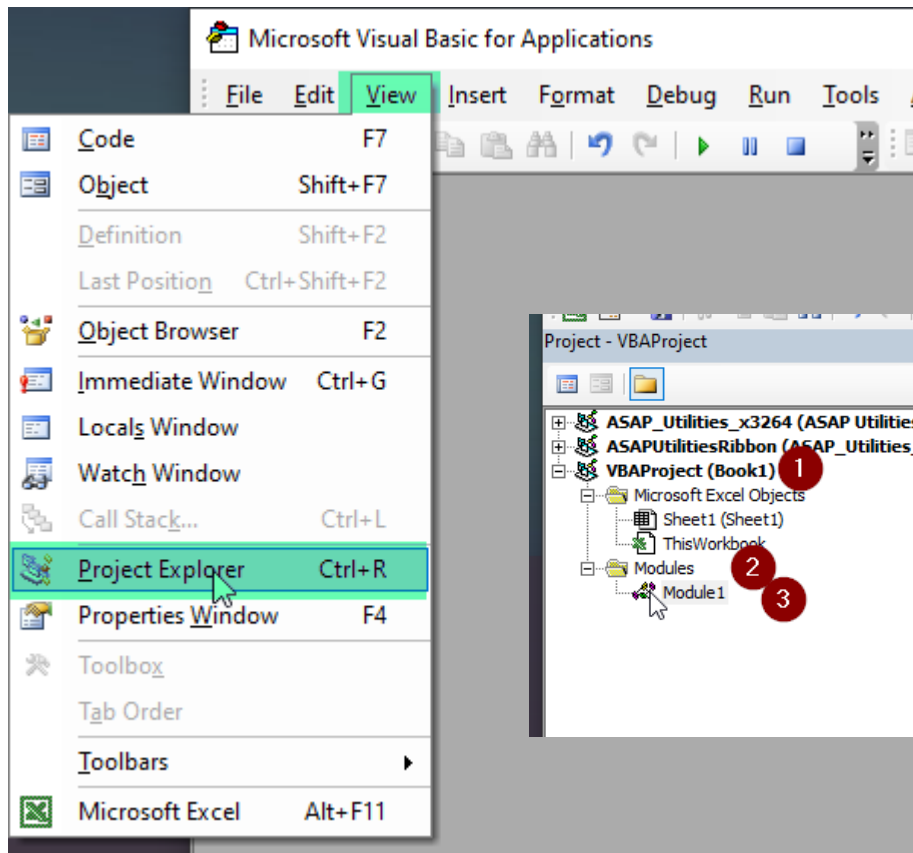
- Repeterende handelingen snel kunnen herhalen.
- Doe gewoon wat je wilt herhalen.
- En als je een letter g aan je macro hebt toegekend, klik op Ctrl g en de macro speelt af.

# WAAR VIND IK DIE CODE DAN?





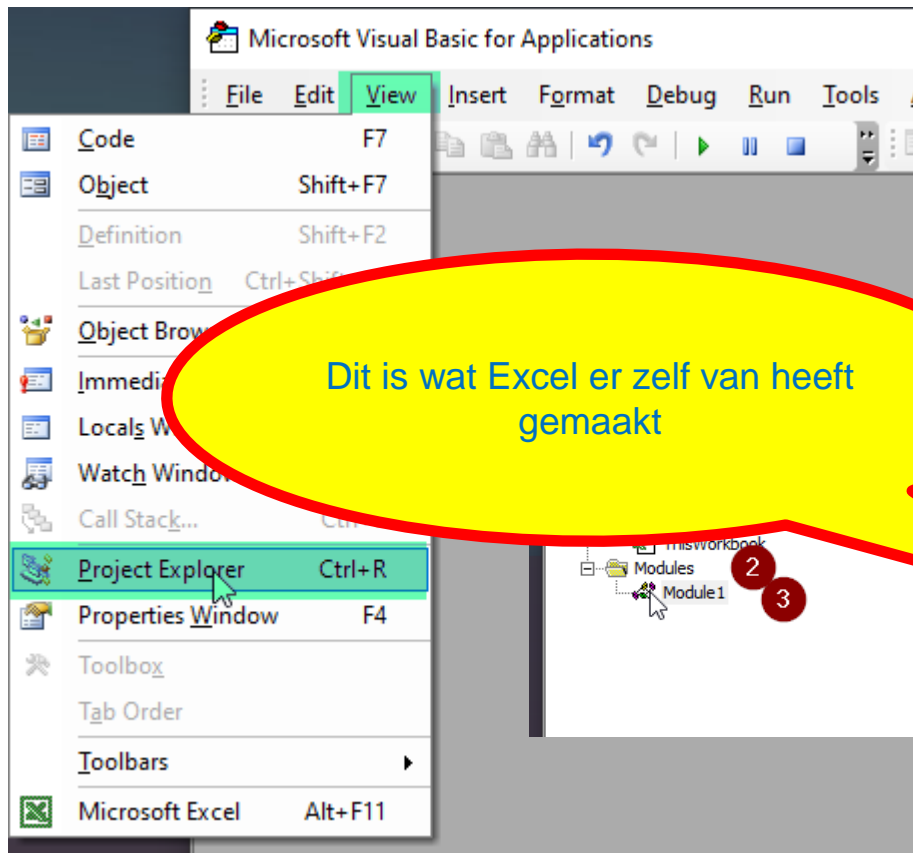
# WAAR VIND IK DIE CODE DAN?



