

מבוא לפקודות מאקרו VBA באקסל

1. הקדמה – למה צריך את זה? הקלה על ביצוע משימות שחוזרות על עצמן. ביצוע משימוש שלא ניתן לבצע כגון לולאות, יצירת פונקציות מותאמות אישית, יצירת טפסים למשתמש ועוד.
2. סביבת עבודה כניסה לעורך ALT+F11. או קליק ימני על שם הגיליון והצג קוד או דרך התפריטים בפקודות מאקרו הצג קוד.
3. בתוך עורך ה VBA כלים אפשרויות – להגדרת סביבת עבודה של העורך. הוספת הערות בעזרת גרש ' לפני השורה.
4. חלונות צפייה במשתנים (ערך המשתנה בכל שלב בתוכנית) - בתפריט דרך DEBUG ← add watch או קליק ימני על המשתנה ואז add watch.
5. הקלטת מאקרו – מומלץ להוסיף את הסרגל של הקלטת מאקרו מסרגלי כלים לנוחות העבודה. ניתן לבצע הקלטה יחסית או מוחלטת. שמירת המאקרו בגיליון מקומי או בגיליון personal.xls בחירה במותאם אישית, יוצרת בפעם הראשונה חוברת(קובץ אקסל עם מספר גיליונות) מוסתרת בשם זה – לא לשכוח לשמור שינויים בסוף העבודה בחוברת זו. מרגע זה המאקרו יהיה זמין בכל חוברות העבודה. במאקרו יחסי שים לב השקוד מתחיל לפי ActiveCell או Selection ולא טווח מפורש. ללחוץ על הפניה יחסית לפני תחילת הקלטת הפקודה שאנו רוצים שתעבוד גם על טווח אחר בגיליון.
6. ניתן להוסיף מקשי קיצור למאקרו לשיפור העבודה. אפשר לשלב עם מקש SHIFT. לא להשתמש בקיצורי עבודה הקיימים כבר בתוכנה כגון ctrl+v ctrl+c וכד'.
7. שם המאקרו ללא רווחים בלי מילים שמורות כגון save print וכד'. ניתן להשתמש בקו תחתון _ לחבר בין מילים. עדיף לעבוד עם שם באנגלית ושם המאקרו יעזור לנו להיזכר מה הוא עושה. לכן לא להיות מאוד קצרים.
8. מתוך הגיליון ALT + F8 פתיחת רשמית מאקרויים קיימים ומשם ניתן להריצים.
9. בתוך העורך לחיצה על F5 - הרצה של הקוד בצורה רציפה.
10. בתוך העורך F8 - הרצת הקוד שלב אחר שלב. יציאה על ידי כפתור הריבוע (reset) עצור או לחיצה על F5 להרצת המאקרו עד סופו.
11. בתוך העורך F9 קביעת נקודת עצירה בקוד או על ידי לחיצה בצד שמאל. להמשך ריצה F5 או F8 לצעד הבא. לחיצה נוספת על F9 או בצד יבטלו את סימן העצירה.
12. יציאה מלולאה באמצע על ידי ESC או CTRL + BREAK. מתאים למקרים שיצרנו לולאות אין סופיות בטעות.

13. ניתן להוסיף כפתור מותאם אישית וחיבורו למאקרו בסרגלי הכלים – דרך הוספה הסרה של סרגלי כלים ובחירה במותאם אישית, ניתן לעצבו או לבחור איור מרשימת האיורים על ידי לחיצה בקליק ימני על הכפתור שיצרנו. ב 2007-2010 דרך סרגל כלים לגישה מהירה ניתן להוסיף את המאקרו לסרגל ולבחור באיור עבורו.
14. הוספת כפתור במשטח העבודה דרך טפסים, הקצאת מקרו ועריכתו. דרך סרגל VB.

