

Macros (Subroutines):

- In Lab session 1 you have already seen how to write a subroutine (also called a Macro) using the VBA editor.
- A subroutine is quite different from a UDF. We can not use it in an Excel WS as a function but we have to “run” it.
- The problem with macros is that one generally needs to know quite a bit about the programming language (VBA) to be able to write them.
- There is however an alternative to this. We can create subroutines by using the Macro recorder.
- The Macro recorder is just an Excel application that automatically translates any action that you carry out on the Excel WS into a subroutine (it writes down the code for you!).
- In this way you do not need to know any VBA commands!

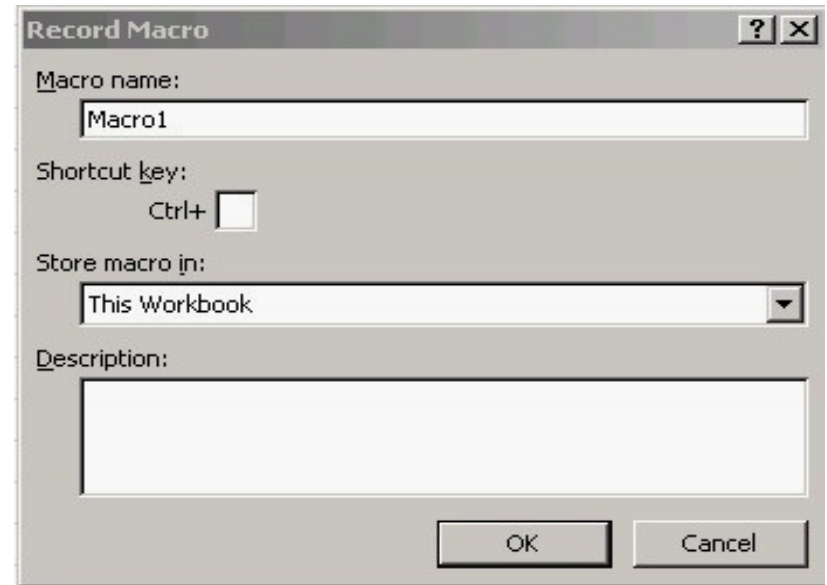
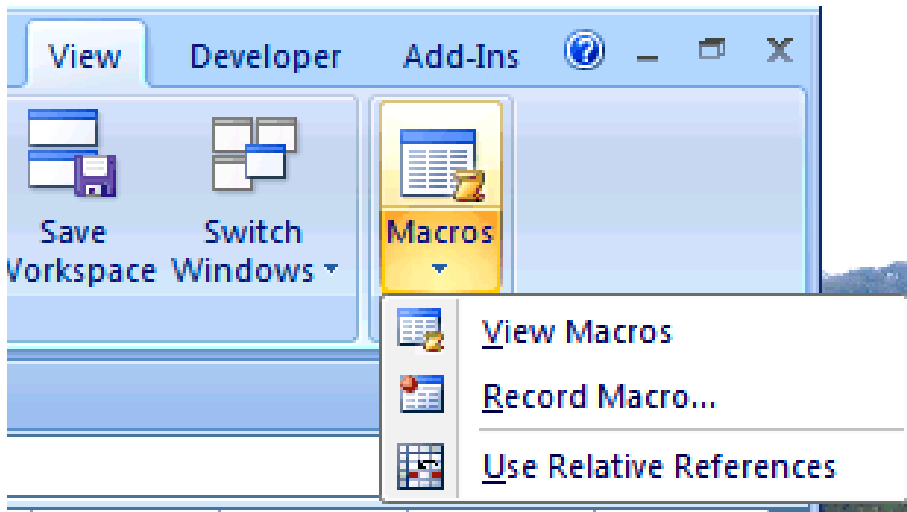
- The advantage of this is that if you want to repeat a certain action on the WS many times you can just run your subroutine.