1. De naam van je workbook
2. De modules map
3. De module zelf

```
(General) maakGeel  
  
Sub maakGeel()  
    ' maakGeel Macro  
    ' Keyboard Shortcut: Ctrl+g  
    '   
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 65535  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

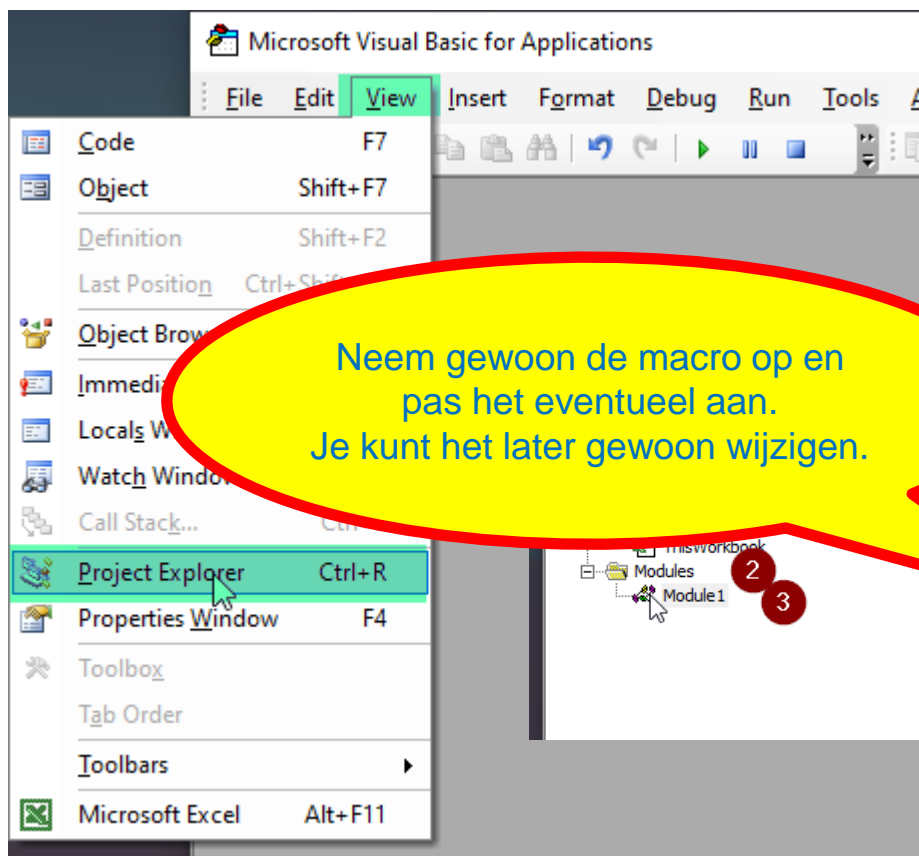
# WAAR VIND IK DIE CODE DAN?



1. De naam van je workbook
2. De modules map
3. De module zelf

```
(General) maakGeel  
Sub maakGeel()  
    maakGeel Macro  
    Keyboard Shortcut: Ctrl+g  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 65535  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

