[50 Excel VBA Oral Interview Questions](http://www.excelitems.com/2010/12/excel-vba-interview-questions.html)

**Ques 01. What is the difference between ByVal and ByRef and which is default ?**

**Solution: ByRef:** If you pass an argument by reference when calling a procedure the procedure access to the actual variable in memory. As a result the variable's value can be changed by the procedure.  
**ByVal:** If you pass an argument by value when calling a procedure the variable's value can be changed with in the procedure only outside the actual value of the variable is retained.  
**ByRef is default:** Passing by reference is the default in VBA. If you do not explicitly specify to pass an argument by value VBA will pass it by reference.

**Ques 02. What is the meaning of Option Explicit and Option Base?**

**Solution:** Option Explicit makes the declaration of Variables Mandatory while Option Base used at module level to declare the default lower bound for array subscripts. For eg. Option Base 1 will make the array lower bound as 1 instead of 0.

**Ques 03. What are various data type and their size?**

Solution:   
i) The **Boolean** data type has only two states, True and False. These types of variables are stored as 16-bit (2 Byte) numbers, and are usually used for flags.  
ii) The **Byte** data type is an 8-bit variable which can store value from 0 to 255.   
iii) The **Double** data type is a 64-bit floating point number used when high accuracy is needed.   
iv) The **Integer** data type is a 16-bit number which can range from -32768 to 32767. Integers should be used when you are working with values that can not contain fractional numbers. In case, you're working over 32767 rows use Long as data type.  
v) The **Long** data type is a 32-bit number which can range from -2,147,483,648 to 2,147,483,647.  
vi) The **Single** data type is a 32-bit number ranging from -3.402823e38 to -1.401298e-45 for negative values and from 1.401298e-45 to 3.402823e38 for positive values. When you need fractional numbers within this range, this is the data type to use.  
vii) The **String** data type is usually used as a variable-length type of variable. A variable-length string can contain up to approximately 2 billion characters. Each character has a value ranging from 0 to 255 based on the ASCII character set.

**Ques 04. Difference between ActiveWorkbook and ThisWorkbook.**

**Solution:** ThisWorkbook refers to the workbook where code is being written while ActiveWorkbook refers to the workbook which is in active state with active window. In case of only one workbook open, ActiveWorkbook is same as ThisWorkbook.

**Ques 05. Code to find a Last used Row in a column or Last used column of a Row.**

**Solution:** Last Row in a column can be find using End(xlUp) and Last Column in a row can be find using End(xlToLeft). For e.g. Range("A1048576").End(xlUp).Row gives last used row of Column A.

**Ques 06. Difference between ActiveX and Form Controls.**

**Solution:**   
i) Forms controls can be used on worksheets and chart sheets. Forms controls can also be placed within embedded charts in Classic Excel (though not in Excel 2007). ActiveX controls can only be used on worksheets. ActiveX controls do not work in MacExcel.  
ii) The Forms controls aren’t very complicated, and they have been part of Excel for longer (they were used in Excel 5/95’s dialog sheets) than the Controls Toolbox (Excel 97), so it stands to reason that they’d be more seamlessly integrated. Being newer, the ActiveX controls have richer formatting possibilities. Both can link to cells and ranges in the worksheet.

**Ques 07. What is the difference b/w Functions and Subroutines?**

**Solution:**   
i) Subroutines never return a value but functions does return values.  
ii) A function could not change the values of actual arguments whereas a subroutine could change them.

**Ques 08. How to debug a VBA code?**

**Solution:** Using Breakpoints(F9), Step-by-step execution (F8), Debug.Print & Immediate Window and Watch window.

**Ques 09. Draw basic Excel Object Model.**

**Solution:** Application --> Workbooks --> Worksheets --> Range / Chart

**Ques 10. What are properties, methods, events and objects?**

**Solution:** For details click here --> http://msdn.microsoft.com/en-us/library/ms172576%28VS.80%29.aspx  
All the controls in the ToolBox except the Pointer are **objects** in Visual Basic. These objects have associated properties, methods and events.   
A **property** is a named attribute of a programming object. Properties define the characteristics of an object such as Size, Color etc. or sometimes the way in which it behaves.  
A **method** is an action that can be performed on objects. For example, a cat is an object. Its properties might include long white hair, blue eyes, 3 pounds weight etc. A complete definition of cat must only encompass on its looks, but should also include a complete itemization of its activities. Therefore, a cat's methods might be move, jump, play, breath etc.  
Visual Basic programs are built around **events**. Events are various things that can happen in a program. Let us consider a TextBox control and a few of its associated events to understand the concept of event driven programming. The TextBox control supports various events such as Change, Click, MouseMove and many more that will be listed in the Properties dropdown list in the code window for the TextBox control. We will look into a few of them as given below.  
\* The code entered in the Change event fires when there is a change in the contents of the TextBox  
\* The Click event fires when the TextBox control is clicked.  
\* The MouseMove event fires when the mouse is moved over the TextBox