15. שיפור הקוד – על ידי הוספת הערות עם הסימן גרש ומחיקת שורות מיותרות המאטות את קוד והופכות אותו למסורבל. לחייב הגדרת משתנים (בעזרת הפקודה Option Explicit להוסיף בראש המודול), לבטל רענון בכל פקודה, הוספת הפקודה הבאה בראש הקוד: Application.ScreenUpdating = Fales

16. פקודת מאקרו מתחילה ב Sub לאחריו שם המאקרו עם סוגריים (נוספות לבד) ומסתיימת ב End Sub (נוסף אוטומטית כאשר לוחצים ENTER). הקוד שלנו ייכתב כאמצע.

17. תחביר הפקודה – הפקודה רשומה משמאל לימין כאשר הסימן נקודה (.) מפריד בין חלקי הפקודה. תחילה מודיעים לפקודה על מה לעבוד, טווח הפקודה - כלומר האם על כל הגיליון, עמודה, שורה, תא או על טווח מסוים ורק לאחר מכן מה לעשות איתו, הפקודה - לדוגמא לצבוע, למרכז, לסמן, לשנות ערך וכד'.

18. השלמה אוטומטית בעורך VB – שילוב המקשים רווח Ctrl+. חוסך הקלדה וטעויות הקלדה ומראה לנו את אפשרויות הפקודה השונות.

19. הכרזה על משתנים – Dim x As Y - לכתיבת קוד ברור ויעיל יותר מומלץ להכריז מראש.

לדוגמא: Dim Max_Num As integer , Long_Text As String

סוגי משתנים:

BYTE 0-255

BOOLEAN True of False

Integer -32,768-32,767

Long -2,147,483,648-2,147,483,647

String מאחסן מחרוזת טקסט

Range מאחסן טווח תאים בתוכו

Date מאחסן בתוכו תאריכים

Variant משתנה על – מאחסן בתוכו את כל סוגי המשתנים – תופס מקום רב בזיכרון, כאשר לא מכריזים על משתנה או שמכריזים ולא מגדירים מאיזה סוג, אז הוא מוגדר אוטומטית מסוג Variant.

השמה של ערך למשתנה על ידי הסימן =

השמה של אובייקט (עמודה, שורה, תא, טווח, גיליון וכד') על ידי הפקודה SET

20. ניתן לחייב את המשתמש להכריז על משתנים – בראש המודול לרשום Option Explicit

21. קריאה למאקרו מתוך מאקרו אחר בעזרת ציון שם המאקרו בתוך מאקרו אחר (ללא הסוגריים)

.22

הסבר עם פקודות בסיסיות באקסל

1. Range

מאפשר להתייחס לטווח או לתא ספציפיים.
לאחר מכן תגיע הפקודה צריכה להתבצע על הטווח זה.
לדוגמאות שונות לשימוש בפקודה: (תרגיל 1)

```
Sub range_c5_select()  
Range("c5").Select  
End Sub
```

```
Sub range_a1_e10_select()  
Range("a1:e10").Select  
End Sub
```

```
Sub range_d8_e12_selesc()  
Range("d8", "e12").Select  
End Sub
```

```
Sub active_to_f10()  
Range(ActiveCell, "f10").Select  
End Sub
```

```
Sub name_selest_clients()  
Range("clients").Select  
End Sub
```

```
Sub select_by_names_clients_and_title()  
Range("title", "clients").Select  
End Sub
```

2. Selection

מתייחס לבחירה אותה ביצעתי קודם . לדוגמה: העתק את הבחירה שלי.

```
Sub selection_copy()  
Selection.Copy  
End Sub
```

3. ActiveCell

התייחס לתא הפעיל כרגע. לדוגמה להכנסת ערך לתא הפעיל:

```
Sub active_Cell_Value()  
ActiveCell.Value = 352  
End Sub
```

4. Rows – (תרגיל 2) – בחירת מספר שורות

```
Sub Row_select_3_TO_7()  
Rows("3:7").Select  
End Sub
```

5. Columns – בחירת מספר עמודות

```
Sub Columns_Select_b_TO_b()  
Columns("b:e").Select  
End Sub
```

6. Cells –

בחירת תאים לפי מספר שורה ועמודה. ניתן להכניס במקום המספרים משתנים כגון לולאות. המספר 4 מייצג את מספר השורה ו 2 את מספר העמודה. פקודה זו תבחר את תא b4.

```
Sub Cells_by_Numbers_row_column()  
Cells(4, 2).Select  
End Sub
```

