Excel Live Coding Interview Guide

Essential Functions & VBA for Mid-Level Data Analysts

# Contents

* 1. Lookup & Reference Functions
* 2. Logical Functions
* 3. Text Functions
* 4. Date & Time Functions
* 5. Statistical & Math Functions
* 6. Array Functions (Dynamic Arrays)
* 7. Pivot Tables
* 8. Data Validation & Conditional Formatting
* 9. VBA Fundamentals
* 10. Common Interview Scenarios

# 1. Lookup & Reference Functions

## VLOOKUP

**Purpose:** Look up a value in the first column of a range and return a value from another column.

**Syntax:** =VLOOKUP(lookup\_value, table\_array, col\_index\_num, [range\_lookup])

**Example:**

* =VLOOKUP(A2, Products!$A$2:$D$100, 3, FALSE)
* Looks up value in A2, searches Products table, returns 3rd column, exact match

**Key Points:**

* FALSE = exact match (most common), TRUE = approximate match
* Can only look LEFT to RIGHT (limitation)
* Use $ signs to lock the table range when copying formula

## HLOOKUP

**Purpose:** Same as VLOOKUP but searches horizontally (in rows instead of columns).

**Syntax:** =HLOOKUP(lookup\_value, table\_array, row\_index\_num, [range\_lookup])

## INDEX/MATCH (Superior to VLOOKUP)

**Purpose:** More flexible lookup - can look in any direction, less prone to errors.

**Syntax:** =INDEX(return\_range, MATCH(lookup\_value, lookup\_range, 0))

**Example:**

* =INDEX(C2:C100, MATCH(A2, B2:B100, 0))
* Find A2 in column B, return corresponding value from column C

**Why INDEX/MATCH > VLOOKUP:**

* Can look LEFT (VLOOKUP cannot)
* Inserting columns won't break the formula
* Faster on large datasets
* More flexible - lookup and return ranges are independent

## Two-Way Lookup (INDEX/MATCH/MATCH)

**Purpose:** Look up value at intersection of row and column.

**Syntax:** =INDEX(data\_range, MATCH(row\_value, row\_headers, 0), MATCH(col\_value, col\_headers, 0))

**Example:**

* =INDEX(B2:M13, MATCH("March", A2:A13, 0), MATCH("Sales", B1:M1, 0))

## XLOOKUP (Excel 365/2021+)

**Purpose:** Modern replacement for VLOOKUP/HLOOKUP/INDEX-MATCH.

**Syntax:** =XLOOKUP(lookup\_value, lookup\_array, return\_array, [if\_not\_found], [match\_mode], [search\_mode])

**Example:**

* =XLOOKUP(A2, Products!A:A, Products!C:C, "Not Found")

**Advantages:**

* Default is exact match (no FALSE needed)
* Can return multiple columns
* Built-in error handling (if\_not\_found)
* Can search from bottom up

# 2. Logical Functions

## IF

**Syntax:** =IF(logical\_test, value\_if\_true, value\_if\_false)

**Example:**

* =IF(A2>=90, "A", IF(A2>=80, "B", IF(A2>=70, "C", "F")))

## IFS (Excel 2019+)

**Purpose:** Multiple conditions without nesting.

**Syntax:** =IFS(condition1, value1, condition2, value2, ...)

**Example:**

* =IFS(A2>=90, "A", A2>=80, "B", A2>=70, "C", TRUE, "F")

## AND / OR / NOT

* =AND(condition1, condition2, ...) - TRUE if ALL conditions are true
* =OR(condition1, condition2, ...) - TRUE if ANY condition is true
* =NOT(condition) - Reverses TRUE/FALSE

**Combined Example:**

* =IF(AND(B2="Sales", C2>10000), "Bonus", "No Bonus")

## IFERROR / IFNA

**IFERROR:** Catches any error and returns alternative value.

* =IFERROR(VLOOKUP(A2, Data, 2, FALSE), "Not Found")

**IFNA:** Catches only #N/A errors (more specific).

* =IFNA(VLOOKUP(A2, Data, 2, FALSE), 0)

