# Excel Formulas Reference Guide

एक्सेल के सभी प्रमुख फ़ॉर्मूला और फंक्शन - विवरण एवं उदाहरण के साथ (Complete Formulas Manual)

Basic to Advanced | Hindi & English Guide | Quick Lookup

💡 **Tip:** Excel में किसी भी फ़ॉर्मूला को शुरू करने से पहले हमेशा = (बराबर का निशान) लगाएं।

## 1. MATH & ARITHMETIC FUNCTIONS (गणित के फ़ॉर्मूले)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **SUM** | `=SUM(number1, [number2], ...)` | दी गई सभी संख्याओं का जोड़ निकालता है। | `=SUM(A1:A10)` |
| **AVERAGE** | `=AVERAGE(number1, [number2], ...)` | दी गई संख्याओं का औसत (Average) निकालता है। | `=AVERAGE(B1:B20)` |
| **PRODUCT** | `=PRODUCT(number1, [number2], ...)` | सभी दी गई संख्याओं का गुणा (Multiply) करता है। | `=PRODUCT(A1:A5)` |
| **SUMIF** | `=SUMIF(range, criteria, [sum_range])` | किसी एक शर्त (Condition) के आधार पर जोड़ता है। | `=SUMIF(A1:A10, ">50")` |
| **SUMIFS** | `=SUMIFS(sum_range, criteria_range1, criteria1, ...)` | एक से अधिक शर्तों के आधार पर जोड़ता है। | `=SUMIFS(C1:C10, A1:A10, "Delhi", B1:B10, ">100")` |
| **ROUND** | `=ROUND(number, num_digits)` | संख्या को निश्चित दशमलव स्थानों तक राउंड करता है। | `=ROUND(12.3456, 2) → 12.35` |
| **ROUNDUP / DOWN** | `=ROUNDUP(number, num_digits)` | संख्या को ऊपर या नीचे की तरफ राउंड करता है। | `=ROUNDUP(12.31, 1) → 12.4` |
| **ABS** | `=ABS(number)` | संख्या का निरपेक्ष मान (Absolute Value) देता है (Negative को Positive)। | `=ABS(-50) → 50` |
| **MOD** | `=MOD(number, divisor)` | भाग देने के बाद शेषफल (Remainder) लौटाता है। | `=MOD(10, 3) → 1` |
| **POWER / SQRT** | `=SQRT(number) / =POWER(net, power)` | वर्गमूल (Square Root) या पावर घात निकालता है। | `=SQRT(64) → 8` |

## 2. LOGICAL FUNCTIONS (तार्किक फ़ॉर्मूले)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **IF** | `=IF(logical_test, value_if_true, value_if_false)` | शर्त सही होने पर एक वैल्यू और गलत होने पर दूसरी देता है। | `=IF(A1>=33, "Pass", "Fail")` |
| **AND** | `=AND(logical1, [logical2], ...)` | सभी शर्तें सही होने पर TRUE देता है। | `=AND(A1>10, B1<20)` |
| **OR** | `=OR(logical1, [logical2], ...)` | कोई भी एक शर्त सही होने पर TRUE देता है। | `=OR(A1="Yes", B1="Yes")` |
| **NOT** | `=NOT(logical)` | लॉजिक को उल्टा कर देता है (TRUE को FALSE)। | `=NOT(A1>50)` |
| **IFERROR** | `=IFERROR(value, value_if_error)` | अगर फ़ॉर्मूला में Error आये, तो कस्टम मैसेज दिखाता है। | `=IFERROR(A1/B1, "Error!")` |
| **IFS** | `=IFS(logical_test1, value1, [test2, value2]...)` | कई IF कंडीशन्स को एक साथ चेक करता है। | `=IFS(A1>90, "A", A1>75, "B")` |

## 3. LOOKUP & REFERENCE FUNCTIONS (खोजने वाले फ़ॉर्मूले)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **VLOOKUP** | `=VLOOKUP(lookup_value, table_array, col_index, [range_lookup])` | वर्टिकल कॉलम में वैल्यू ढूंढकर संबंधित डाटा लाता है। | `=VLOOKUP(101, A1:D100, 2, FALSE)` |
| **HLOOKUP** | `=HLOOKUP(lookup_value, table_array, row_index, [range_lookup])` | हॉरिजॉन्टल (पंक्ति) में वैल्यू ढूंढकर डाटा लाता है। | `=HLOOKUP("Sales", A1:Z5, 3, FALSE)` |
| **XLOOKUP** | `=XLOOKUP(lookup_val, lookup_arr, return_arr, [if_not_found])` | VLOOKUP और HLOOKUP का आधुनिक और शक्तिशाली विकल्प। | `=XLOOKUP(A2, EmployeeDB[ID], EmployeeDB[Name])` |
| **INDEX** | `=INDEX(array, row_num, [column_num])` | रो और कॉलम नंबर के आधार पर सेल की वैल्यू देता है। | `=INDEX(A1:C10, 2, 3)` |
| **MATCH** | `=MATCH(lookup_value, lookup_array, [match_type])` | किसी वैल्यू की पोजीशन (रो या कॉलम नंबर) बताता है। | `=MATCH("Apple", A1:A10, 0)` |
| **INDIRECT** | `=INDIRECT(ref_text)` | टेक्स्ट के रूप में लिखे सेल एड्रेस से वैल्यू लेता है। | `=INDIRECT("B" & 5)` |
| **OFFSET** | `=OFFSET(reference, rows, cols, [height], [width])` | स्टार्टिंग सेल से दूरी के हिसाब से रेंज की वैल्यू देता है। | `=OFFSET(A1, 2, 3)` |

