

# ACC575: Data Analytics for Accounting

## Excel Functions

Jaeyoon Yu, Ph.D.  
Central Michigan University

September 28, 2025

# Table of Contents

---

## 1 Introduction

- Function Dialogue Box
- formulatext()
- filter()

## 2 Logical Functions

- and()
- or()
- not()
- if()
- ifs()
- iferror()

## 3 Math Functions

- count(), counta()
- sum(), average(), median(), min(), max()

## 4 Conditional Calculations (e.g., countif(); sumif()); averageif())

## 5 Database Functions

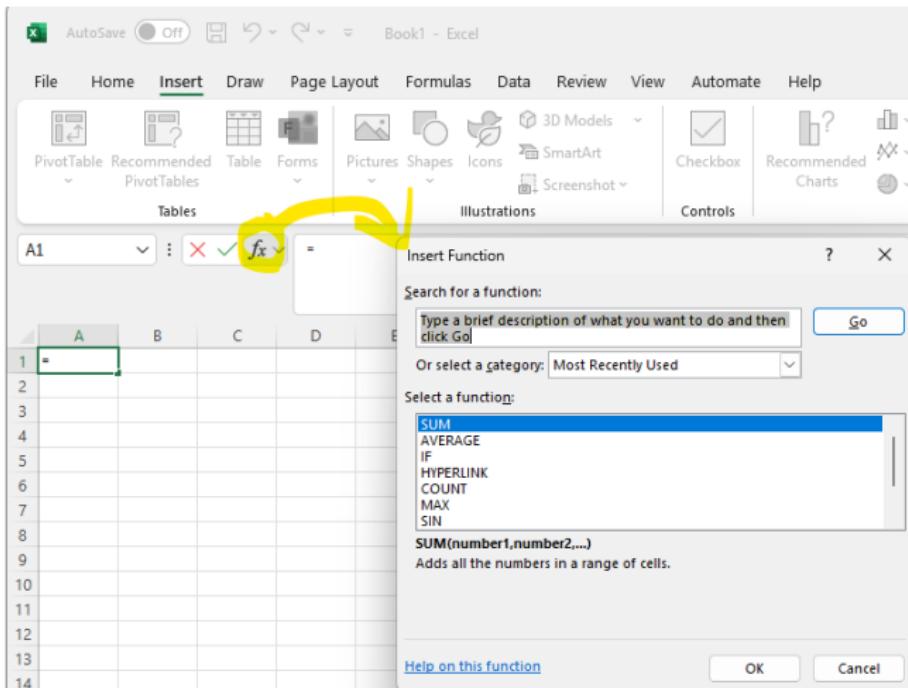
## 6 Lookup Functions

- vlookup()
- xlookup()
- index()
- match()

- Functions are built-in formulas in Excel that perform specific calculations or operations.
- Functions are used to perform calculations, analyze data, and manipulate text and numbers.
- Functions are essential for financial analysis, data analysis, and reporting.

## Function Dialogue Box

- The **Function Dialogue Box** is a tool that helps you create formulas in Excel.
- It is a tool that helps you create formulas in Excel.



- Search xlookup() function and click Go.
- Choose the function and click OK.
- You can see the function dialogue box including:
  - ▶ Name
  - ▶ Arguments (input parameters)
  - ▶ Description

The screenshot shows the Microsoft Excel interface with the ribbon at the top. The 'Insert' tab is selected. Below the ribbon, the formula bar displays the formula `=XLOOKUP()`. A 'Function Arguments' dialog box is open, centered over the formula bar. The dialog box is titled 'XLOOKUP' and contains five input fields: 'Lookup\_value', 'Lookup\_array', 'Return\_array', 'If\_not\_found', and 'Match\_mode'. Each field has a small icon to its right and a description to its right of the icon. Below the dialog box, the Excel worksheet area is visible, showing rows 1 through 13 and columns A through E. Row 1 contains the formula `=P()`. The description in the dialog box reads: 'Searches a range or an array for a match and returns the corresponding item from a second range or array. By default, an exact match is used.' Below this description, it says 'Lookup\_value is the value to search for.' At the bottom of the dialog box, there is a 'Formula result =' label followed by an equals sign (=) and a 'Help on this function' link. At the very bottom right of the dialog box are 'OK' and 'Cancel' buttons.

# FORMULATEXT()

## Description

- FORMULATEXT() is a lookup function that returns the formula from a referenced cell as text.
- It displays the actual formula syntax rather than the calculated result, useful for documentation and auditing.