## SWITCH (Excel 2019+)

**Purpose:** Compare one value against multiple values.

**Syntax:** =SWITCH(expression, value1, result1, value2, result2, ..., [default])

**Example:**

* =SWITCH(A2, 1, "Jan", 2, "Feb", 3, "Mar", "Unknown")

# 3. Text Functions

|  |  |  |
| --- | --- | --- |
| **Function** | **Purpose** | **Example** |
| LEFT(text, num\_chars) | Extract characters from left | =LEFT("Hello", 2) → "He" |
| RIGHT(text, num\_chars) | Extract characters from right | =RIGHT("Hello", 2) → "lo" |
| MID(text, start, num\_chars) | Extract from middle | =MID("Hello", 2, 3) → "ell" |
| LEN(text) | Count characters | =LEN("Hello") → 5 |
| TRIM(text) | Remove extra spaces | =TRIM(" Hi ") → "Hi" |
| CLEAN(text) | Remove non-printable chars | =CLEAN(A1) |
| UPPER(text) / LOWER(text) | Change case | =UPPER("hello") → "HELLO" |
| PROPER(text) | Capitalize first letters | =PROPER("john doe") → "John Doe" |
| CONCAT(text1, text2, ...) | Join text (Excel 2016+) | =CONCAT(A1, " ", B1) |
| TEXTJOIN(delimiter, ignore\_empty, range) | Join with delimiter | =TEXTJOIN(", ", TRUE, A1:A5) |
| SUBSTITUTE(text, old, new) | Replace text | =SUBSTITUTE("Mr.", "Mr", "Ms") → "Ms." |
| FIND(find\_text, within\_text) | Find position (case-sensitive) | =FIND("o", "Hello") → 5 |
| SEARCH(find\_text, within\_text) | Find position (case-insensitive) | =SEARCH("O", "Hello") → 5 |

## Splitting Text (Common Interview Task)

**Split "John Doe" into First and Last Name:**

* First Name: =LEFT(A1, FIND(" ", A1)-1)
* Last Name: =RIGHT(A1, LEN(A1)-FIND(" ", A1))

**Extract domain from email:**

* =RIGHT(A1, LEN(A1)-FIND("@", A1))

# 4. Date & Time Functions

|  |  |  |
| --- | --- | --- |
| **Function** | **Purpose** | **Example** |
| TODAY() | Current date | =TODAY() → 12/5/2024 |
| NOW() | Current date & time | =NOW() → 12/5/2024 10:30 |
| YEAR(date) | Extract year | =YEAR("3/15/2024") → 2024 |
| MONTH(date) | Extract month (1-12) | =MONTH("3/15/2024") → 3 |
| DAY(date) | Extract day | =DAY("3/15/2024") → 15 |
| WEEKDAY(date, [type]) | Day of week (1-7) | =WEEKDAY("3/15/2024") → 6 (Friday) |
| WEEKNUM(date) | Week number of year | =WEEKNUM("3/15/2024") → 11 |
| EOMONTH(start\_date, months) | End of month | =EOMONTH("3/15/2024", 0) → 3/31/2024 |
| EDATE(start\_date, months) | Add/subtract months | =EDATE("3/15/2024", 2) → 5/15/2024 |
| DATEDIF(start, end, unit) | Difference between dates | =DATEDIF(A1, B1, "M") → months |
| NETWORKDAYS(start, end) | Working days between | =NETWORKDAYS(A1, B1) excludes weekends |
| TEXT(value, format) | Format date as text | =TEXT(A1, "MMMM YYYY") → "March 2024" |

## Common Date Calculations

* Age from birthdate: =DATEDIF(A1, TODAY(), "Y")
* First day of month: =EOMONTH(A1, -1)+1
* Last day of month: =EOMONTH(A1, 0)
* Quarter: =ROUNDUP(MONTH(A1)/3, 0)
* Fiscal year (starts July): =IF(MONTH(A1)>=7, YEAR(A1)+1, YEAR(A1))