7. Offset

מאפשר פנייה יחסית לתא בו אנו נמצאים. בדוגמה זו נגיע לתא f7.

```
Sub Offset_row_column()  
Range("b4").Offset(3, 4).Select  
End Sub
```

8. קיצורי מקלדת (תרגיל 3)

דרכים לבחור תאים ב VBA התואמים לעבודה עם קיצורי המקלדת באקסל:

```
Sub Ctrl_Down_arrow()  
Selection.End(xlDown).Select  
End Sub
```

```
Sub Ctrl_Up_arrow()  
Selection.End(xlUp).Select  
End Sub
```

```
Sub Ctrl_Right_arrow()  
Selection.End(xlToRight).Select  
End Sub
```

```
Sub Ctrl_Left_arrow()  
Selection.End(xlToLeft).Select  
End Sub
```

```
Sub Shift_Ctrl_Down_arrow()  
Range(Selection, Selection.End(xlDown)).Select  
End Sub
```

```
Sub Shift_Ctrl_Up_arrow()  
Range(Selection, Selection.End(xlUp)).Select  
End Sub
```

```
Sub Ctrl_Star()  
Selection.CurrentRegion.Select  
End Sub
```

```
Sub Select_all_ctrl_a()  
Cells.Select  
End Sub
```

```
Sub Ctrl_home()  
Range("a1").Select  
End Sub
```

```
Sub CTRL_END()  
ActiveCell.SpecialCells(xlCellTypeLastCell).Select  
End Sub
```

9. בחירת גיליונות (תרגיל 4)

```
Sub Select_sheet_By_Name()  
Sheets("By_name").Select  
End Sub
```

```
Sub Select_Sheet_3_By_Number()  
Sheets(3).Select  
End Sub
```

```
Sub Select_Last_Sheet()  
Sheets(Sheets.Count).Select  
End Sub
```

10. הוספת גיליון בסוף

```
Sub Add_New_sheet()  
Sheets.Add After:=Sheets(Sheets.Count)  
End Sub
```

11. שינוי שם הגיליון שהוספנו

```
Sub Rename_Sheet()  
Sheets.Add After:=Sheets(Sheets.Count)  
ActiveSheet.Name = Sheets.Count + 1  
End Sub
```

12. מתן שם לטווח תאים (תרגיל 5)

```
Sub Name_To_renge()  
Range("c8:g16").Name = "clients2"  
End Sub
```


13. העתקה של טווח תאים והדבקה במקום אחר, לסיים ביטול ה"נמלים הצועדות".

```
Sub Copy_By_Name_And_Paste()  
Range("clients2").Copy  
Range("c20").Select  
ActiveSheet.Paste
```

```
Application.CutCopyMode = False  
End Sub
```

14. With

הפקודה with - כאשר מבצעים מספר פקודות על אותו טווח ניתן לכתוב זאת בעזרת הפקודה with. עבודה עם פקודה זו הופך אותה לברורה יותר וקצרה יותר. כל הפקודות שרשומות תחת with ממתייחסות אליה. ניתן ליעל את הקוד על ידי מחיקת שורות מיותרות. (ראה תרגיל 6)

```
With Range  
.command  
.command  
End with
```

With Selection.Font

```
.Name = "Aharoni"  
.Size = 11  
.Strikethrough = False  
.Superscript = False  
.Subscript = False  
.OutlineFont = False  
.Shadow = False  
.Underline = xlUnderlineStyleNone
```

.Color = -10477568

.TintAndShade = 0

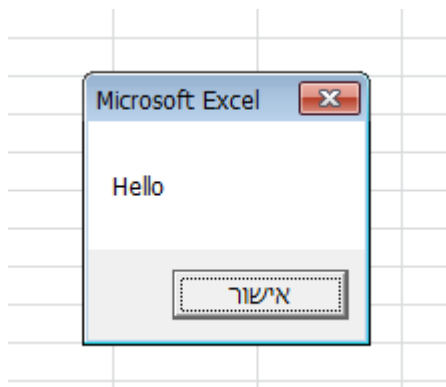
.ThemeFont = xlThemeFontNone

End With

תרגול עבודה עם מחרוזות ראה תרגיל 7.