## Syntax

=FORMULATEXT(reference)

- **reference:** The cell reference containing the formula (e.g., A1)
- Returns the formula as text, or #N/A if the cell doesn't contain a formula
- Useful for documenting and auditing spreadsheet formulas

## FORMULATEXT() Example: Simple Data Analysis

**Description:** Uses FORMULATEXT() to display the formulas used for calculating sum and mean of simple data values.

	A	B	C	D	E	F
1	<u>Values</u>			<u>Statistic</u>	<u>Formula</u>	<u>FORMULATEXT Output</u>
2	10			Sum	=SUM(\$A\$2:\$A\$6)	=SUM(\$A\$2:\$A\$6)
3	20			Mean	=AVERAGE(\$A\$2:\$A\$6)	=AVERAGE(\$A\$2:\$A\$6)
4	30					
5	40					
6	50					

### Formulas:

- $=\text{SUM}(\text{$A}\$2:\text{$A}\$6)$  in cell E2 (result: 150)
- $=\text{AVERAGE}(\text{$A}\$2:\text{$A}\$6)$  in cell E3 (result: 30)
- $=\text{FORMULATEXT}(E2)$  in cell F2 (shows:  $=\text{SUM}(\text{$A}\$2:\text{$A}\$6)$ )
- $=\text{FORMULATEXT}(E3)$  in cell F3 (shows:  $=\text{AVERAGE}(\text{$A}\$2:\text{$A}\$6)$ )

# FILTER()

## Description

- FILTER() is a dynamic array function that filters a range of data based on specified criteria.
- It returns an array of values that meet the given conditions, automatically resizing based on results.
- It is a very useful function for filtering data based on specified criteria.

## Syntax

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

- **array:** The range of data to filter (e.g., A1:A10)
- **include:** The condition or criteria for filtering (e.g., B1:B10>100)
- **if\_empty:** Value to return if no results match (optional)
- Returns a dynamic array of filtered results

## FILTER() Example 1: High-Value Assets

**Description:** Filters a list of assets to show only those with values greater than \$50,000.

	A	B
1	<u>Asset Name</u>	<u>Value</u>
2	Equipment A	75,000
3	Vehicle B	35,000
4	Building C	120,000
5	Computer D	2,500

**Formula in cell D2:** =FILTER(\$A\$2:\$A\$5, \$B\$2:\$B\$5>50000)

**Filtered Results:**

	D
1	<u>High-Value Assets</u>
2	Equipment A
3	Building C

## FILTER() Example 2: Department Filter

**Description:** Filters employee data to show only employees from the "Finance" department.

	A	B	C
1	<u>Name</u>	<u>Department</u>	<u>Salary</u>
2	John Smith	Finance	65,000
3	Jane Doe	Marketing	55,000
4	Bob Johnson	Finance	70,000
5	Alice Brown	HR	60,000

**Formula in cell E2:** =FILTER(\$A\$2:\$C\$5, \$B\$2:\$B\$5="Finance")

**Filtered Results:**

	E	F	G
1	<u>Name</u>	<u>Department</u>	<u>Salary</u>
2	John Smith	Finance	65,000
3	Bob Johnson	Finance	70,000

# Table of Contents

---

## 1 Introduction

- Function Dialogue Box
- formulatext()
- filter()

## 2 Logical Functions

- and()
- or()
- not()
- if()
- ifs()
- iferror()

## 3 Math Functions

- count(), counta()
- sum(), average(), median(), min(), max()

## 4 Conditional Calculations (e.g., countif(); sumif()); averageif())

## 5 Database Functions

## 6 Lookup Functions

- vlookup()
- xlookup()
- index()
- match()

# AND()

## Description

- AND() is a logical function that returns TRUE if all conditions are TRUE, and FALSE if any condition is FALSE.
- It allows you to test multiple conditions simultaneously in your spreadsheet.

## Syntax

=AND(logical1, [logical2], ...)

- **logical1:** The first condition to test (e.g., A1>100)
- **logical2:** The second condition to test (optional)
- *...continue for additional conditions as needed*
- Returns TRUE only if ALL conditions are TRUE

## AND() Example 1: Loan Qualification

**Description:** Checks if applicant qualifies for loan based on income > \$50,000, credit score > 600, and debt-to-income ratio < 40%.

	A	B	C	D
1	<u>Income</u>	<u>Credit Score</u>	<u>Debt Ratio</u>	<u>Qualified</u>
2	60,000	650	0.35	
3	45,000	700	0.30	
4	55,000	550	0.25	

**Formula in cell D2:** =AND(A2>50000, B2>600, C2<0.4)

## AND() Example 2: Investment Criteria

**Description:** Evaluates if investment meets criteria: market cap > \$100,000, ROE > 15%, and debt ratio < 5%.

	A	B	C	D
1	<u>Market Cap</u>	<u>ROE</u>	<u>Debt Ratio</u>	<u>Meets Criteria</u>
2	150,000	0.18	0.03	
3	80,000	0.20	0.02	
4	120,000	0.12	0.04	

**Formula in cell D2:** =AND(A2>100000, B2>0.15, C2<0.05)

# OR()

## Description

- OR() is a logical function that returns TRUE if any condition is TRUE, and FALSE only if all conditions are FALSE.
- It allows you to test multiple conditions where only one needs to be met.

## Syntax

=OR(logical1, [logical2], ...)