# HOE WEET IK DAN WAT ER MOET KOMEN?

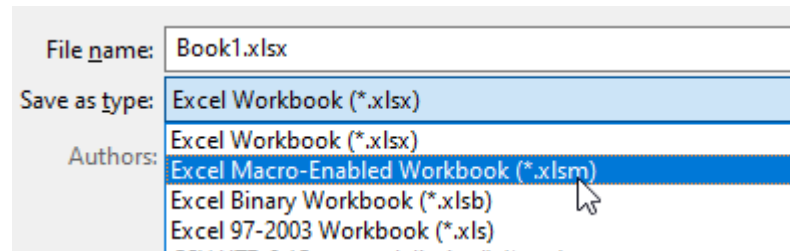


1. De naam van je workbook
2. De modules map
3. De module zelf

```
(General) maakGeel  
Sub maakGeel()  
    maakGeel Macro  
    Keyboard Shortcut: Ctrl+g  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 65535  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

# OPSLAAN

- De macro hoort bij je workbook
- Als je het mailt naar iemand, gaat de macro mee.
- Als je het opslaat, moet dat als xlsx



# DIT IS DE OPGENOMEN MACRO

Microsoft Visual Basic for Applications - Book1 - [Module2 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help

Ln 15, Col 1

Project - VBAProject

VBAProject (Book1)

- Microsoft Excel Objects
  - Sheet1 (Sheet1)
  - ThisWorkbook
- Modules
  - Module1
  - Module2

VBAProject (form keuze geleiders.xlsm)

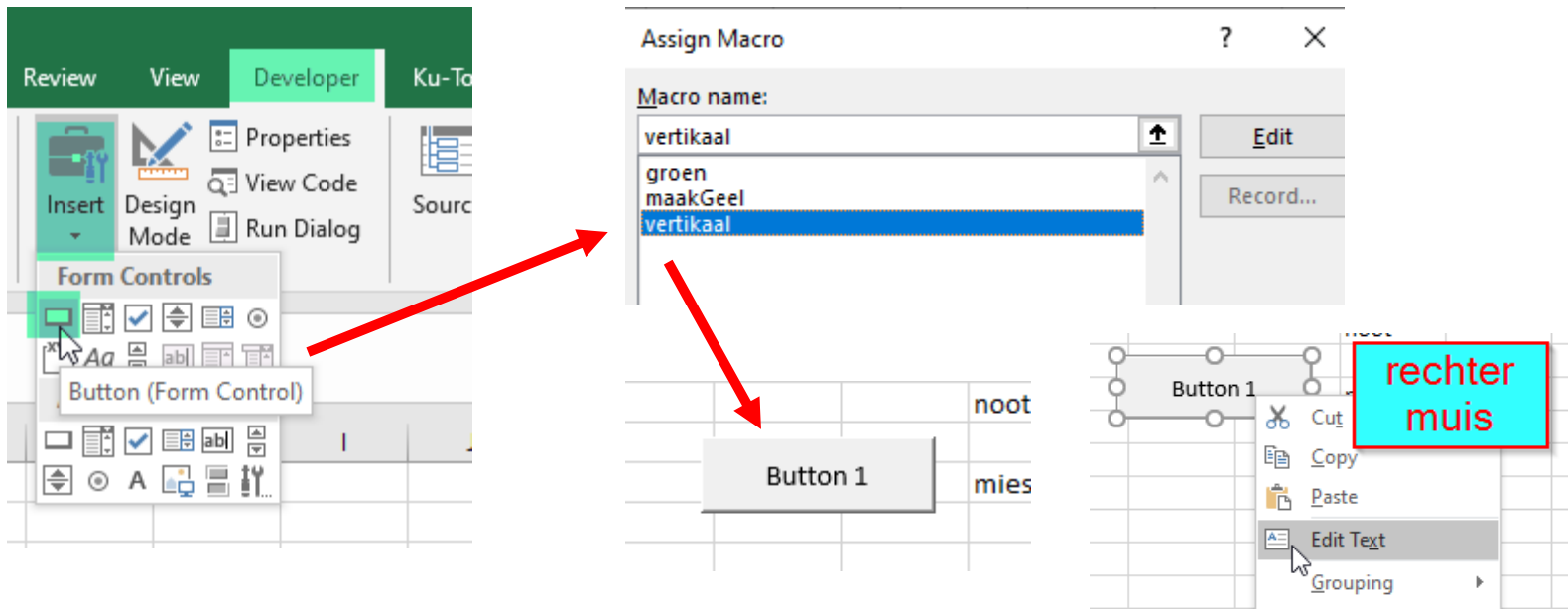
(General)

```
Sub Macro1 ()  
    ' Macro1 Macro  
    '   
    '   
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 65535  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

Modules zijn ervoor om macro's bij elkaar te houden.