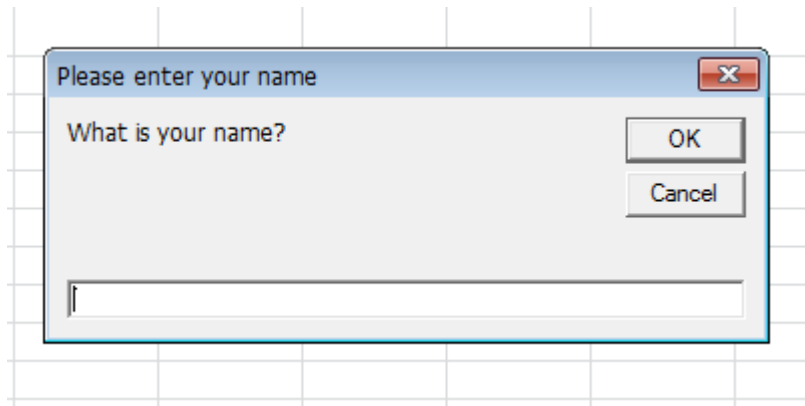
תרגול עבודה עם פונקציות תאריך ושעה ראה תרגיל 8.

15. MsgBox (תרגיל 9) למשתמש בתוכנה



```
Sub msgbox_hello()  
MsgBox "hello"  
End Sub
```

16. תיבת דו שיח לקליטת נתונים מהמשתמש InputBox



```
Sub hello_box2()  
your_name = InputBox("What is your name?", "Please enter your  
name")  
End Sub
```

17. דוגמה לשילוב 2 תיבות גם יחד.

```
Sub hello_box3()  
Your_name = InputBox("What is your name?", "Please enter your  
name")  
MsgBox "Hello " & Your_name & vbNewLine & "Have a nice day",  
vbMsgBoxRight  
ans = MsgBox("do you like icecream?", vbYesNo + vbCritical)  
End Sub
```

18. If Then, Else ,Elseif , IFF

דרכים שונות לכתוב משפטי תנאי (תרגיל 10).

```
Sub Total_Income_Check()  
Total = If(ActiveCell.Value > 120000, "go home", "back to work")  
End Sub
```

```
Sub if_then_one_line()  
If a > 5 Then MsgBox a  
End Sub
```

```
Sub if_then_only()
```

```
  If a > 5 Then
```

```
    MsgBox a
```

```
    MsgBox a + 10
```

```
    MsgBox a + 100
```

```
  End If
```

```
End Sub
```

```
Sub if_then_else()
```

```
  If a > 5 Then
```

```
    MsgBox a
```

```
    MsgBox a + 10
```

```
    MsgBox a + 100
```

```
  Else
```

```
    MsgBox "have a nice day"
```

```
  End If
```

```
End Sub
```

```
Sub if_then_elseif()  
If a > 5 Then  
MsgBox a  
MsgBox a + 10  
MsgBox a + 100  
Elseif a < 200 Then  
MsgBox "go back to work"  
Else  
MsgBox "have a nice day"  
End If  
  
End Sub
```

19. Case מאפשר להציב בצורה קלה מספר רב של תנאים (תרגיל מסכם)

```
Sub SelectDivident()  
Dim Dividend As Currency  
Dim Total As Currency  
Range("f22").Activate  
Total = ActiveCell.Value  
Select Case Total  
Case Is >= 150000  
    Dividend = Total * 0.03  
Case Is >= 100000  
    Dividend = Total * 0.02  
Case Is >= 50000
```

```
Dividend = Total * 0.01
```

```
Case Else
```

```
Dividend = Total * 0.005
```

```
End Select
```

```
MsgBox " Your dividend is : " & Dividend
```

```
End Sub
```