**Ques 11. How to hide a worksheet so that a user cannot unhide it?**

**Solution:** Use Sheet's visible property and set it to **xlSheetVeryHidden** . For eg. Sheets(1).Visible = xlSheetVeryHidden will super hide the first worksheet of the workbook.

**Ques 12. Union is used for \_\_\_\_\_\_\_\_\_\_\_\_\_ ?**

**Solution:** To unite the different ranges depending on the logic. It is similar to set union, here range works as set. For eg. Set nrange = Union(rng1,rng2)

**Ques 13. Which are the 2 macro languages and which do you use ?**

**Solution:** XLM (used in Excel 97 or before) and VBA(used for 2000 and after). Obviously, VBA is in use these days.

**Ques 14. Can you lock cells such that only specific users can modify them?**

**Solution:** There is a option "Allow users to edit ranges" can be used for this purpose.

**Ques 15. How can you add a drop-down list to a cell so the user can choose a value from the list?**

**Solution:** Using 'Data Validation'.

**Ques 16. How can you increase the number of rows in a worksheet?**

**Solution:** No one can't. They're fixed as 65536(2^16) in Excel 2003 or before and 1048576(2^20) in Excel 2007 & Excel 2010.

**Ques 17. How can you increase the number of columns in a worksheet?**

**Solution:** No one can't. They're fixed as 256(2^8) in Excel 2003 or before and 16384(2^14) in Excel 2007 & Excel 2010.

**Ques. 18. How will you distribute a workbook such that it can't be copied using macro or anything?**

**Solution:** We can create a workbook which cannot be modified but can not create a workbook which can't be copied.(It depends on system security, it has nothing to do with Excel or VBA)

**Ques 19. Your colleague created a dashboard and when you enter a value, it appears with two decimal places. For example, when you enter 265 it shows up as 2.65. What's wrong?**

**Solution:** By chance Excel's fixed-decimal mode was turned on. To return to normal,   
**Excel 2003 -->**  
Click Tools and then Options to display the Options dialog box.  
Then click the Edit tab and remove the check mark from the "Fixed decimal " option.  
**Excel 2007 -->**   
Click Office button on Top-Left corner and click 'Excel Options'.  
Go to Advanced and Uncheck 'Automatically insert a decimal point' option.  
**Excel 2010 -->**  
Click File button on Top-Left corner and click 'Excel Options'.  
Go to Advanced and Uncheck 'Automatically insert a decimal point' option.  
  
Of course, this feature can be useful when entering some types of data, but most of the time, you'll want to keep the fixed-decimal mode turned off.

**Ques 20. How can you prevent a user for adding or deleting sheets?**

**Solution:** You need to protect the workbook's structure.  
**Excel 2003 -->**  
Select Tools - Protection - Protect Workbook. In the Protect Workbook dialog box, make sure that the Structure checkbox is checked.   
**Excel 2007/2010 -->**  
Go to Review --> Click 'Protect Workbook' --> Click 'Protect Structure and Windows'   
  
If you specify a password, that password will be required to unprotect the workbook. When a workbook's structure is protected, the user may not:  
         \* Add/Delete a sheet  
         \* Hide/Unhide a sheet  
         \* Rename a sheet  
         \* Move a sheet

**Ques 21. What types of workbook protection are available?**

**Solution:** Excel provides three ways to protect a workbook:  
         \* Require a password to open the workbook  
         \* Prevent users from adding sheets, deleting sheets, hiding sheets, and unhiding sheets  
         \* Prevent users from changing the size or position of windows

**Ques 22. The Font dialog box allows you to select different Fonts, their style, their size, and some other special effects. How do you bring up this Font dialog box?**

**Solution:** Use Application.Dialogs(xlDialogFont).Show or Application.Dialogs(xlDialogFormatFont).Show to load font dialog box from Excel VBA.