# 5. Statistical & Math Functions

## Basic Aggregations

* **SUM(range):** Add all values
* **AVERAGE(range):** Mean of values
* **COUNT(range):** Count numbers only
* **COUNTA(range):** Count non-empty cells
* **COUNTBLANK(range):** Count empty cells
* **MAX(range) / MIN(range):** Largest/smallest value
* **MEDIAN(range):** Middle value
* **MODE(range):** Most frequent value
* **STDEV(range):** Standard deviation (sample)
* **VAR(range):** Variance

## Conditional Aggregations (CRITICAL for Interviews)

**SUMIF / COUNTIF / AVERAGEIF (Single Condition)**

* =SUMIF(range, criteria, [sum\_range])
* =SUMIF(A:A, "Sales", B:B) → Sum column B where column A = "Sales"
* =COUNTIF(A:A, ">100") → Count cells > 100
* =COUNTIF(A:A, "\*apple\*") → Count cells containing "apple"

**SUMIFS / COUNTIFS / AVERAGEIFS (Multiple Conditions)**

* =SUMIFS(sum\_range, criteria\_range1, criteria1, criteria\_range2, criteria2, ...)
* =SUMIFS(C:C, A:A, "Sales", B:B, ">2024-01-01") → Sum C where A="Sales" AND B>date
* =COUNTIFS(A:A, "Sales", B:B, ">=100", B:B, "<500") → Between 100 and 500

## Ranking Functions

* =LARGE(range, k) → k-th largest value. =LARGE(A:A, 3) → 3rd largest
* =SMALL(range, k) → k-th smallest value
* =RANK(number, range, [order]) → Rank of a number. 0=descending, 1=ascending
* =PERCENTILE(range, k) → Value at k-th percentile. =PERCENTILE(A:A, 0.9) → 90th percentile

## Rounding Functions

* =ROUND(number, decimals) → Standard rounding
* =ROUNDUP(number, decimals) → Always round up
* =ROUNDDOWN(number, decimals) → Always round down
* =CEILING(number, significance) → Round up to nearest multiple
* =FLOOR(number, significance) → Round down to nearest multiple

# 6. Array Functions (Dynamic Arrays - Excel 365)

## FILTER

**Purpose:** Filter a range based on criteria (spills results).

**Syntax:** =FILTER(array, include, [if\_empty])

**Examples:**

* =FILTER(A2:D100, C2:C100="Sales") → All rows where col C = "Sales"
* =FILTER(A2:D100, (B2:B100>100)\*(C2:C100="Sales")) → Multiple conditions (AND)
* =FILTER(A2:D100, (B2:B100>100)+(C2:C100="Sales")) → Multiple conditions (OR)

## SORT

**Syntax:** =SORT(array, [sort\_index], [sort\_order], [by\_col])

* =SORT(A2:C100, 2, -1) → Sort by 2nd column, descending
* =SORT(FILTER(A2:C100, B2:B100>100), 3, 1) → Filter then sort

## UNIQUE

**Syntax:** =UNIQUE(array, [by\_col], [exactly\_once])

* =UNIQUE(A2:A100) → Distinct values in column A
* =UNIQUE(A2:C100) → Unique rows across columns A-C

## SORTBY

* =SORTBY(array, by\_array1, [order1], ...) → Sort by external column
* =SORTBY(A2:B100, C2:C100, -1) → Sort A:B by column C descending

## SEQUENCE

* =SEQUENCE(rows, [cols], [start], [step]) → Generate number series
* =SEQUENCE(12, 1, 1, 1) → 1 to 12 vertically

## LET (Define Variables)

**Purpose:** Create named variables within a formula for reuse and readability.

**Syntax:** =LET(name1, value1, [name2, value2], ..., calculation)

**Example:**

* =LET(sales, SUMIF(A:A,"Sales",B:B), costs, SUMIF(A:A,"Costs",B:B), sales-costs)

# 7. Pivot Tables

## Creating a Pivot Table

1. Select your data (include headers)

2. Insert → PivotTable

3. Choose New Worksheet or Existing Worksheet

4. Drag fields to: Rows, Columns, Values, Filters

## Pivot Table Components

* **Rows:** Categories displayed as rows (e.g., Product, Region)
* **Columns:** Categories displayed as columns (e.g., Month, Year)
* **Values:** Numbers to aggregate (Sum, Count, Average, etc.)
* **Filters:** Dropdown filters for the entire pivot