20. For Next (תרגיל 11) עבודה על לולאות

```
Sub FOR_NEXT_LOOP()
```

```
For i = 1 To 10
```

```
MsgBox i, vbMsgBoxRight
```

```
Next i
```

```
End Sub
```

21. Do Loop , Do until , Do While עבודה עם לולאות ללא מספר פעמים מוגדר
מראש אלא עד אשר מתקיים תנאי מסוים . לדוגמה :

```
Sub do_loop()
```

```
Do
```

```
MsgBox "CTRL + PAUSE BREAK TO EXIT"
```

```
Loop
```

```
End Sub
```

```
Sub Do_while_pass_Is_One()  
Do Until Password = 1  
Password = InputBox("Enter your password")  
Loop  
End Sub
```

```
Sub Do_while_pass_Is_One()  
Do Until Password = 1  
Password = InputBox("Enter your password")  
Loop  
End Sub
```

כתיבת פונקציה מותאמת אישית באקסל – מתווספת לרשימת הפונקציות של התוכנה בקטגוריה "פונקציות מוגדרות של המשתמש" (תרגיל 12)

```
Function Shekel_Kolel_Maam(Dollar_rate, Maam, Price_in_Shekels)  
Shekel_Kolel_Maam = Dollar_rate * Maam * Price_in_Shekels  
End Function
```

22. Application.WorksheetFunction קריאה לפונקציה של האקסל בתוך הקוד

```
Sub Sum_In_Dollars()  
Range("f16").Value =  
Application.WorksheetFunction.Sum(Range("f9:f15"))  
End Sub
```

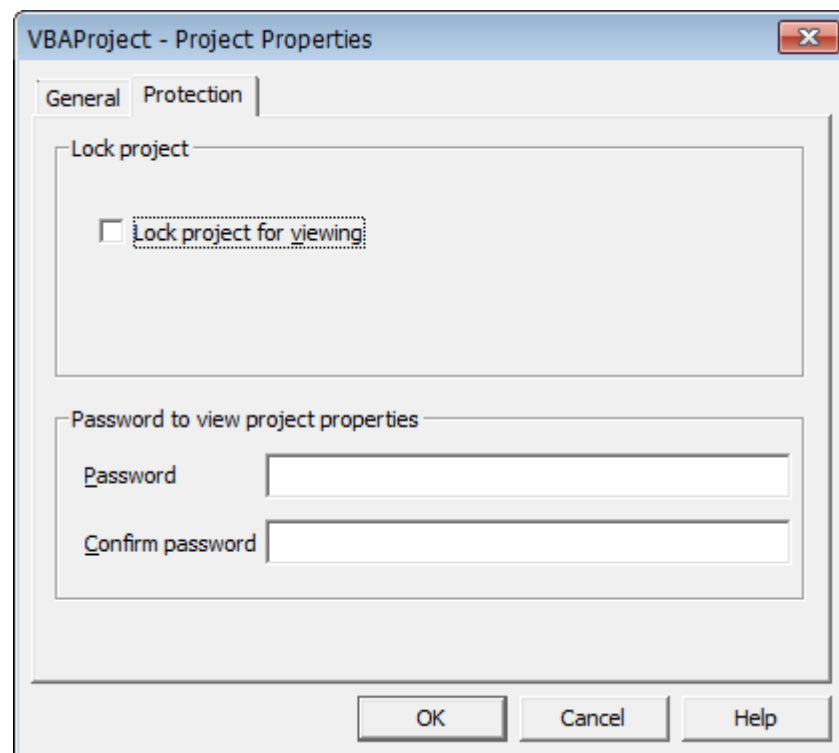
```
Sub AVRG_In_Dollars()  
Range("f17").Value =  
Application.WorksheetFunction.Average(Range("f9:f15"))  
End Sub
```

23. השמעת צפצוף וביצוע השהייה(תרגיל מסכם).

```
Sub beep3()  
Dim i As Integer  
For i = 1 To 3  
beep  
Application.Wait Now + TimeSerial(0, 0, 1)  
Next  
End Sub
```

24.