- **logical1:** The first condition to test (e.g., A1>100)
- **logical2:** The second condition to test (optional)
- *...continue for additional conditions as needed*
- Returns TRUE if ANY condition is TRUE

## OR() Example 1: Risk Assessment

**Description:** Flags high-risk investments if any condition is met: volatility > 10%, debt ratio > 30%, or risk rating = "High".

	A	B	C	D
1	<u>Volatility</u>	<u>Debt Ratio</u>	<u>Risk Rating</u>	<u>High Risk</u>
2	0.08	0.25	Medium	
3	0.15	0.35	Low	
4	0.12	0.20	High	
5	0.18	0.15	Low	

**Formula in cell D2:** =OR(A2>0.1, B2>0.3, C2="High")

## OR() Example 2: Accounting - Flagging Unusual Transactions

**Description:** Flags transactions for review if any of the following are true: amount exceeds \$10,000, transaction type is International, or entry method is Manual Entry.

	A	B	C	D
1	<u>Amount</u>	<u>Type</u>	<u>Entry Method</u>	<u>Review Needed</u>
2	12,500	Domestic	Import	
3	8,000	International	Manual Entry	
4	5,000	Domestic	Manual Entry	
5	7,500	Domestic	Import	

**Formula in cell D2:** =OR(A2>10000, B2="International", C2="Manual Entry")

# NOT()

## Description

- NOT() is a logical function that reverses the logical value of its argument.
- It returns TRUE if the condition is FALSE, and FALSE if the condition is TRUE.

## Syntax

=NOT(logical)

- **logical:** The condition you want to reverse (e.g., A1>100)
- Returns the opposite logical value
- TRUE becomes FALSE, FALSE becomes TRUE

## NOT() Example 1: Exclusion Criteria

**Description:** Identifies accounts that are NOT excluded from analysis.

	A	B	C
1	<u>Account Status</u>	<u>Formula</u>	<u>Include in Analysis</u>
2	Active	=NOT(A2="Excluded")	
3	Excluded	=NOT(A3="Excluded")	
4	Pending	=NOT(A4="Excluded")	

**Formula in cell C2:** =NOT(A2="Excluded")

## NOT() Example 2: Budget Overrun Check

**Description:** Checks if actual expenses are NOT over budget (i.e., within budget limits).

	A	B	C
1	<u>Actual</u>	<u>Budget</u>	<u>Within Budget</u>
2	8,000	10,000	
3	12,000	10,000	
4	9,500	10,000	

**Formula in cell C2:** =NOT(A2>B2)

## Description

- IF() is a logical function that returns one value if a condition is TRUE and another value if it's FALSE.
- It allows you to make decisions in your spreadsheet based on certain criteria.

## Syntax

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

- **logical\_test:** The condition you want to test (e.g., A1>100)
- **value\_if\_true:** The value returned if the condition is TRUE
- **value\_if\_false:** The value returned if the condition is FALSE

## IF() Example 1: Asset Classification

**Description:** Classifies assets above \$50,000 as "Major Asset".

	A	B
1	<u>Asset Value</u>	<u>Classification</u>
2	75,000	
3	40,000	

**Formula in cell C2:** =IF(A2>50000, "Major", "Minor")

## IF() Example 2: Profitability Check

**Description:** Determines if net income shows profit or loss.

	A	B
1	<u>Net Income</u>	<u>Result</u>
2	5,000	
3	-2,000	

**Formula in cell C2:** =IF(A2>0, "Profitable", "Loss")

## IF() Example 3: Nested IF for Credit Rating

**Description:** Assigns credit ratings based on net income levels.  
A: >100, B: 50–100, C: <50

	A	B
1	<u>Net Income</u>	<u>Rating</u>
2	150	
3	75	
4	30	

**Formula in cell C2:** =IF(A2>100, "A", IF(A2>50, "B", "C"))

## Description

- IFS() is a logical function that checks multiple conditions and returns a value corresponding to the first TRUE condition.
- It simplifies complex nested IF statements, making formulas easier to read and maintain.

## Syntax

=IFS(logical\_test1, value\_if\_true1, [logical\_test2, value\_if\_true2], ...)

- **logical\_test1:** The first condition to test (e.g., A1>100)
- **value\_if\_true1:** The value returned if logical\_test1 is TRUE
- **logical\_test2:** The second condition to test (optional)
- **value\_if\_true2:** The value returned if logical\_test2 is TRUE
- **...continue for additional conditions as needed**

## IFS() Example 1: Nested Credit Rating

**Description:** Assigns credit ratings based on net income levels.  
A: >100, B: 50–100, C: <50

	A	B
1	<u>Net Income</u>	<u>Rating</u>
2	150	
3	75	
4	30	

**Formula in cell C2:** =IFS(A2>100, "A", A2>50, "B", TRUE, "C")

## IFS() Example 2: Student Grade Assignment

**Description:** Assigns letter grades based on student scores using multiple conditions.

<u>Score</u>	<u>Grade</u>
95	
82	
76	
61	
54	

**Formula in cell C2:** =IFS(A2>=90, "A", A2>=80, "B", A2>=70, "C", A2>=60, "D", TRUE, "F")

# IFERROR()

## Description

- IFERROR() is a logical function that returns a specified value if a formula evaluates to an error, otherwise returns the result of the formula.
- It helps prevent error messages from appearing in your spreadsheet and provides user-friendly alternatives.