# AFSPELEN MACRO

Ik wil in het worksheet een knopje zodat als ik daar op klik, die subroutine wordt uitgevoerd.



# AFSPELEN MACRO

- Klik op de toetsen combinatie bijvoorbeeld Ctrl-g
- Met een button in je scherm (zoals hierboven)
- Ga in Visual Basic in de code staan en klik op
  - F5 in de macro in één keer te doorlopen
  - F8 om het stapsgewijs te doorlopen, telkens met F8 voor de volgende regel.

# AFSPELEN MACRO

Visual Basic for Applications - [Module1 (Code Behind Sheet1)]

File Edit View Insert Format Debug Run

Project - VBAProject

- ASAP\_Utilityes\_x3264 (ASAP Utilityes.xlan)
- ASAPUtilityesRibbon (ASAP\_Utilityes\_ribbc)
- VBAProject (Book1)
  - Microsoft Excel Objects
    - Sheet1 (Sheet1)
    - ThisWorkbook
  - Modules
    - Module1

(General) maakGeel

```
Sub maakGeel  
    ' maakGeel Macro  
    ' Keyboard Shortcut: Ctrl+G  
    With Selection.Interior  
        .Pattern = xlSolid  
        .PatternColorIndex = xlAutomatic  
        .Color = 65535  
        .TintAndShade = 0  
        .PatternTintAndShade = 0  
    End With  
End Sub
```