הגנה על גליון מצפייה בקוד . קליק ימני בעורך vbaproject properties
עבור ללושונות protection והזן סיסמא. סמן וי בנעילה לצפייה ולחץ אישור.



25.

מקרו אירוע – ברמת גליון , חוברת עבודה, כאשר פותחים או משנים וכד' יש לכתוב תא הפקודה בתוך הגליון או חוברת העבודה לא בתוך המודול.(ראה תרגיל מסכם)

```
Private Sub Workbook_BeforeSave(ByVal SaveAsUI As Boolean, Cancel As Boolean)  
Answer = MsgBox("Do you really want to save the workbook?", vbYesNo)
```



```
If Answer = vbNo Then Cancel = True  
End Sub
```

26. התמודדות עם שגיאות ובאגים במהלך ריצת הקוד

על ידי שימוש בפקודה `OnError` ניתן להורת לתוכנה מה לבצע. אפשרות אחת היא השימוש בפקודה `GoTo` אשר עובר לבצע את הפקודות הנמצאות בתווית (`label`):

```
On Error Goto ErrHandler:  
N = 1 / 0 ' cause an error  
'  
' more code  
'  
Exit Sub  
ErrHandler:  
' error handling code  
Resume Next  
End Sub
```

הפקודה `Resume` מורה ל `VBA` להמשיך להריץ את הקוד מהשורה שיצרה את הבעיה. אם הבעיה לא נפתרה הקוד יכנס למעגל אין סופי עד שהשגיאה תיפתר. לכן שימוש ב `Resume Next` יכול לעזור. מורה ל `VBA` להמשיך לרוץ מיד לאחר השורה שגרמה לשגיאה.
אפשרויות שימוש בפקודה :

```
Resume  
Resume Next  
Resume <label>
```

מבנה לדוגמה :

```
On Error GoTo ErrHandler:  
N = 1 / 0  
'  
' code that is skipped if an error occurs  
'  
Label1:  
'  
' more code to execute  
'
```

Exit Sub

ErrorHandler:

' go back to the line at Label1:

Resume Label1:

קוד לדוגמה העושה שימוש בשיטות השונות.

Sub MyMacro3Resume()

Dim Result As Integer

Dim ErrMsg As String

Dim MyWorkbook As Workbook

' Run the Error handler "ErrorHandler" when an error occurs.

On Error GoTo Errhandler

ChDrive "B:"

ChDir "B:\"

ChDir "B:\XLfiles"

Workbooks.Open "Book1.xls"

NewWorkbook:

' Disable the error handler.

On Error GoTo 0

Set MyWorkbook = ActiveWorkbook

MsgBox "The destination workbook is " & MyWorkbook.Name

' Exit the macro so that the error handler is not executed.

Exit Sub

Errhandler:

Select Case Err

Case 68, 75: ' Error 68: "Device not available"

' Error 75: "Path/File access error

ErrMsg = "There is an error reading drive B. Please " & _

"insert a disk and then press OK to continue or " & _

"press Cancel to end this operation."

Result = MsgBox(ErrMsg, vbOKCancel)

' Resume at the line where the error occurred if the user

' clicks OK; otherwise end the macro.

If Result = vbOK Then Resume Next

Case 76: ' Error 76: Path not found

ErrMsg = "The disk in drive B does not have an XLFiles " & _

"directory. Please insert the correct disk."

Result = MsgBox(ErrMsg, vbOKCancel)

' Resume at the line where the error occurred if the user

' clicks OK; otherwise end the macro.

If Result = vbOK Then Resume Next

Case Else: ' A different error occurred.