## Syntax

```
=IFERROR(value, value_if_error)
```

- **value:** The formula or expression to evaluate
- **value\_if\_error:** The value to return if an error occurs
- Returns the formula result if no error, otherwise returns the specified value

## IFERROR() Example 1: Division by Zero

---

**Description:** Calculates return on assets (ROA) and returns "-" if assets are zero to avoid calculation errors.

	A	B	C
1	<u>Net Income</u>	<u>Total Assets</u>	<u>ROA (A/B)</u>
2	10,000	100,000	0.10
3	5,000	0	-
4	8,000	160,000	0.05

**Formula in cell C2:** =IFERROR(A2/B2, "-")

# Table of Contents

---

## 1 Introduction

- Function Dialogue Box
- formulatext()
- filter()

## 2 Logical Functions

- and()
- or()
- not()
- if()
- ifs()
- iferror()

## 3 Math Functions

- count(), counta()
- sum(), average(), median(), min(), max()

## 4 Conditional Calculations (e.g., countif(); sumif()); averageif())

## 5 Database Functions

## 6 Lookup Functions

- vlookup()
- xlookup()
- index()
- match()

- COUNT(): counts the number of cells containing numbers
- COUNTA(): counts the number of non-empty cells
- SUM(): adds all numbers in a range
- AVERAGE(): calculates the arithmetic mean of numbers
- MEDIAN(): finds the middle value in a range
- MIN(): finds the smallest value in a range
- MAX(): finds the largest value in a range

# COUNT()

## Description

- COUNT() is a statistical function that counts the number of cells containing numbers in a range.
- It only counts cells with numeric values and ignores text, empty cells, or error values.

## Syntax

=COUNT(value1, [value2], ...)

- **value1:** The first value or range to count (e.g., A1:A10)
- **value2:** Additional values or ranges to count (optional)
- Returns the number of cells containing numbers

# COUNTA()

## Description

- COUNTA() is a statistical function that counts the number of non-empty cells in a range.
- It counts cells containing any type of data including numbers, text, dates, and formulas.

## Syntax

=COUNTA(value1, [value2], ...)

- **value1:** The first value or range to count (e.g., A1:A10)
- **value2:** Additional values or ranges to count (optional)
- Returns the number of non-empty cells

## COUNT() and COUNTA() Example: Employee Data Analysis

**Description:** Demonstrates the difference between COUNT() and COUNTA() using a small employee dataset with some missing salary information. Only last names are shown.

	A	B	C	D	E
1	Last Name	Salary		Function	Result
2	Smith	75000		COUNT(\$A\$2:\$A\$5)	0
3	Doe	82000		COUNT(\$B\$2:\$B\$5)	3
4	Johnson			COUNTA(\$A\$2:\$A\$5)	4
5	Brown	68000		COUNTA(\$B\$2:\$B\$5)	3

### Formulas:

- COUNT(\$A\$2:\$A\$5) returns 0 (no numeric values in last names)
- COUNT(\$B\$2:\$B\$5) returns 3 (only numeric salary values)
- COUNTA(\$A\$2:\$A\$5) returns 4 (all employee last names)
- COUNTA(\$B\$2:\$B\$5) returns 3 (all non-empty salary cells)

# SUM()

## Description

- SUM() is a mathematical function that adds all numbers in a range of cells.
- It calculates the total of all numeric values in the specified range.

## Syntax

=SUM(number1, [number2], ...)

- **number1:** The first number or range to add (e.g., A1:A10)
- **number2:** Additional numbers or ranges to add (optional)
- Returns the sum of all specified numbers

# AVERAGE()

## Description

- AVERAGE() is a mathematical function that calculates the arithmetic mean of numbers in a range.
- It finds the sum of all values divided by the count of values.

## Syntax

=AVERAGE(number1, [number2], ...)

- **number1:** The first number or range to average (e.g., A1:A10)
- **number2:** Additional numbers or ranges to average (optional)
- Returns the arithmetic mean of all specified numbers

# MEDIAN()

## Description

- MEDIAN() is a mathematical function that finds the middle value in a set of numbers.
- It returns the value that separates the higher half from the lower half of the data.

## Syntax

=MEDIAN(number1, [number2], ...)

- **number1:** The first number or range to find median (e.g., A1:A10)
- **number2:** Additional numbers or ranges to include (optional)
- Returns the middle value of all specified numbers

## Math Functions Example: Summary Statistics

**Description:** Calculates summary statistics for 2024 fiscal year assets of major companies using mathematical functions.