## Key Pivot Table Skills for Interviews

* Grouping dates (by Month, Quarter, Year)
* Calculated fields: Insert → Fields, Items & Sets → Calculated Field
* Show values as: % of Grand Total, % of Column, Running Total, Rank
* Slicers: Visual filters (Insert → Slicer)
* Refresh data: Right-click → Refresh
* Drill down: Double-click a value to see underlying data
* Sorting and filtering within pivot
* Multiple value fields (e.g., Sum and Count)

## GETPIVOTDATA Function

**Purpose:** Extract specific values from a pivot table.

* =GETPIVOTDATA("Sales", $A$3, "Region", "East", "Year", 2024)

# 8. Data Validation & Conditional Formatting

## Data Validation

**Location:** Data → Data Validation

* **Whole Number / Decimal:** Restrict to numbers within range
* **List:** Dropdown from range or comma-separated values
* **Date / Time:** Restrict to date/time range
* **Text Length:** Limit character count
* **Custom:** Formula-based validation

**Dependent Dropdown Example:**

* List source: =INDIRECT(A1) where A1 contains a named range

## Conditional Formatting

**Location:** Home → Conditional Formatting

* **Highlight Cell Rules:** Greater than, Less than, Equal to, Text Contains
* **Top/Bottom Rules:** Top 10, Bottom 10%, Above/Below Average
* **Data Bars:** Horizontal bars showing relative values
* **Color Scales:** 2-color or 3-color gradient based on value
* **Icon Sets:** Arrows, flags, shapes based on value
* **New Rule → Formula:** Custom formula: =A1>B1 or =$A1="Complete"

**Interview Tip:** When using formulas in conditional formatting, use mixed references (e.g., $A1) to apply formatting across rows/columns correctly.

# 9. VBA Fundamentals

## Getting Started

* Open VBA Editor: Alt + F11
* Insert Module: Insert → Module
* Run Macro: F5 or Alt + F8
* Record Macro: Developer → Record Macro

## Basic Structure

Sub MacroName()  
 ' This is a comment  
 Dim variableName As DataType  
  
 ' Your code here  
  
End Sub

## Variable Types

|  |  |  |
| --- | --- | --- |
| **Type** | **Use For** | **Declaration** |
| String | Text | Dim name As String |
| Integer | Whole numbers (-32,768 to 32,767) | Dim count As Integer |
| Long | Large whole numbers | Dim bigNum As Long |
| Double | Decimal numbers | Dim price As Double |
| Boolean | True/False | Dim isValid As Boolean |
| Date | Dates | Dim startDate As Date |
| Variant | Any type (flexible but slower) | Dim anything As Variant |
| Range | Cell range object | Dim rng As Range |
| Worksheet | Worksheet object | Dim ws As Worksheet |
| Workbook | Workbook object | Dim wb As Workbook |

## Common Operations

**Working with Cells:**

' Read a cell value  
value = Range("A1").Value  
value = Cells(1, 1).Value ' Row 1, Column 1  
  
' Write to a cell  
Range("A1").Value = "Hello"  
Cells(1, 1).Value = 100  
  
' Select a range  
Range("A1:B10").Select  
  
' Copy and Paste  
Range("A1:A10").Copy Destination:=Range("C1")  
  
' Clear contents  
Range("A1:A10").ClearContents

**Loops:**

' For Loop  
For i = 1 To 10  
 Cells(i, 1).Value = i \* 2  
Next i  
  
' For Each (loop through range)  
Dim cell As Range  
For Each cell In Range("A1:A10")  
 If cell.Value > 100 Then  
 cell.Interior.Color = vbYellow  
 End If  
Next cell  
  
' Do While Loop  
i = 1  
Do While Cells(i, 1).Value <> ""  
 ' Process row i  
 i = i + 1  
Loop