1) Recording a Macro:

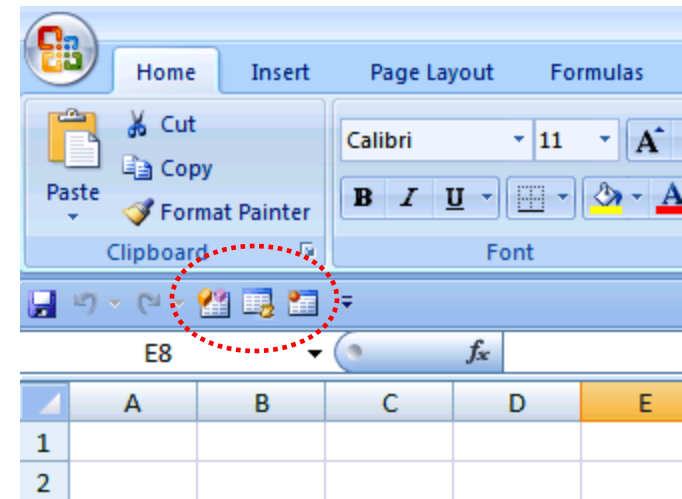
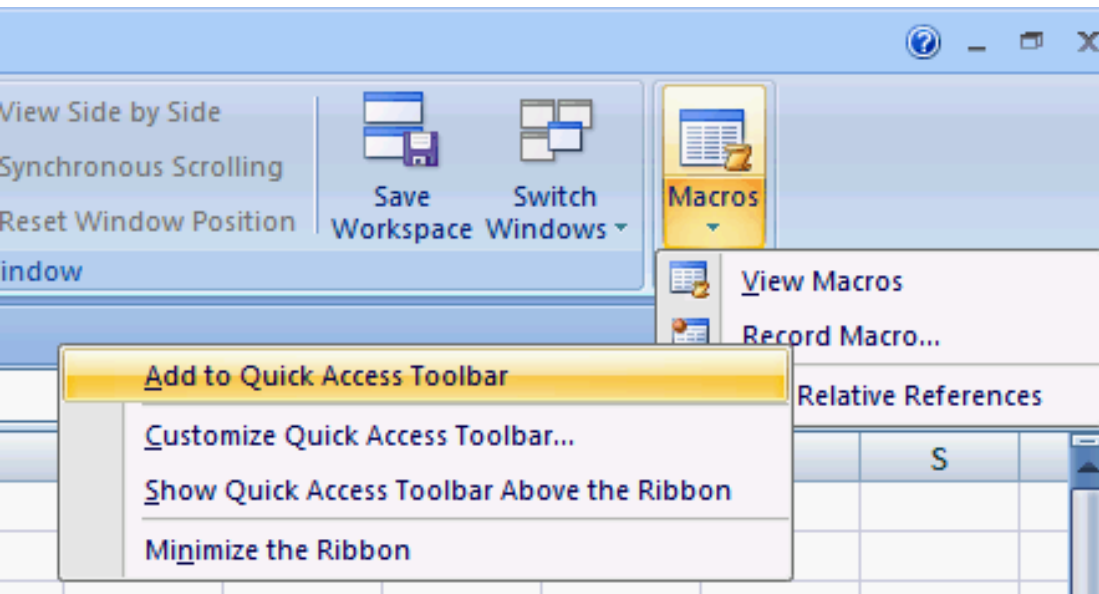
i) open a worksheet

ii) Select the “View” Tab → Macros → Record Macro ↵

⇒ the record Macro dialog box opens up



- If you are going to be using your Macro a lot from now on, it will be useful for you to have a shortcut for this.
- You can add the “**view macros**” and “**record macros**” options to the “**quick access toolbar**” so that in the future you just need to click on the corresponding icons.
- You just have to right click on “**view macros**” or “**record macros**” and choose the option “**Add to Quick Access Toolbar**”.