	A	B	C	D	E
1	<u>FirmName</u>	<u>Assets (Billions)</u>		<u>Statistic</u>	<u>Value</u>
2	Apple Inc.	352.58		Total	1760.55
3	Microsoft Corp.	411.98		Average	352.11
4	Amazon.com Inc.	527.85		Median	402.39
5	Alphabet Inc.	402.39		Maximum	527.85
6	Tesla Inc.	65.75		Minimum	65.75

### Formulas:

- =SUM(\$B\$2:\$B\$6) for total assets
- =AVERAGE(\$B\$2:\$B\$6) for mean assets
- =MEDIAN(\$B\$2:\$B\$6) for median assets
- =MAX(\$B\$2:\$B\$6) for maximum assets
- =MIN(\$B\$2:\$B\$6) for minimum assets

# Table of Contents

---

## 1 Introduction

- Function Dialogue Box
- formulatext()
- filter()

## 2 Logical Functions

- and()
- or()
- not()
- if()
- ifs()
- iferror()

## 3 Math Functions

- count(), counta()
- sum(), average(), median(), min(), max()

## 4 Conditional Calculations (e.g., countif(); sumif()); averageif())

## 5 Database Functions

## 6 Lookup Functions

- vlookup()
- xlookup()
- index()
- match()

- COUNTIF(): counts the number of cells in a range that meet specified criteria
- SUMIF(): adds all numbers in a range that meet specified criteria
- AVERAGEIF(): calculates the arithmetic mean of numbers in a range that meet specified criteria
- MEDIANIF(): finds the middle value in a range that meet specified criteria
- MINIF(): finds the smallest value in a range that meet specified criteria
- MAXIF(): finds the largest value in a range that meet specified criteria

Note: There are also COUNTIFS(), SUMIFS(), AVERAGEIFS(), MEDIANIFS(), MINIFS(), and MAXIFS() for multiple criteria (beyond the scope of this course).

# COUNTIF()

## Description

- COUNTIF() is a conditional function that counts the number of cells in a range that meet specified criteria.
- It provides a count of cells that satisfy the given condition.

## Syntax

=COUNTIF(range, criteria)

- **range:** The range of cells to evaluate against the criteria
- **criteria:** The condition that determines which cells to count
- Returns the count of cells that meet the specified criteria

# SUMIF()

## Description

- SUMIF() is a conditional function that sums values in a range based on specified criteria.
- It allows you to calculate totals for specific conditions without manually filtering data.

## Syntax

=SUMIF(range, criteria, [sum\_range])

- **range:** The range of cells to evaluate against the criteria
- **criteria:** The condition that determines which cells to sum
- **sum\_range:** The range of cells to sum (optional, defaults to range)
- Returns the sum of cells that meet the specified criteria

# AVERAGEIF()

## Description

- AVERAGEIF() is a conditional function that calculates the average of values in a range based on specified criteria.
- It finds the arithmetic mean of cells that meet the specified condition.

## Syntax

```
=AVERAGEIF(range, criteria, [average_range])
```

- **range:** The range of cells to evaluate against the criteria
- **criteria:** The condition that determines which cells to average
- **average\_range:** The range of cells to average (optional, defaults to range)
- Returns the average of cells that meet the specified criteria

## Example: Large Company Analysis

**Description:** Calculates conditional statistics for companies with total assets  $\geq 100$  billion using 2024 fiscal year data.

	A	B	C	D	E
1	<u>Company</u>	<u>Assets (Billions)</u>			
2	Apple Inc.	352.58			
3	Microsoft Corp.	411.98		<u>Statistic</u>	<u>Value</u>
4	Amazon.com Inc.	527.85		<u>Count (<math>\geq 100</math>)</u>	4
5	Alphabet Inc.	402.39		<u>Average (<math>\geq 100</math>)</u>	423.70
6	Tesla Inc.	65.75		<u>Sum (<math>\geq 100</math>)</u>	1694.80

### Formulas:

- `=SUMIF($B$2:$B$6, ">=100", $B$2:$B$6)` for sum of assets  $\geq 100$  billion
- `=AVERAGEIF($B$2:$B$6, ">=100", $B$2:$B$6)` for average of assets  $\geq 100$  billion
- `=COUNTIF($B$2:$B$6, ">=100")` for count of companies  $\geq 100$  billion

## Example: Large Company Analysis with Adjustable Cutoff

**Description:** Calculates conditional statistics for companies with total assets greater than or equal to the user-defined cutoff (cell D1) using 2024 fiscal year data. The cutoff value for assets (in billions) can be set by the user in cell D1. All statistics below reference this cell.