**Conditionals:**

' If Statement  
If score >= 90 Then  
 grade = "A"  
ElseIf score >= 80 Then  
 grade = "B"  
Else  
 grade = "C"  
End If  
  
' Select Case  
Select Case department  
 Case "Sales"  
 bonus = salary \* 0.1  
 Case "Marketing"  
 bonus = salary \* 0.08  
 Case Else  
 bonus = salary \* 0.05  
End Select

**Working with Worksheets:**

' Reference worksheet  
Set ws = ThisWorkbook.Worksheets("Sheet1")  
  
' Add new worksheet  
Worksheets.Add.Name = "NewSheet"  
  
' Loop through all worksheets  
Dim ws As Worksheet  
For Each ws In ThisWorkbook.Worksheets  
 Debug.Print ws.Name  
Next ws  
  
' Find last row with data  
lastRow = ws.Cells(ws.Rows.Count, "A").End(xlUp).Row  
  
' Find last column with data  
lastCol = ws.Cells(1, ws.Columns.Count).End(xlToLeft).Column

## Practical VBA Examples for Interviews

**Example 1: Highlight Duplicates**

Sub HighlightDuplicates()  
 Dim rng As Range  
 Dim cell As Range  
 Dim dict As Object  
  
 Set dict = CreateObject("Scripting.Dictionary")  
 Set rng = Range("A1:A" & Cells(Rows.Count, 1).End(xlUp).Row)  
  
 For Each cell In rng  
 If dict.exists(cell.Value) Then  
 cell.Interior.Color = vbYellow  
 dict(cell.Value).Interior.Color = vbYellow  
 Else  
 dict.Add cell.Value, cell  
 End If  
 Next cell  
End Sub

**Example 2: Create Summary Report**

Sub CreateSummary()  
 Dim wsData As Worksheet, wsSummary As Worksheet  
 Dim lastRow As Long, i As Long  
 Dim category As String  
 Dim dict As Object  
  
 Set wsData = Worksheets("Data")  
 Set dict = CreateObject("Scripting.Dictionary")  
  
 lastRow = wsData.Cells(Rows.Count, 1).End(xlUp).Row  
  
 ' Sum by category  
 For i = 2 To lastRow  
 category = wsData.Cells(i, 1).Value  
 If dict.exists(category) Then  
 dict(category) = dict(category) + wsData.Cells(i, 2).Value  
 Else  
 dict.Add category, wsData.Cells(i, 2).Value  
 End If  
 Next i  
  
 ' Create summary sheet  
 On Error Resume Next  
 Application.DisplayAlerts = False  
 Worksheets("Summary").Delete  
 Application.DisplayAlerts = True  
 On Error GoTo 0  
  
 Set wsSummary = Worksheets.Add  
 wsSummary.Name = "Summary"  
  
 ' Output results  
 wsSummary.Range("A1").Value = "Category"  
 wsSummary.Range("B1").Value = "Total"  
  
 i = 2  
 For Each key In dict.Keys  
 wsSummary.Cells(i, 1).Value = key  
 wsSummary.Cells(i, 2).Value = dict(key)  
 i = i + 1  
 Next key  
End Sub

**Example 3: Export Each Sheet to CSV**

Sub ExportSheetsToCSV()  
 Dim ws As Worksheet  
 Dim savePath As String  
  
 savePath = ThisWorkbook.Path & "\"  
  
 For Each ws In ThisWorkbook.Worksheets  
 ws.Copy  
 ActiveWorkbook.SaveAs savePath & ws.Name & ".csv", xlCSV  
 ActiveWorkbook.Close False  
 Next ws  
  
 MsgBox "Export complete!"  
End Sub

**Example 4: Autofilter and Copy Results**

Sub FilterAndCopy()  
 Dim wsSource As Worksheet, wsDest As Worksheet  
 Dim lastRow As Long  
  
 Set wsSource = Worksheets("Data")  
 Set wsDest = Worksheets("Filtered")  
  
 ' Clear destination  
 wsDest.Cells.Clear  
  
 lastRow = wsSource.Cells(Rows.Count, 1).End(xlUp).Row  
  
 ' Apply filter  
 wsSource.Range("A1:D" & lastRow).AutoFilter Field:=2, Criteria1:=">1000"  
  