ErrMsg = "An error has occurred opening " & _

"B:\XLFiles\Book1.xls. Use the active workbook as " & _

"the destination?"

Result = MsgBox(ErrMsg, vbYesNo)

' Resume at the label "NewWorkbook" if the user clicks Yes;

' otherwise end the macro.

If Result = vbYes Then Resume NewWorkbook

End Select

' End the macro.

End Sub

27. משתנים מערכיים

מאפשרים לנו לאחסן משתנה אחד מספר רב של נתונים כאשר בתחילה מגדירים את גודל המערך וניתן להגיע לכל אחד מהנתונים לפי שם המשתנה ומספר בסוגרים שבתוכו לפי הדוגמה למטה. שימושי מאוד בקליטת נתונים מתוך בסיס נתונים.

```
Private Sub students_names()
```

```
Dim StudentName(1 To 5) As String
```

```
For i = 1 To 5
```

```
StudentName(i) = InputBox("Enter student Name")
```

```
Cells(i, 1) = StudentName(i)
```

```
Next
```

```
End Sub
```

```
Private Sub students_marks()
```

```
Dim StudentName(3) As String, StudentID(3) As String, StudentMark(3) As Single
```

```
For i = 1 To 3
```

```
StudentName(i) = InputBox("Enter student Name")
```

```
StudentID(i) = InputBox("Enter student ID")
```

```
StudentMark(i) = InputBox("Enter student Mark")
```

```
Cells(i, 1) = StudentName(i)
```

```
Cells(i, 2) = StudentID(i)
```

```
Cells(i, 3) = StudentMark(i)
```

```
Next
```

```
End Sub
```

דוגמה למשתנים מערכיים **דו מימדים** שניתן לקלוט לתוכם 2 נתונים בכל פעם.

```
Private Sub SalesPerformance()  
Dim SalesVolume(1 To 5, 1 To 2) As Single  
Dim SalesPerson As Integer, Day As Integer  
    For SalesPerson = 1 To 5  
        For Day = 1 To 2  
            SalesVolume(SalesPerson, Day) = InputBox("Enter Sales Volume")  
Cells(SalesPerson, Day) = SalesVolume(SalesPerson, Day)  
            Next Day  
        Next SalesPerson  
    End Sub
```

28. משתנה גלובלי

ניתן להגדיר משתנה גלובלי בשורה ההצהרה (השורה הראשונה) משתנה שתקף לכל המאקרואים. הערך של המשתנה לא מתאפס ממקרו אחד לשני ולכן ניתן לבצע דרכו חישובים בין מאקרואים שונים מאחר והוא שומר על הערך ביציאה ממאקרו אחד לשני. שנרץ את הפונקציה השנייה לאחר הראשונה X יהיה שווה 21 כי הוא שומר את הערך שלו מהפונקציה הראשונה.

(תזכורת : פקודה המאלצת להכריז על משתנים) Option Explicit

הכרזה של מתנה גלובלי בשורה ההצהרה – Dim x As Long

```
Sub global_ex1()
```

```
x = 20
```

```
MsgBox x
```

```
End Sub
```

```
Sub global_Ex2()
```

```
x = x + 1
```

```
MsgBox x
```

```
End Sub
```

29. משתנה סטטי

פועל רק בתוך מקארו אחד. שומר על הערך של המשתנה בסיום ריצת הקוד.
כל פעם שמריצים את הקוד הוא מתחיל מהערך הקודם ששמר ביציאה.
המשתנה מתאפס כאשר סוגרים את חוברת העבודה ופותחים מחדש.

```
Sub static_var()
```

```
Static v
```

```
v = v + 1
```

```
MsgBox v
```

```
End Sub
```

30. משתנה קבוע

ניתן להגדיר בשורה הראשונה את המשתנה הקבוע. משתנה זה טוב לכל המאקרואים.
ניתן לדרוס את הערך שלו בתוך מאקרו מסוים על ידי הצהרה מחדש של ערכו.
בפונקציה הראשונה הערך של f יהיה 5 לפי הקבוע למעלה.
בפונקציה השנייה הצהרנו מחדש על המשתנה ושינינו את ערכו לכן יהיה 6.