**Ques 23. What is ADO, OLEDB & ODBC?**

**Solution:**   
ADO : ActiveX Data Objects is universal data access framework that encompasses the functionality of DAO.  
ODBC : Open Database Connectivity(ODBC) is a windows technology that lets a database client application connect to a external database.  
OLEDB : Low level programming interface designed to access a wide variety of data access Object Linking and Embedding (OLE).

**Ques 24. How to set the custom paper size in Excel Object through VB?**

**Solution:** Activesheet.PageSetup.PaperSize = xlPaperLetter (Similarly xlPaperA4 or xlPaperLegal etc.)

**Ques 25. What is the method for returning more than one values from a function in VB?**

**Solution:** Any of the three methods can be used:  
i) Create a class with the properties you require to return and then return the object of the class from the function.  
ii) Using ByRef for the values.  
iii) Return an array of the values.

**Ques 26. Does VBA supports OOP principles?**

**Solution:** Yes because VBA is VB6.0 based which is an Object Based Programming Language and is also known as 'Event Driven Programming' and it supports Polymorphism, Encapsulation and partially Inheritance.

**Ques 27. To set the command button for ESC, Which property needs to be changed?**

**Solution:** Set Cancel property of Button to True on the Form.

**Ques 28. What is Type Library and its purpose?**

**Solution:** Type libraries are files that explicitly describe some or all of the contents of components. This includes information about the methods properties constants and other members exposed by the component. Development tools such as Visual Basic make use of the information contained in the type library to help you as a developer access and use the component. In addition type libraries provide a convenient way to include a simple level of descriptive documentation for component members. You can use them through 'Tools --> References' in VBE.

**Ques 29. How do you use the Getsetting and Savesetting functions to read and write registry settings?**

**Solution:** A computer registry can be used to store configuration settings and application initialization. We can use Getsetting function to read registry settings and save settings function to write registry settings. Application name, section, key, setting, and default are to be specified for registry modifying. It is advisable to know about your computer settings before modifying registry settings.

**Ques 30. What is a Variant, what the pros and cons of its use?**

**Solution:** Variant data type is able to hold any other data type, including numbers, strings, dates, and object references. A Variant's descriptor is only 16 bytes long (4 short words for the type, and 2 long words for the data, or data pointer).  
Pros: You cannot use Null with any variable type other than Variant.  
You don't need to worry about what you have declared a variable as.  
When a Variant has been declared but not assigned a value, it contains the special value Empty.  
Cons: A developer may not remember and misuse a variable assigning any value to it which will be type-casted without errors.

**Ques 31. Give technical reasons which made Microsoft withdraw its support for VBA in Mac?**

**Solution:** The reasons which made Microsoft drop its support to VBA are as follows, Microsoft visual basic relies heavily on machine code which was written for Power PC architecture. Also it would take another two years for developing VBA support for its architecture. It also states that Microsoft will incorporate VBA in the next script of office release for Mac.

**Ques 32. What is a volatile function?**

**Solution:** Volatile functions are a type of function that will always recalculate. That means whenever Excel needs to calculate any part of the worksheet, those cells containing volatile functions will also calculate.

**Ques 33. Give some examples of Volatile function.**

**Solution:**   
Some of Excel’s functions are obviously volatile: RAND(), NOW(), TODAY()  
Others are less obviously volatile: OFFSET(), CELL(), INDIRECT(), INFO()  
Some are volatile in some versions of Excel but not in others: INDEX()became non-volatile in Excel 97.  
A number of functions that are documented by Microsoft as volatile do not actually seem to be volatile when tested:  
INDEX(), ROWS(), COLUMNS(), AREAS()  
and CELL("Filename") IS volatile although a MSKBN article says its not.  
One particular syntax of SUMIF is volatile in Excel 2002 and subsequent versions:  
=SUMIF(A1:A4,">0",B1) is volatile whereas =SUMIF(A1:A4,">0",B1:B4) is not volatile.

**Ques 34. How do you make a UDF volatile?**

**Solution:** By adding Application.Volatile statement to it. It must be the first line of your User Defined Function.

**Ques 35. Is it possible to apply 'Application.Volatile(False)' to a volatile public function like INDEX and make it not volatile?**

**Solution:** Actually INDEX is not a volatile function, even though some MicroSoft documentation says it is. Anyway no its not possible to apply Application.Volatile(False) to a built-in Excel function except by duplicating what the built-in function does inside a UDF.