2) **Example:** let us record a macro that adds up all entries on cells A1 to A30

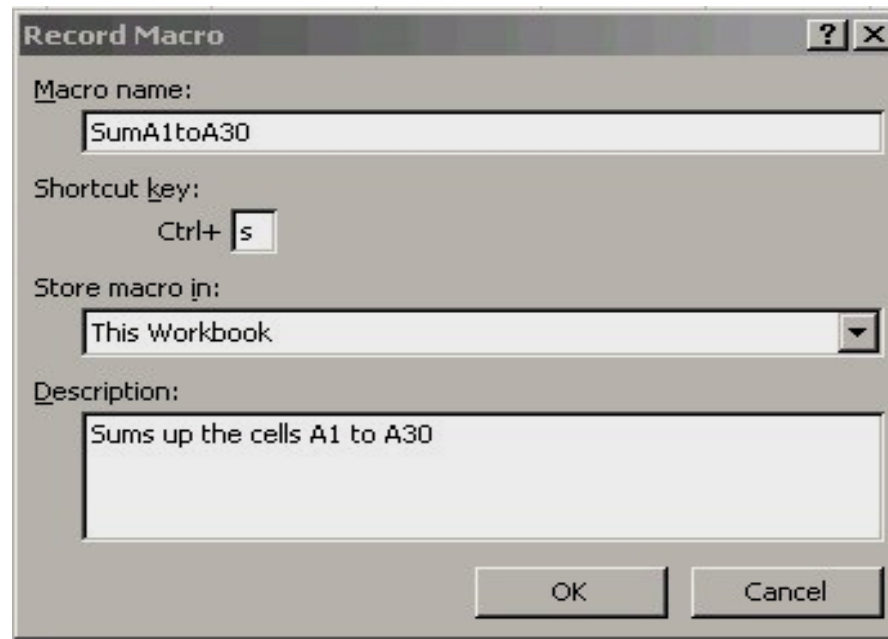
a) enter Macro Name, e.g. “SumA1toA30“

b) enter a letter for the shortcut key, e.g. “s“

c) store the macro somewhere, e.g. “This workbook“

d) fill in the description box, e.g. “sum up the cells A1:A30“

e) Ok ↵ , the recording is on. Now all actions you carry out on the worksheet will be recorded and a VBA code will be produced.



3) While the recording is on we must now carry out the action the sum of the cells A1:A30

- select a cell in your worksheet different from column A
- write: **The sum of the cells A1:A30 is:**
- select the adjacent cell and write: **=Sum(A1:A30)**
- the effect of this is that in the cell in which you wrote **=Sum(A1:A30)** this sum will be displayed
- you can now change the content of A1:A30 and the sum will be updated automatically

4) select View → Macros → Stop Recording

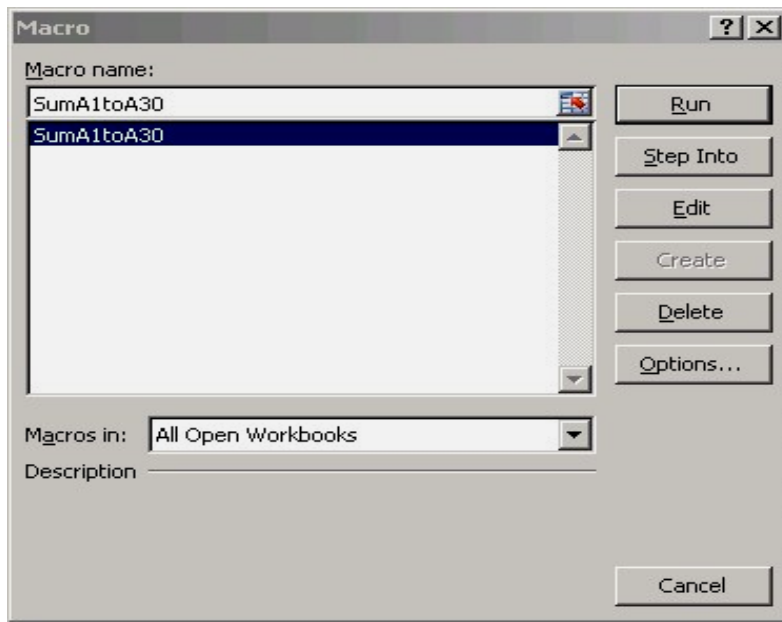
-alternatively in the quick access toolbar the “record macro” icon would have changed into a square



-You just have to click on that square to stop recording.