dat wat is geselecteerd

de binnenkant ervan

maak dat geel

F5 in één keer of F8 stap voor stap



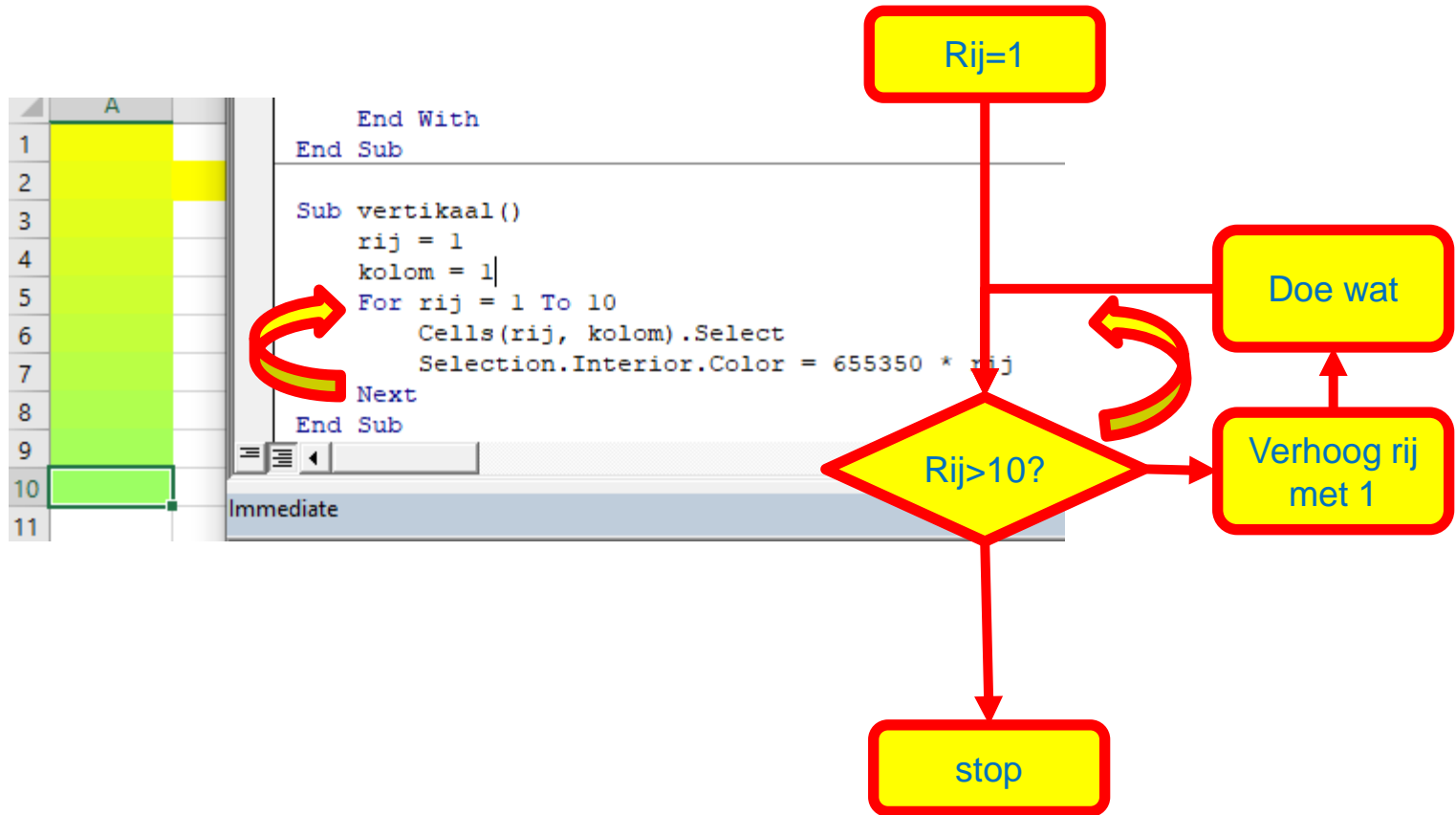
# AANPASSEN MACRO

The image shows a screenshot of Microsoft Excel and the Microsoft Visual Basic for Applications (VBA) editor. The Excel window displays a worksheet with columns A through F and rows 1 through 12. Column A is highlighted in yellow, and column C is highlighted in green. The text 'lijst' is in the name box. The VBA editor window shows the following code:

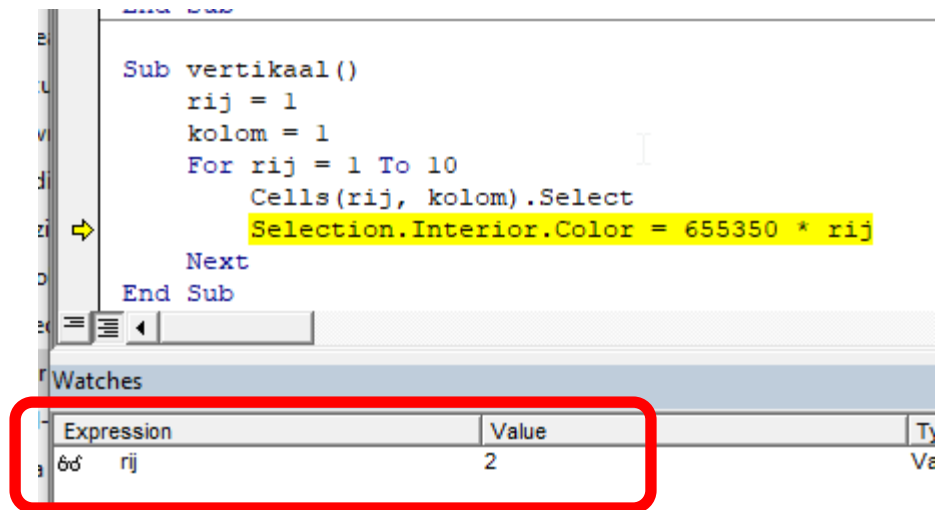
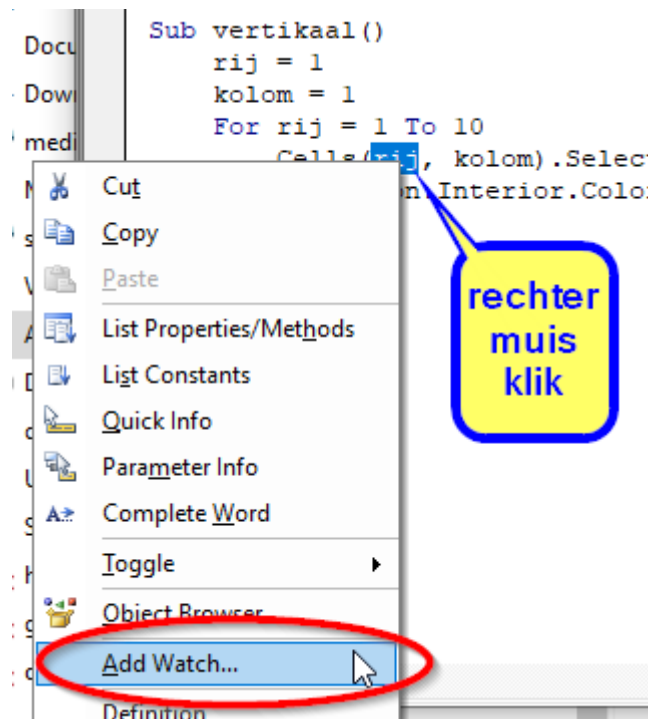
```
Sub test()  
    'dit is commentaar  
    kleur = 655350 'geel  
    Cells(3, 4).Interior.Color = kleur  
    Range("aap").Interior.Color = kleur  
    Range("lijst").Interior.Color = kleur  
End Sub
```

Red arrows point from the VBA code to the corresponding elements in the Excel worksheet: from 'Cells(3, 4)' to cell D3, from 'Range("aap")' to cell B5, and from 'Range("lijst")' to column A. A blue callout box labeled 'range("lijst")' points to the yellow highlighted area in column A.