**Ques 36. What is Excel dependency tree?**

**Solution:** Dependency trees are excel way of minimizing the calculation by tracking what has changed since last calculation. It allows Excel to recalculate only:  
        \* Formulae/Names that have changed.  
        \* Formulae containing Volatile Functions  
        \* Formulae dependent on changed or volatile formulae or cells or names.  
  
Excel determines dependencies by looking at the cells referred to by each formula and by the argument list of each function.  
Dependency trees are immediately updated whenever a formula is entered or changed.  
To force the dependency trees to be rebuilt and all formulae calculation use CTRL+ALT+SHIFT+F9.

**Ques 37. What are keyboard shortcuts and their equivalent VBA methods for Formulae calculation and building Excel dependency trees?**

**Solution:**

|  |  |  |
| --- | --- | --- |
| Shortcut Combination | VBA Equivalent | Meaning |
| F9 | Application.Calculate | Recalculate |
| Ctrl+Alt+F9 | Application.CalculateFull | Full Calculation |
| Ctrl+Alt+Shift+F9 | Application.CalculateFullRebuild | Rebuild Excel Dependency Tree   and Full Calculation |
| Shift+F9 | Sheets(1).Calculate | Calculate Worksheet |

**Ques 38. What does Range.Dirty used for ?**

**Solution:** To add the specified cells to the list of cells requiring calculation at the next recalculation.

**Ques 39. How do you check the Calculation state ?**

**Solution:** Pretty simply, using the Application.CalculationState property which tells if calculation has completed ( xlDone ), is pending ( xlPending) , or is in process ( xlCalculating ).

**Ques 40. How do you define Excel Calculation Interruption Key to stop the calculation?**

**Solution:** Using Application.CalculationInterruptKey= XlAnyKey | XLEscKey | XlNokey.  
Remember using XlNokey, calculation cannot be interrupted.

**Ques 41. What does 'Workbook.ForceFullCalculation' do?**

**Solution:** When this property is set to True, dependencies are not loaded at open, the dependency dirty chain is not updated, and every calculation of the workbook is a full calculation rather than a recalculation.  
  
If you have a workbook that has so many complex dependencies that loading the dependencies at workbook open takes a long time or recalculation takes longer than full calculation, you can use this property to force Excel to skip loading the dependencies and always use full calculation. Also if making a change to the workbook takes a long time in manual mode because of the time taken to dirty all the dependencies of the cell being changed, then setting Workbook.ForceFullCalculation to True will eliminate the delay.  
  
      \* Although this is a workbook property the effect is at Application level rather than workbook level.  
      \* In Excel 2007 setting the property back to False once it has been set to True has no effect on the current Excel session.  
      \* The property is saved and restored with the workbook.  
      \* Once this property has been set to True 'Calculate' shows in the status bar and cannot be removed by using calculation keys such as F9.

**Ques 42. What are excel dependency tree limits?**

**Solution:** There are two limits to the number of dependencies that Excel versions prior to Excel 2007 can track before it must do full calculations instead of recalculations.  
       \* The number of different areas in a sheet that may have dependencies is limited to 65,536.  
       \* The number of cells that may depend on a single area is limited to 8K.  
After the workbook has passed these limits, Excel no longer attempts to recalculate only changed cells. Instead, it recalculates all cells at each calculation.

**Ques 43. Mention those conditions when status bar shows CALCULATE.**

**Solution:** There are five known conditions in which the status bar will show CALCULATE:  
       \* The Calculation Option has been set to Manual and the workbook contains uncalculated formulae. Try setting calculation to Automatic (Tools-->Options-->Calculate). Note that Excel sets the calculation mode from the first workbook opened in a session: when you open two workbooks, one saved in manual mode and one saved in automatic mode, they will both have the calculation mode of the first workbook opened.  
       \* The Iteration Option is turned on and the workbook contains circular references. Check that turning off Iteration (Tools-->Options-->Calculation) and pressing F9 shows "Circular Reference" in the statusbar.  
       \* You are using Excel 2000 without the SR1 update and have a user-defined function that attempts to define a name and depends on a volatile function: see MSKB Q248179  
       \* You have hit one of Excels limits for tracking dependencies.  
       \* You are using Excel 2007 and have set Workbook.ForceFullCalculation to True

**Ques 44. What do you know about multi threaded calculation?**

**Solution:** Excel 2007 can split calculation across multiple processors or cores. When Excel 2007 loads a workbook, it determines from the operating system how many processors are available and then creates a separate calculation thread for each processor. These threads can then run in parallel. The beauty of this system is that it scales extremely well with the number of processors.  
  