5) Viewing the code:

- The recording has produced a VBA code, which alternatively we could have programmed by hand:
- Let s see what we have just produced:
- Select View → Macros → View macros
⇒ a window called Macros opens up
- the window “Macro name“ shows the name of all Macros that you have in this particular workbook



- Select Edit ↵
⇒ the code we have just produced will show up

Sub SumA1toA30()

'

' SumA1toA30 Macro

' Sums up the cells A1 to A30

'

' Keyboard Shortcut: Ctrl+s

'

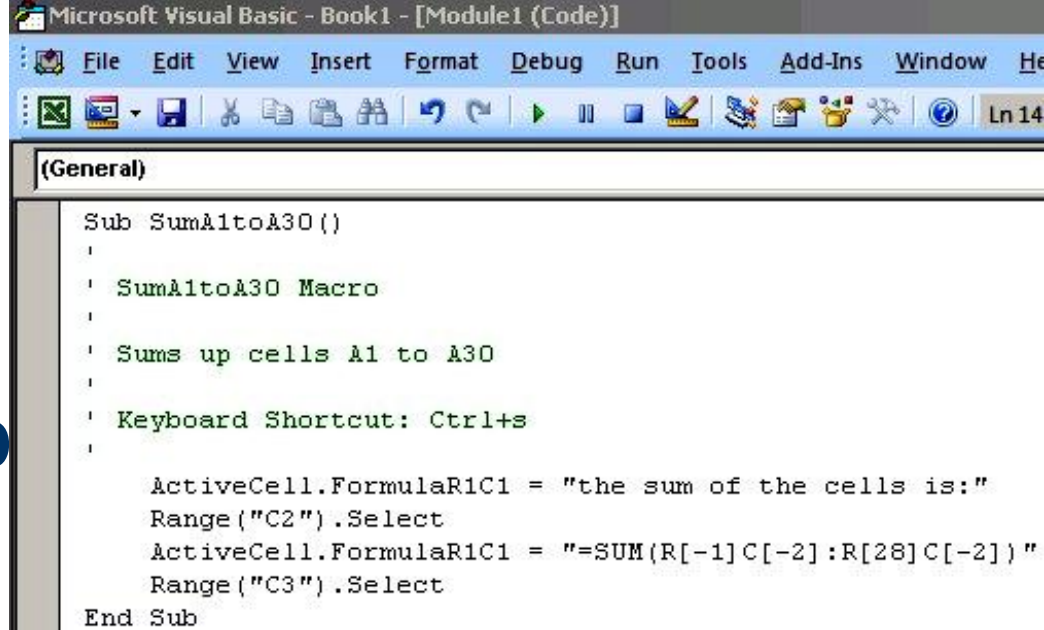
ActiveCell.FormulaR1C1 = "The sum of the cells A1:A30 is:"

Range("C2").Select

ActiveCell.FormulaR1C1 = "=SUM(R[-1]C[-2]:R[28]C[-2])"

Range("C3").Select

End Sub



The screenshot shows the Microsoft Visual Basic editor window for a macro named 'SumA1toA30'. The code is as follows:

```
Sub SumA1toA30()  
'  
' SumA1toA30 Macro  
' Sums up cells A1 to A30  
'  
' Keyboard Shortcut: Ctrl+s  
'  
    ActiveCell.FormulaR1C1 = "the sum of the cells is:"  
    Range("C2").Select  
    ActiveCell.FormulaR1C1 = "=SUM(R[-1]C[-2]:R[28]C[-2])"  
    Range("C3").Select  
End Sub
```

6) Activating the Macro:

a) Select View → Macros → View Macros ↵

⇒ a window called Macros opens up and the macro's name appears in the window "Macro name:"