	A	B	C	D	E
1	<u>Company</u>	<u>Assets (Billions)</u>		<u>Cutoff</u>	100
2	Apple Inc.	352.58			
3	Microsoft Corp.	411.98		<u>Statistic</u>	<u>Value</u>
4	Amazon.com Inc.	527.85		<u>Count (<math>\geq</math>D1)</u>	4
5	Alphabet Inc.	402.39		<u>Average (<math>\geq</math>D1)</u>	423.70
6	Tesla Inc.	65.75		<u>Sum (<math>\geq</math>D1)</u>	1694.80

### Formulas:

- =SUMIF(\$B\$2:\$B\$6, ">=" & \$D\$1, \$B\$2:\$B\$6) for sum of assets  $\geq$  cutoff
- =AVERAGEIF(\$B\$2:\$B\$6, ">=" & \$D\$1, \$B\$2:\$B\$6) for average of assets  $\geq$  cutoff
- =COUNTIF(\$B\$2:\$B\$6, ">=" & \$D\$1) for count of companies  $\geq$  cutoff

*Note: Change the value in cell D1 to update the cutoff for all statistics.*

# Table of Contents

---

## 1 Introduction

- Function Dialogue Box
- formulatext()
- filter()

## 2 Logical Functions

- and()
- or()
- not()
- if()
- ifs()
- iferror()

## 3 Math Functions

- count(), counta()
- sum(), average(), median(), min(), max()

## 4 Conditional Calculations (e.g., countif(); sumif()); averageif())

## 5 Database Functions

## 6 Lookup Functions

- vlookup()
- xlookup()
- index()
- match()

- DCOUNT(): counts the number of cells in a range that meet specified criteria
- DSUM(): adds all numbers in a range that meet specified criteria
- DAVERAGE(): calculates the arithmetic mean of numbers in a range that meet specified criteria
- DMAX(): finds the largest value in a range that meet specified criteria
- DMIN(): finds the smallest value in a range that meet specified criteria

## What are Database Functions?

- Database functions perform calculations on data that meets specific criteria.
- They work with structured data organized in rows and columns with headers.
- Unlike regular functions
  - ▶ They can apply multiple conditions (AND or OR) simultaneously.
  - ▶ They can work with multiple columns at once.

## Key Components

- **Database:** The range containing your data (including headers)
- **Field:** The column you want to calculate (count, sum, average, etc.)
- **Criteria:** The conditions that determine which rows to include

# DCOUNT()

## Description

- DCOUNT() is a database function that counts the number of cells containing numbers in a database column that meet specified criteria.
- It counts numeric values only and ignores text, empty cells, or error values in the filtered results.

## Syntax

=DCOUNT(database, field, criteria)

- **database:** The range containing the database (including headers)
- **field:** The column to count (column label or number)
- **criteria:** The range containing conditions for filtering
- Returns the count of numeric values that meet the criteria

# DSUM()

## Description

- DSUM() is a database function that calculates the sum of values in a database column that meet specified criteria.
- It allows you to sum specific records based on multiple conditions without manually filtering data.

## Syntax

=DSUM(database, field, criteria)

- **database:** The range containing the database (including headers)
- **field:** The column to sum (column label or number)
- **criteria:** The range containing conditions for filtering
- Returns the sum of values that meet the criteria

# DAVERAGE()

## Description

- DAVERAGE() is a database function that calculates the average of values in a database column that meet specified criteria.
- It allows you to average specific records based on multiple conditions without manually filtering data.

## Syntax

=DAVERAGE(database, field, criteria)

- **database:** The range containing the database (including headers)
- **field:** The column to average (column label or number)
- **criteria:** The range containing conditions for filtering
- Returns the average of values that meet the criteria

## How to Read Criteria Ranges:

- **Same Row** = AND logic: Both conditions must be true
- **Different Rows** = OR logic: Either condition can be true
- **Empty Cells** = Ignored in the criteria

## How to Work with Criteria Ranges: AND logic

AND Logic - Same Row:

	A	B	C	D	E	F
1	Last Name	Department	Salary		Department	Salary
2	Smith	ACC	65,000		ACC	$\geq 70000$
3	Doe	ACC	70,000			

Example Above: Department = "ACC" AND Salary  $\geq 70,000$

- Row 2 has both Department and Salary conditions
- Both conditions must be met simultaneously (AND logic)
- Only records meeting BOTH criteria are included

## How to Work with Criteria Ranges: OR logic

### OR Logic - Different Rows:

	A	B	C	D	E	F
1	Last Name	Department	Salary		Department	Salary
2	Smith	ACC	65,000		Fin	
3	Doe	ACC	70,000			>71000
4	Johnson	ACC	72,000			

**Example Above:** Department = "Fin" OR Salary > 71,000

- Row 2: Department = "Fin" (Salary condition is empty)
- Row 3: Salary > 71,000 (Department condition is empty)
- Records meeting EITHER condition are included (OR logic)
- A record needs to satisfy only ONE of the criteria

## Example 1: Single Condition

**Description:** Calculate the count, sum, and average salary of employees in the Accounting (ACC) department using database functions. The criteria is set in column E.