# AANPASSEN MACRO



# WAT ZIJN DE VARIABELEN



Als je met F8 door deze subroutine stapt, zie je de waarde van rij veranderen.

# FUNCTIES IN VBA

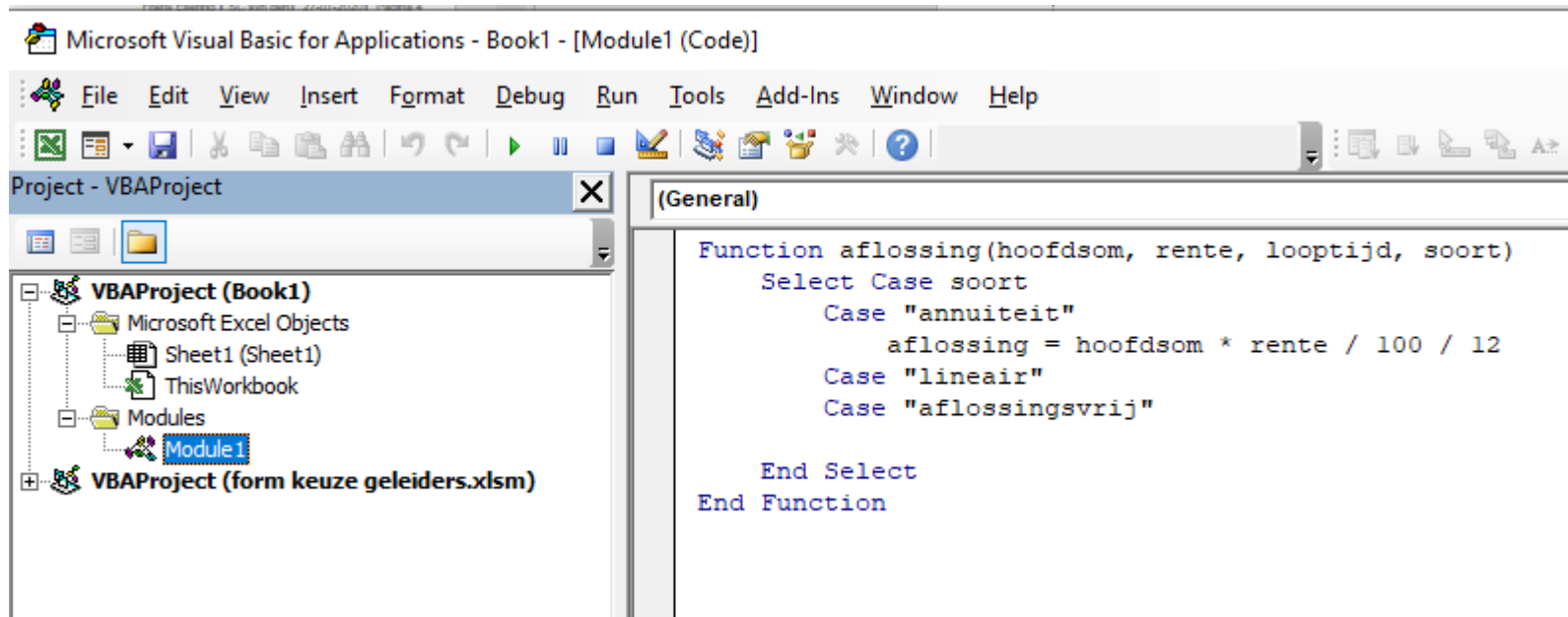
# FUNCTIES IN VBA

- Als je een functie hebt met meerdere “IF” statement, wordt het al gauw onleesbaar.