· in case you have more than one, select the one you want

Select Run ↵

⇒ what you have recorded before will be executed now

b) Use the shortcut:

- our example just: Ctl + s

c) If you were editing the code:

Select ▶ ↵

⇒ a window called Macros opens up ⇒ a)

- **Example:** We could use this macro to calculate once more

$$\sum_{a=1}^n a = \frac{n(n+1)}{2}$$

- first you have to fill in: 1→A1 , 2→A2, 3→A3 ... 30→A30
- use the autofill option!
- activate the Macro ⇒
- **The sum of the cells A1:A30 is 465**

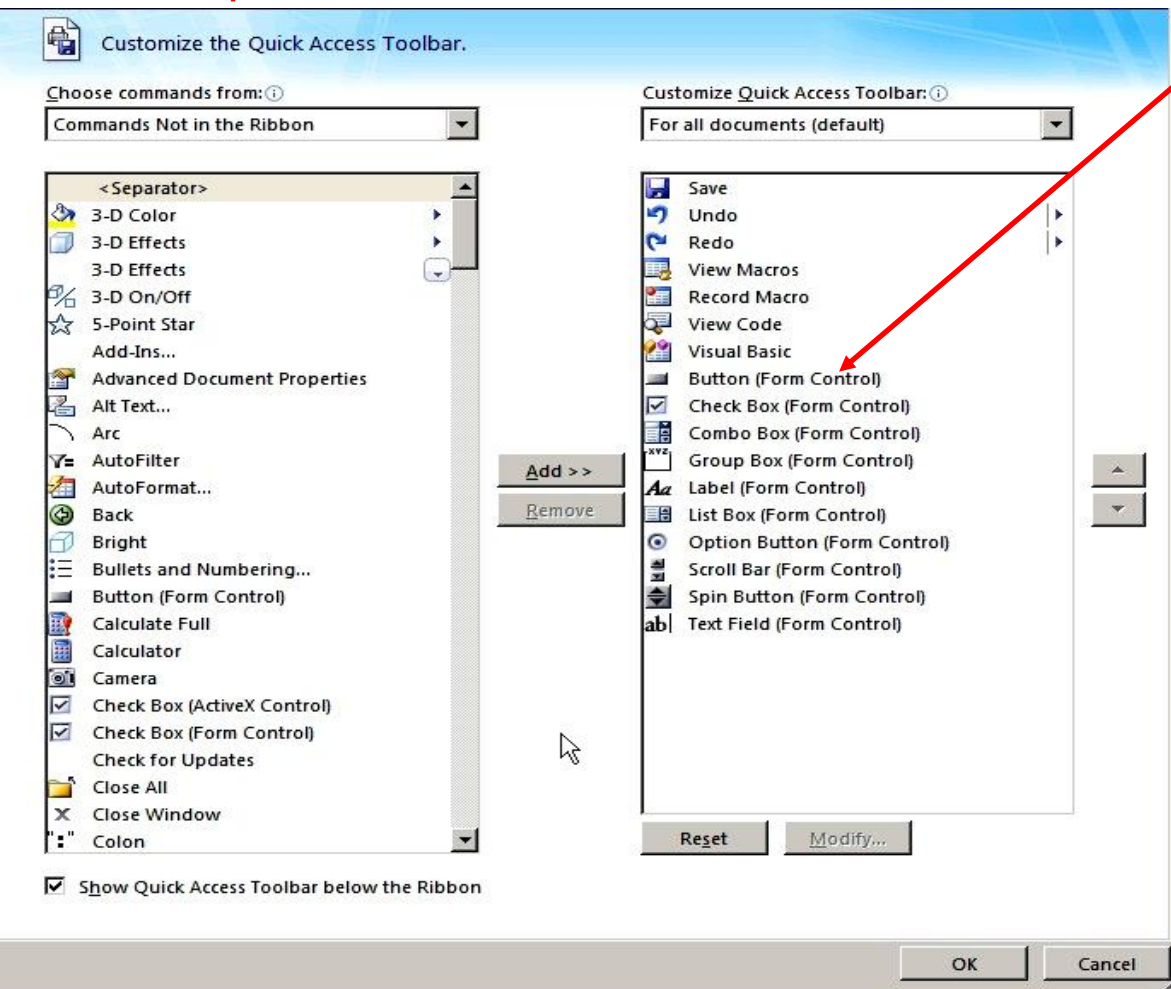
The macro is only useful here if we want to do such a calculation many times so that doing Ctl +s is actually a lot faster than writing =sum(A1:A30)!