	A	B	C	D	E	F
1	Last Name	Department	Salary		Department	
2	Smith	ACC	65,000		ACC	
3	Doe	ACC	70,000			
4	Johnson	ACC	72,000			
5	Brown	ACC	60,000			
6	Wilson	FIN	75,000		Statistic	Value
7	Lee	FIN	68,000		Count	6
8	Taylor	MKT	55,000		Sum	410,000
9	Davis	MKT	58,000		Average	68,333

### Formulas:

- =DCOUNT(\$A\$1:\$C\$9, "Salary", \$E\$1:\$E\$2)
- =DSUM(\$A\$1:\$C\$9, "Salary", \$E\$1:\$E\$2)
- =DAVERAGE(\$A\$1:\$C\$9, "Salary", \$E\$1:\$E\$2)

## Example 2: OR Condition

**Description:** Calculate statistics for employees in either ACC or Fin departments using OR logic. Criteria are placed on separate rows.

	A	B	C	D	E	F
1	Last Name	Department	Salary		Department	
2	Smith	ACC	65,000		ACC	
3	Doe	ACC	70,000		FIN	
4	Johnson	ACC	72,000			
5	Brown	ACC	60,000		Statistic	Value
6	Wilson	FIN	75,000		Count	6
7	Lee	FIN	68,000		Sum	410,000
8	Taylor	MKT	55,000		Average	68,333
9	Davis	MKT	58,000			

### Formulas:

- =DCOUNT(\$A\$1:\$C\$9, "Salary", \$E\$1:\$E\$3)
- =DSUM(\$A\$1:\$C\$9, "Salary", \$E\$1:\$E\$3)
- =DAVERAGE(\$A\$1:\$C\$9, "Salary", \$E\$1:\$E\$3)

### Example 3: AND Condition

**Description:** Calculate statistics for ACC department employees with salary  $\geq \$70,000$  using AND logic with multiple criteria columns.

	A	B	C	D	E	F
1	Last Name	Department	Salary		Department	Salary
2	Smith	ACC	65,000		ACC	$\geq 70000$
3	Doe	ACC	70,000			
4	Johnson	ACC	72,000			
5	Brown	ACC	60,000		Statistic	Value
6	Wilson	FIN	75,000		Count	2
7	Lee	FIN	68,000		Sum	142,000
8	Taylor	MKT	55,000		Average	71,000
9	Davis	MKT	58,000			

#### Formulas:

- =DCOUNT(\$A\$1:\$C\$9, "Salary", \$E\$1:\$F\$2)
- =DSUM(\$A\$1:\$C\$9, "Salary", \$E\$1:\$F\$2)
- =DAVERAGE(\$A\$1:\$C\$9, "Salary", \$E\$1:\$F\$2)

## Example 4: Mixed OR Condition

**Description:** Calculate statistics for employees who are either in Fin department OR have salary > \$71,000 using OR logic with different criteria types.

	A	B	C	D	E	F
1	Last Name	Department	Salary		Department	Salary
2	Smith	ACC	65,000		FIN	
3	Doe	ACC	70,000			>71000
4	Johnson	ACC	72,000			
5	Brown	ACC	60,000		Statistic	Value
6	Wilson	FIN	75,000		Count	4
7	Lee	FIN	68,000		Sum	295,000
8	Taylor	MKT	55,000		Average	73,750
9	Davis	MKT	58,000			

### Formulas:

- =DCOUNT(\$A\$1:\$C\$9, "Salary", \$E\$1:\$F\$3)
- =DSUM(\$A\$1:\$C\$9, "Salary", \$E\$1:\$F\$3)
- =DAVERAGE(\$A\$1:\$C\$9, "Salary", \$E\$1:\$F\$3)

# Table of Contents

---

## 1 Introduction

- Function Dialogue Box
- formulatext()
- filter()

## 2 Logical Functions

- and()
- or()
- not()
- if()
- ifs()
- iferror()

## 3 Math Functions

- count(), counta()
- sum(), average(), median(), min(), max()

## 4 Conditional Calculations (e.g., countif(); sumif()); averageif())

## 5 Database Functions

## 6 Lookup Functions

- vlookup()
- xlookup()
- index()
- match()

# VLOOKUP()

## Description

- VLOOKUP() is a lookup function that searches for a value in the first column of a table and returns a value from the same row in a specified column.
- It is a very useful function for looking up values in a table.

## Syntax

```
=VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])
```

- **lookup\_value:** The value to search for (e.g., "Product A")
- **table\_array:** The table range containing the data (e.g., A1:D10)
- **col\_index\_num:** Column number to return (1st column = 1, 2nd = 2, etc.)
- **range\_lookup:** TRUE for approximate match, FALSE for exact match

## VLOOKUP() Example 1

---

**Description:** Looks up the price of "Product B" from the product table.

	A	B	C	D	E
1	<u>Product</u>	<u>Category</u>	<u>Price</u>		"Price of Product B:"
2	Product A	Electronics	25		40
3	Product B	Clothing	40		
4	Product C	Electronics	60		
5	Product D	Books	15		
6	Product E	Clothing	35		

**Formula in cell E1:**

- =VLOOKUP("Product B", \$A\$2:\$C\$6, 3, FALSE)

## VLOOKUP() Example 2

**Description:** Looks up the department for employee "John".

	A	B	C	D	E
1	Employee	Department	Salary		"Department of John:"
2	Alice	Finance	50000		IT
3	Bob	Marketing	45000		
4	John	IT	55000		
5	Mary	Finance	48000		
6	Tom	Marketing	42000		

**Formula in cell E1:**

- =VLOOKUP("John", \$A\$2:\$C\$6, 2, FALSE)

# XLOOKUP()

## Description

- XLOOKUP() searches for a value and returns a corresponding value from another range.
- It is a more powerful version of VLOOKUP().