	A	B	C	D	E	F	G
1	hoofdsom	100000	euro				
2	rente	2	%				
3	looptijd	30	jaar				
4	soort hypotheek	annuiteit					
5	te betalen rente per maand	166.667					
6							

Formula bar: `=IF(B4="annuiteit";B1*B2/100/12;IF(B4="lineair";B1*B2/100*B3/B1/12;IF(B4="aflossingsvrij";B1*B2/100*B3*(B1-(B1*B2/100+B3)*(1-B2)/100))))`

# DE FUNCTIE IN VBA



The screenshot displays the Microsoft Visual Basic for Applications editor interface. The title bar reads "Microsoft Visual Basic for Applications - Book1 - [Module1 (Code)]". The menu bar includes File, Edit, View, Insert, Format, Debug, Run, Tools, Add-Ins, Window, and Help. The Project Explorer on the left shows a project named "VBAProject" containing "VBAProject (Book1)" with sub-items "Microsoft Excel Objects" (Sheet1, ThisWorkbook) and "Modules" (Module1). The Properties Window on the right shows the "(General)" tab. The main code area contains the following VBA function:

```
Function aflossing(hoofdsom, rente, looptijd, soort)
    Select Case soort
        Case "annuiteit"
            aflossing = hoofdsom * rente / 100 / 12
        Case "lineair"
        Case "aflossingsvrij"

    End Select
End Function
```

# VOEG HET TOE MET INSERT FUNCTION

The screenshot shows the Microsoft Excel interface. The 'Formulas' ribbon is highlighted with a red circle. The 'fx' icon in the 'Function Library' group is also circled in red. The 'Insert Function' dialog box is open, with the 'User Defined' category selected in the 'Or select a category by' dropdown and the 'aflossing' function selected in the 'Select a function' list. The spreadsheet below shows a table with columns A, B, and C, and rows 1 to 12. Row 1: A1=hoofdsom, B1=100000, C1=euro. Row 2: A2=rente, B2=2, C2=%. Row 3: A3=looptijd, B3=30, C3=jaar. Row 4: A4=soort hypotheek, B4=annuïteit. Row 5: A5=te betalen rente per maand, B5=166.667. Row 6: A6 is empty, B6 contains an equals sign (=).

A	B	C	D
hoofdsom	100000	euro	
rente	2	%	
looptijd	30	jaar	
soort hypotheek	annuïteit		
te betalen rente per maand	166.667		
	=PMT(;	B3;B4)	

Function Arguments

aflossing

**Hoofdsom** B1 ↑ = 100000

**Rente** B2 ↑ = 2

**Looptijd** B3 ↑ = 30

**Soort** B4 ↑ = "annuïteit"

= 166.6666667

No help available.