There is yet one more way of running a Macro. We can create an specific button associated to the Macro, so that each time we click that button the Macro is run!

7) Customized buttons (and other objects):

a) Make the “Button” option visible on the quick access toolbar

RC on the quick access toolbar → Customize quick access toolbar → Commands not in the ribbon → **Button (Form control)** → Add

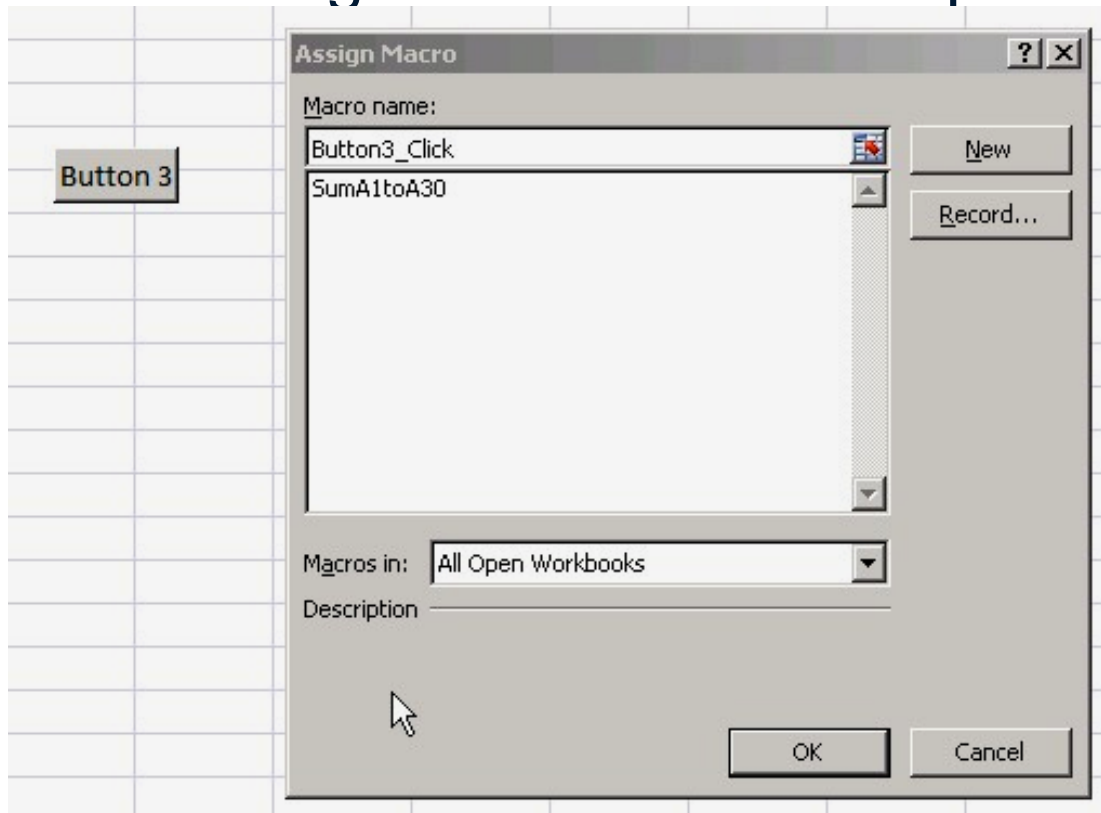


→ OK

You can also make other commands visible in the quick access toolbar by going down the list and selecting them!