 ' Copy visible cells  
 wsSource.Range("A1:D" & lastRow).SpecialCells(xlCellTypeVisible).Copy \_  
 Destination:=wsDest.Range("A1")  
  
 ' Remove filter  
 wsSource.AutoFilterMode = False  
End Sub

# 10. Common Interview Scenarios

## Scenario 1: Clean and Transform Data

**Task:** You have a column with names in "LastName, FirstName" format. Split into two columns and clean extra spaces.

**Solution:**

First Name: =TRIM(MID(A1, FIND(",", A1)+1, 100))  
Last Name: =TRIM(LEFT(A1, FIND(",", A1)-1))

## Scenario 2: Calculate Running Total

**Task:** Create a running total of sales by date.

**Solution:**

=SUMIF($A$2:A2, "<="&A2, $B$2:B2)  
Or: =SUM($B$2:B2)

## Scenario 3: Find Duplicates

**Task:** Flag duplicate entries in column A.

**Solution:**

=IF(COUNTIF($A$2:A2, A2)>1, "Duplicate", "Unique")  
Or: =IF(COUNTIF(A:A, A2)>1, "Duplicate", "Unique")

## Scenario 4: Rank Within Groups

**Task:** Rank salespeople within each region by sales amount.

**Solution:**

=SUMPRODUCT((A2=$A$2:$A$100)\*(C2<$C$2:$C$100))+1  
Where A=Region, C=Sales

## Scenario 5: Dynamic Data Validation

**Task:** Create dropdown that shows products based on selected category.

**Solution:**

1. Create named ranges for each category  
2. Data Validation → List → =INDIRECT(CategoryCell)

## Scenario 6: Conditional Summing Across Sheets

**Task:** Sum a cell across multiple sheets.

**Solution:**

=SUM(Sheet1:Sheet12!A1)  
Or: =SUMPRODUCT(SUMIF(INDIRECT("'"&{"Sheet1","Sheet2","Sheet3"}&"'!A:A"),"Criteria",INDIRECT("'"&{"Sheet1","Sheet2","Sheet3"}&"'!B:B")))

## Scenario 7: Year-over-Year Comparison

**Task:** Calculate YoY growth percentage.

**Solution:**

YoY Growth: =(CurrentYear - PriorYear) / PriorYear  
=IFERROR((B2-C2)/C2, 0)

## Scenario 8: Find Nth Occurrence

**Task:** Find the 3rd occurrence of "Error" in a column.

**Solution:**

=SMALL(IF(A:A="Error", ROW(A:A)), 3)  
(Enter with Ctrl+Shift+Enter if not Excel 365)

# Interview Tips Summary

## DO:

* Think out loud - explain your approach
* Ask clarifying questions about the data
* Start simple, then optimize
* Use keyboard shortcuts (Ctrl+Shift+End, Ctrl+Down)
* Know INDEX/MATCH over VLOOKUP
* Demonstrate pivot table skills
* Show you can handle errors (IFERROR)
* Mention performance (avoid volatile functions on large data)

## DON'T:

* Don't start coding without understanding the problem
* Don't hardcode values - use cell references
* Don't forget to handle edge cases (blanks, errors)
* Don't use entire column (A:A) in SUMIFS on large data
* Don't panic if you forget exact syntax - explain the concept

## Essential Keyboard Shortcuts

|  |  |
| --- | --- |
| **Shortcut** | **Action** |
| Ctrl+Shift+End | Select to last used cell |
| Ctrl+Down/Up/Left/Right | Jump to edge of data |
| Ctrl+Shift+L | Toggle AutoFilter |
| Alt+= | AutoSum |
| Ctrl+; | Insert current date |
| Ctrl+Shift+; | Insert current time |
| F4 | Toggle absolute reference ($) |
| Ctrl+` | Show/hide formulas |
| Alt+Enter | New line within cell |
| Ctrl+D | Fill down |
| Ctrl+R | Fill right |