**Soort**

Formula result = 166.6666667

[Help on this function](#) OK

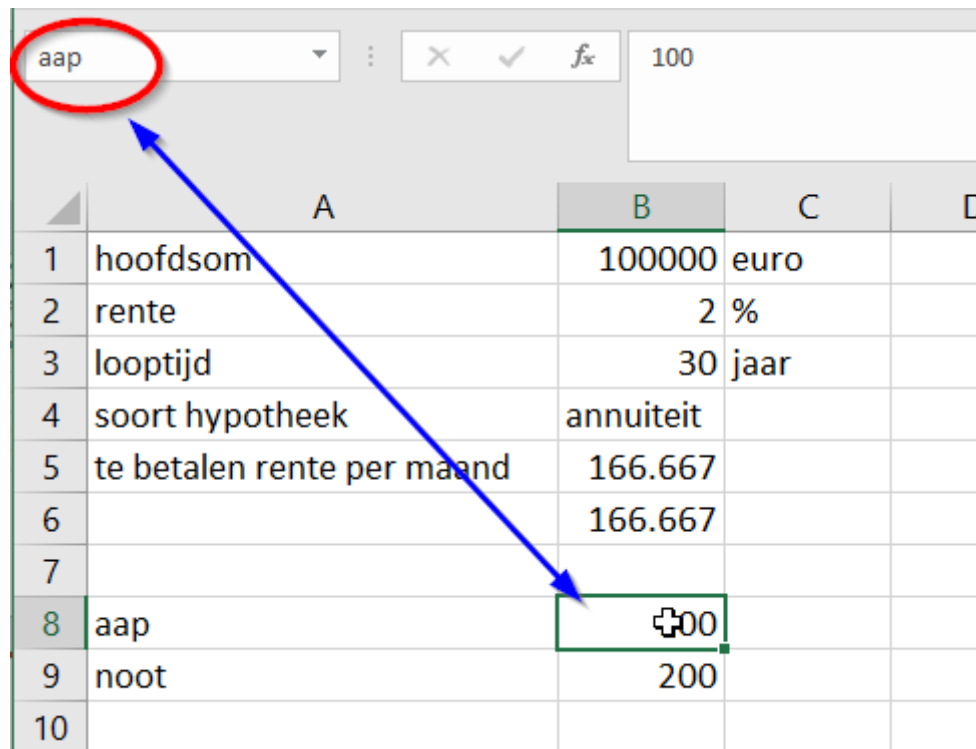


# CELNAMEN

The screenshot shows the Microsoft Excel interface. The 'Formulas' ribbon is active, and the 'Create from Selection' button is circled in red. A dialog box titled 'Create Names from Selection' is open, with the 'Left column' option checked and circled in red. The spreadsheet below shows a table with financial data, with rows 8 and 9 highlighted in yellow.

	A	B	C	D	E	F
1	hoofdsom	100000	euro			
2	rente	2	%			
3	looptijd	30	jaar			
4	soort hypotheek	annuïteit				
5	te betalen rente per maand	166.667				
6		166.667				
7						
8	aap	100				
9	noot	200				
10						

# DIE CEL HEEFT DE NAAM AAP



	A	B	C	D
1	hoofdsom	100000	euro	
2	rente	2	%	
3	looptijd	30	jaar	
4	soort hypotheek	annuïteit		
5	te betalen rente per maand	166.667		
6		166.667		
7				
8	aap	+00		
9	noot	200		
10				

The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D	E
6		166.667			
7					
8	aap	100		100	
9	noot	200			
10					
11					

The formula bar at the top shows the formula `=aap` and a blue arrow points from the formula bar to cell D8, which contains the value 100. Cell B8 is highlighted in yellow and also contains the value 100. Cell D8 is circled in red.

- Je kunt nu overal die naam “=aap” gebruiken

---

```
Function aflossing(hoofdsom, rente, looptijd, soort)
  Select Case soort
    Case "annuïteit"
      aflossing = hoofdsom * rente / 100 / 12
    Case "lineair"
      aflossing = ActiveSheet.Range("aap")
    Case "aflossingsvrij"

  End Select
End Function
|
```



# INSPECTIE OBJECTEN

Alsof je vanuit de maan een briefje stuurt naar

- Aarde
- Europa
- Nederland
- Apeldoorn
- Neptunus
- 56
- Kamer
- Muuroost
- geel

```
Sub maakGeel()  
    With Selection.Interior  
        .Color = 65535  
    End With  
    ActiveWorkbook.ActiveSheet.Cells(2, 3).Interior.Color = 3  
End Sub  
  
Sub groen()  
    With Selection.Interior  
        .Color = 5287936  
    End With  
End Sub
```

Expression	Value	Type
ActiveWorkbook		Object/ThisWorkbook
AcceptLabelsInFormulas	False	Boolean
AccuracyVersion	0	Long
ActiveChart	Nothing	Chart
ActiveSheet		Object/Sheet1
Application		Application/Application
AutoFilter	Nothing	AutoFilter
Cells	3	Range/Range
AddIndent	False	Variant/Boolean
AllowEdit	True	Boolean
Application		Application/Application

# BESTANDSBEHEER

Sub test

```
Set fs = CreateObject("Scripting.FileSystemObject")
```

```
Set a = fs.CreateTextFile("c:\testfile.txt", True)
```

```
a.WriteLine("This is a test.")
```

```
a.Close
```

End sub

<https://docs.microsoft.com/en-us/office/vba/language/reference/user-interface-help/filesystemobject-object>

Method	Description
<a href="#">BuildPath</a>	Appends a name to an existing path.
<a href="#">CopyFile</a>	Copies one or more files from one location to another.
<a href="#">CopyFolder</a>	Copies one or more folders from one location to another.
<a href="#">CreateFolder</a>	Creates a new folder.
<a href="#">CreateTextFile</a>	Creates a text file and returns a TextStream object that can be used to read from, or write to the file.
<a href="#">DeleteFile</a>	Deletes one or more specified files.
<a href="#">DeleteFolder</a>	Deletes one or more specified folders.
<a href="#">DriveExists</a>	Checks if a specified drive exists.
<a href="#">FileExists</a>	Checks if a specified file exists.
<a href="#">FolderExists</a>	Checks if a specified folder exists.
<a href="#">GetAbsolutePathName</a>	Returns the complete path from the root of the drive for the specified path.
<a href="#">GetBaseName</a>	Returns the base name of a specified file or folder.