8) Assign a Macro to a button:

- Select Button (a grey box)
- select a cell in your worksheet
- ⇒ the “Assign Macro” window opens up



→ select the Macro which you want to attach to this button, e.g. SumA1toA30 → Ok ↵

9) Activating the Macro:

- Selecting now this button will activate the Macro you have attached to it, e.g. SumA1toA30

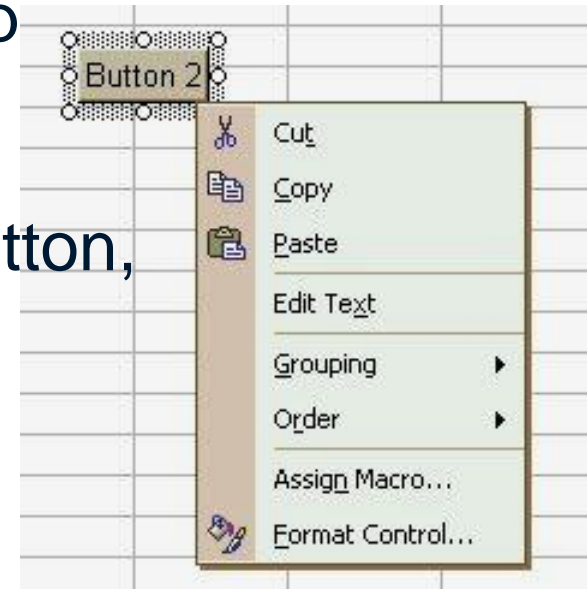
10) Changing the button design:

- attach a better text to the button:

- select the right mouse button (moving first over the button) → a variety of commands opens up