Most workbooks show a significant improvement in calculation speed on a system with multiple cores. The degree of improvement depends on how many independent calculation trees the workbook contains. If you make a workbook that contains one continuous chain of formulas, it will not show any multithreaded calculation (MTC) performance gain, whereas a workbook that contains several independent chains of formulas will show gains close to the number of processors available.

**Ques 45. How can we dial a phone number?**

**Solution:** Shell command present in VBA can be used to start the dialer present in windows operating system. Phone number can be used to connect to your modem. With the use of shell and sendkeys you can dial to your user. Shell starts windows application and sendkeys inform the window to dial according to the keystrokes of the application. A macro can be used to start the cardfile program which activates the auto dialer feature.

**Ques 46. What do you know about the interpretation features of VBA?**

**Solution:** VBA is licensed to Microsoft and this compatible with and only Microsoft products. Code written is compiled by an intermediate language called P-code and this is stored in hosting applications such as Excel, Word and Access. The intermediate code is interpreted by a virtual machine. This code and intermediate language is the exclusive right of Microsoft.

**Ques 47. Explain about insert module and Goal Seek functions present in VBA?**

Solution: The chief use of VBA is to make use of its special function which helps in repeated actions. Goal seek function helps to reduce manual entry of the code each and every time. This solves the problem of repeated function entry by automating functions and actions. Sub routines are inserted into the using the VBA editor and command insert module.

**Ques 48. State the difference between Visual Basic, VB Script and Visual Basic for Applications?**

**Solution:** Visual basic is useful if you are planning to develop your programs from scratch.This language helps you in developing Active x controls, exe files, etc.  
VB script is a powerful tool, through which you can create small scale applications on web pages, automation applications, etc. Integrated development environment is not present for VB script.  
Visual Basic for Applications are very useful in automating your existing application. VB application is useful for developing already existing applications.

**Ques 49. Write a macro to select all the non-blank cells of Activesheet ?**

**Solution:**   
Sub NonBlankCells()  
    On Error Resume Next  
    Union(Cells.SpecialCells(xlCellTypeFormulas, 23), Cells.SpecialCells(xlCellTypeConstants, 23)).Select  
    If Err.Number <> 0 Then  
        Cells.SpecialCells(xlCellTypeFormulas, 23).Select  
    Else  
        Exit Sub  
    End If  
    If Err.Number <> 0 Then  
        Cells.SpecialCells(xlCellTypeConstants, 23).Select  
    Else  
        Exit Sub  
    End If  
    On Error GoTo 0  
End Sub  
  
**Ques 50. What is the difference between UsedRange and CurrentRegion properties ?**  
**Solution:**  
**i)** The current region is a range bounded by any combination of blank rows and blank columns.   
This property is useful for many operations that automatically expand the selection to include the entire current region, such as the AutoFormat method. This property cannot be used on a protected worksheet.  
The UsedRange property is used to select the range of used cells on a worksheet. It returns a Range object that represents the used range on the specified worksheet.  
ii) Every non-blank cell got its CurrentRegion and its keyboard shortcut is Ctrl+Shift+Spacebar.  
iii) There can be many current regions but there is only one used range in a worksheet.

### [17 ways to Optimize VBA Code for FASTER Macros](http://exceldailytip.blogspot.com/2010/12/17-ways-to-optimize-vba-code-for-faster.html)

Here is a summary of the article:  
1. Analyze the Logic   
2. Turn off ScreenUpdating  
3. Turn off 'Automatic Calculations'   
4. Disable Events  
5. Hide Page breaks   
6. Use 'WITH' statement  
7. Use vbNullString instead of ""   
8. Release memory of Object variables  
9. Reduce the number of lines using colon(:)   
10. Prefer constants  
11. Avoid Unnecessary Copy and Paste   
12. Clear the Clipboard after Paste  
13. Avoid 'Macro Recorder' style code.   
14. Use 'For Each' than 'Indexed For'  
15. Use 'Early Binding' rather 'Late Binding'   
16. Avoid using Variant  
17. Use Worksheet Functions wherever applicable