Const f = 5 – הצהרה על משתנה קבוע בשורת ההצהרה

```
Sub const1()
```

```
MsgBox f
```

```
End Sub
```

```
Sub const2()
```

```
Dim f As Integer
```

```
f = 6
```

```
MsgBox f
```

```
End Sub
```

30. auto_open

שם מיוחד ושמור לפונקציה אשר רצה אוטומטית עם פתיחת חוברת העבודה.

דוגמה : פונקציה זו תגרום להקפת הודעה על המסך של התוכנה ברגע שהיא נפתחת.

```
Sub auto_open()
```

```
MsgBox "Hello"
```

```
End Sub
```

31. For Each לולאות שאוספים ולמערכים

מאפשר להריץ לולאה על אוספים כגון שורות , טווחים , גליונות וכד'
בדוגמה הפקודה תרוץ על כל הגליונות ותקפיץ הודעה עם השם שלהם.
ניתן להשתמש בכל שם במקום sht (שם של משתנה) דוגמאות:

```
Sub my_for_loop3()  
For Each sht In ActiveWorkbook.Worksheets  
MsgBox sht.Name  
Next sht  
End Sub
```

```
Sub my_for_loop4()  
For Each pvt In ActiveSheet.PivotTables  
MsgBox pvt.Name  
Next pvt  
End Sub
```

```
Sub LoopRange()  
Dim rng As Range  
Dim row As Range  
Dim cell As Range  
Dim i As Integer  
Set rng = Range("A1:C2")  
For Each row In rng.Rows  
For Each cell In row.Cells  
i = i + 1 'Do Something  
MsgBox i  
Next cell  
Next row  
End Sub
```

32. Data Type טבלת המרכזת את כל סוגי המשתנים –

Data Type	Size in Memory	Description	Range of Values
Byte	1 byte	Represents an unsigned (non-negative) number - often used for binary data	0 to 255
Boolean	2 bytes	A simple True or False value	True or False
Integer	2 bytes	Integer (no decimals)	-32,768 to +32,767
Long	4 bytes	Long Integer (no decimals)	- 2,147,483,648 to +2,147,483,647
Single	4 bytes	Single Precision Floating Point Number	-3.4e38 to +3.4e38
Double	8 bytes	Double Precision Floating Point Number	-1.8e308 to +1.8e308
Currency	8 bytes	A Floating Point Number with a fixed number of decimal places	- 922,337,203,685,477.5808 to +922,337,203,685,477.5807
Date	8 bytes	Date & Time - The Date type is represented internally by a floating point number. The integer part of the number represents the date, and the decimal portion represents the time.	1st January 100 to 31st December 9999
Object	4 bytes	A reference to an object	Any Object Reference
String	varies	Holds a series of characters. The String type can be defined to have a fixed or a variable length, although it is most commonly define to have a variable length	Fixed - Up to 65,500 characters Variable - Up to approx. 2 billion characters
Variant	varies	Can hold Dates, Floating Point Numbers or Strings of Characters, and should therefore be used when you are not sure what type of data to expect.	Number - same as Double Type String - same as String Type

33. Set

הפקודה יוצרת אובייקט חדש (אובייקטים הם המרכיבים של התוכנה כגון : שורות , עמודות, גליונות , טווחים וכד') בצורה זו ניתן לעבוד בקלות עם אובייקטים במקום לקרוא להם בשם מורכב.

Dim NewRange As Range
Set NewRange = Range("A2:A6")

' Cells

Set myRange = ActiveSheet.Range("A1:D5")

' Row

Set myRange = ActiveSheet.Rows(1)

' Column

Set myRange = ActiveSheet.Columns(2)

' Group of Cells

Set myRange = Application.Union(_
 ActiveSheet.Range("A1:D1"), _
 ActiveSheet.Range("C3:C5"))

' Select method

myRange.Select

בהצלחה

יואב.