→ select Edit text ↵

→ type a meaningful text onto the button,
e.g. Sum A1 to A30



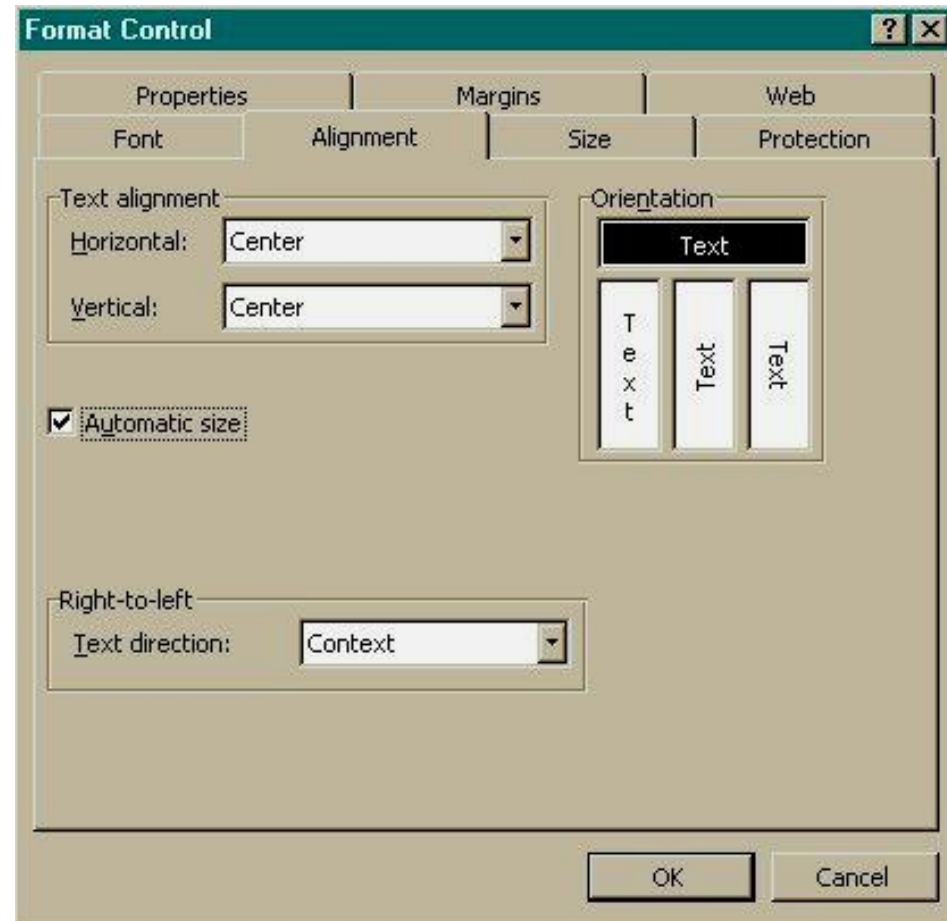
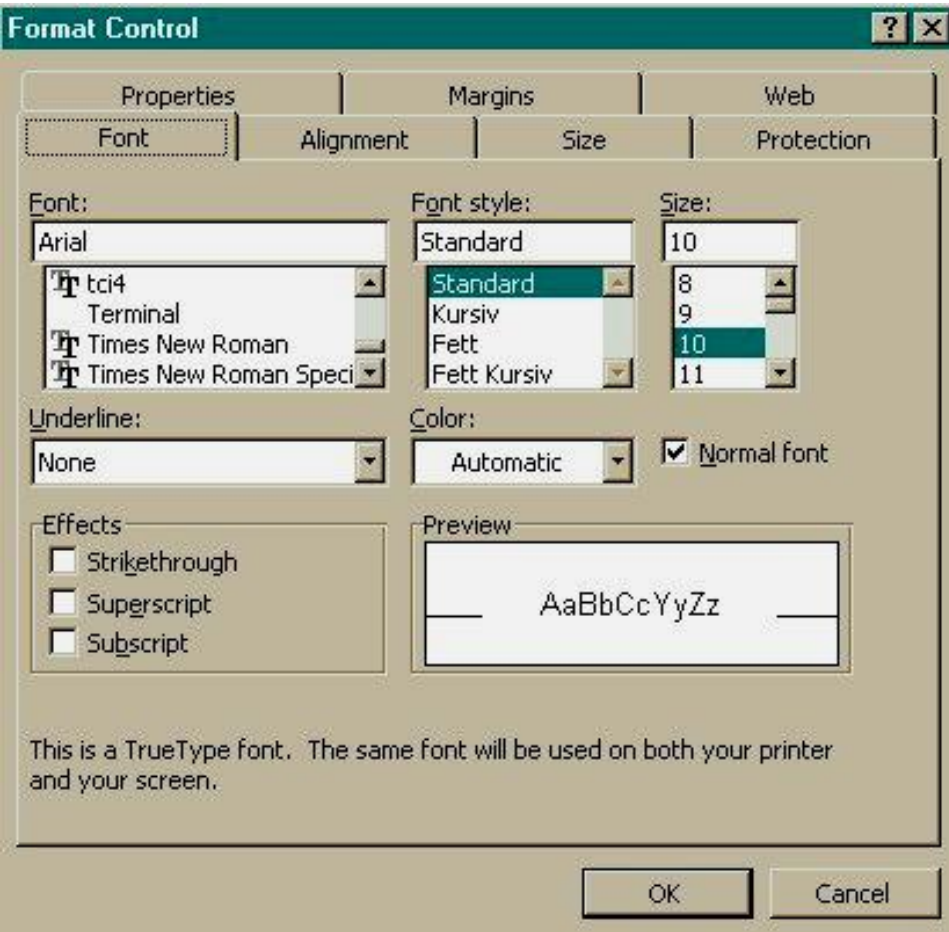
- change the size of the button:

- select the right mouse button (moving first over the button) → select Format Control ↵

- change the size of the button:

- select the right mouse button (moving first over the button) → select Format Control ↵

→ Alignment



- similarly you can change the writing direction, the text fonts, the text and button size, the margins of the button, the colour, the status of the protection, etc.a

Sum of A1 to A30

Sum of A1 to A30

Sum of A1 to A30

- You can also assign Macros to other objects (different from the button) or even to a picture that you have imported previously to the WS.