## Syntax

```
=XLOOKUP(lookup_value, lookup_array, return_array, [if_not_found],  
[match_mode], [search_mode])
```

- **lookup\_value:** The value to search for (e.g., "Product B")
- **lookup\_array:** The range to search in (e.g., A2:A6)
- **return\_array:** The range to return values from (e.g., C2:C6)
- **if\_not\_found:** Value to return if no match (optional)
- **match\_mode:** 0 for exact match, -1 for exact or next smaller, 1 for exact or next larger
- **search\_mode:** 1 for first to last, -1 for last to first

## XLOOKUP() Example 1

---

**Description:** Looks up the price of "Product D" using XLOOKUP.

	A	B	C	D	E
1	<u>Product</u>	<u>Category</u>	<u>Price</u>		"Price of Product D:"
2	Product A	Electronics	25		15
3	Product B	Clothing	40		
4	Product C	Electronics	60		
5	Product D	Books	15		
6	Product E	Clothing	35		

**Formula in cell E1:**

- =XLOOKUP("Product D", \$A\$2:\$A\$6, \$C\$2:\$C\$6)

## XLOOKUP() Example 2

**Description:** Looks up the department for employee "Bob" with error handling.

	A	B	C	D	E
1	<u>Employee</u>	<u>Department</u>	<u>Salary</u>		"Department of Bob:"
2	Alice	Finance	50000		Marketing
3	Bob	Marketing	45000		
4	John	IT	55000		
5	Mary	Finance	48000		
6	Tom	Marketing	42000		

**Formula in cell E1:**

- =XLOOKUP("Bob", \$A\$2:\$A\$6, \$B\$2:\$B\$6, "Not Found")

# INDEX()

## Description

- INDEX() returns the value of a cell at the intersection of a specific row and column in a range.
- Provides flexible data retrieval by position, useful for dynamic references.

## Syntax

=INDEX(array, row\_num, [column\_num])

- **array:** The range of cells (e.g., A1:C10)
- **row\_num:** Row number within the array (1st row = 1, 2nd = 2, etc.)
- **column\_num:** Column number within the array (optional for single column)

## INDEX() Example 1

---

**Description:** Returns the value at row 3, column 2 of the range A2:C6.

	A	B	C	D	E
1	<u>Product</u>	<u>Category</u>	<u>Price</u>		"Row 3, Column 2:"
2	Product A	Electronics	25		Electronics
3	Product B	Clothing	40		
4	Product C	Electronics	60		
5	Product D	Books	15		
6	Product E	Clothing	35		

**Formula in cell E1:**

- =INDEX(\$A\$2:\$C\$6, 3, 2)

## INDEX() Example 2

---

**Description:** Returns the price (column 3) of the 4th product in the list.

	A	B	C	D	E
1	<u>Product</u>	<u>Category</u>	<u>Price</u>		"Row 4, Column 3:"
2	Product A	Electronics	25		15
3	Product B	Clothing	40		
4	Product C	Electronics	60		
5	Product D	Books	15		
6	Product E	Clothing	35		

**Formula in cell E1:**

- =INDEX(\$A\$2:\$C\$6, 4, 3)

# MATCH()

## Description

- MATCH() searches for a value in a range and returns its relative position.
- Provides the position number, not the actual value.

## Syntax

```
=MATCH(lookup_value, lookup_array, [match_type])
```

- **lookup\_value:** The value to search for (e.g., "Product B")
- **lookup\_array:** The range to search in (e.g., A1:A10)
- **match\_type:** 0 for exact match, 1 for ascending, -1 for descending

## MATCH() Example 1: Vertical Match

---

**Description:** Finds the position of "Product C" in the product list.

	A	B	C	D
1	<u>Product</u>	<u>Category</u>		"Position of Product C:"
2	Product A	Electronics		3
3	Product B	Clothing		
4	Product C	Electronics		
5	Product D	Books		
6	Product E	Clothing		

**Formula in cell E1:**

- `=MATCH("Product C", $A$2:$A$6, 0)`

## MATCH() Example 2: Horizontal Match

---

**Description:** Finds the position of the department "IT" in a horizontal list of department names.

	A	B	C	D	E	F
1	Accounting	Finance	Marketing	IT		Position of IT
2						4

**Formula in cell F2:**

- `=MATCH("IT", $A$1:$D$1, 0)`