## 4. TEXT FUNCTIONS (टेक्स्ट फ़ॉर्मूले)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **CONCATENATE / CONCAT** | `=CONCATENATE(text1, [text2], ...)` | दो या अधिक टेक्स्ट स्ट्रिंग्स को आपस में जोड़ता है। | `=CONCAT(A1, " ", B1)` |
| **TEXTJOIN** | `=TEXTJOIN(delimiter, ignore_empty, text1, ...)` | किसी सिंबल (जैसे कॉमा, स्पेस) के साथ टेक्स्ट जोड़ता है। | `=TEXTJOIN(", ", TRUE, A1:A5)` |
| **LEFT / RIGHT / MID** | `=LEFT(text, [num_chars])` | टेक्स्ट की बाईं, दाईं या बीच की अक्षर संख्या निकालता है। | `=LEFT("Excel", 2) → "Ex"` |
| **LEN** | `=LEN(text)` | सेल में कुल अक्षरों (Characters) की संख्या गिनता है। | `=LEN("Hello World") → 11` |
| **UPPER / LOWER / PROPER** | `=UPPER(text) / =PROPER(text)` | अक्षरों को Capital, Small या Proper Case में बदलता है। | `=PROPER("john doe") → "John Doe"` |
| **TRIM** | `=TRIM(text)` | टेक्स्ट के आगे, पीछे या बीच के अतिरिक्त स्पेस हटाता है। | `=TRIM(" Data ") → "Data"` |
| **REPLACE / SUBSTITUTE** | `=SUBSTITUTE(text, old_text, new_text)` | पुराने टेक्स्ट को नए टेक्स्ट से बदलता है। | `=SUBSTITUTE(A1, "2023", "2024")` |
| **TEXT** | `=TEXT(value, format_text)` | किसी नंबर या डेट को टेक्स्ट फॉर्मेट में बदलता है। | `=TEXT(TODAY(), "dd-mmm-yyyy")` |

## 5. STATISTICAL & COUNTING FUNCTIONS (गिनती और सांख्यिकी)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **COUNT** | `=COUNT(value1, [value2], ...)` | केवल नंबर वाले सेलों की गिनती करता है। | `=COUNT(A1:A10)` |
| **COUNTA** | `=COUNTA(value1, [value2], ...)` | सभी नॉन-एम्प्टी (जो खाली नहीं हैं) सेलों को गिनता है। | `=COUNTA(A1:A10)` |
| **COUNTBLANK** | `=COUNTBLANK(range)` | केवल खाली (Empty) सेलों की गिनती करता है। | `=COUNTBLANK(A1:A10)` |
| **COUNTIF** | `=COUNTIF(range, criteria)` | किसी एक शर्त के आधार पर सेलों को गिनता है। | `=COUNTIF(A1:A10, "Pass")` |
| **COUNTIFS** | `=COUNTIFS(criteria_range1, criteria1, ...)` | एक से अधिक शर्तों के आधार पर सेलों की गिनती करता है। | `=COUNTIFS(A1:A10, "Pass", B1:B10, ">80")` |
| **MAX / MIN** | `=MAX(number1, ...) / =MIN(number1, ...)` | सबसे बड़ी या सबसे छोटी संख्या निकालता है। | `=MAX(B1:B50)` |
| **LARGE / SMALL** | `=LARGE(array, k) / =SMALL(array, k)` | k-वीं सबसे बड़ी या k-वीं सबसे छोटी संख्या निकालता है। | `=LARGE(A1:A10, 2) (2nd Largest)` |
| **RANK** | `=RANK(number, ref, [order])` | किसी लिस्ट में किसी संख्या की रैंक बताता है। | `=RANK(A1, A1:A10)` |

## 6. DATE & TIME FUNCTIONS (दिनांक एवं समय)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **TODAY** | `=TODAY()` | आज की तारीख (Current Date) दिखाता है। | `=TODAY()` |
| **NOW** | `=NOW()` | वर्तमान तारीख और समय (Date and Time) दोनों दिखाता है। | `=NOW()` |
| **DAY / MONTH / YEAR** | `=DAY(serial_number)` | तारीख में से दिन, महीना या साल अलग करता है। | `=MONTH(TODAY())` |
| **DATEDIF** | `=DATEDIF(start_date, end_date, unit)` | दो तारीखों के बीच का अंतर (वर्ष, माह, दिन) निकालता है। | `=DATEDIF(A1, B1, "Y")` |
| **EDATE / EOMONTH** | `=EDATE(start_date, months)` | कुछ महीनों बाद या पहले की तारीख / महीने का अंतिम दिन। | `=EOMONTH(TODAY(), 0)` |
| **NETWORKDAYS** | `=NETWORKDAYS(start_date, end_date, [holidays])` | दो तारीखों के बीच के कार्य दिवस (Working Days) गिनता है। | `=NETWORKDAYS(A1, B1)` |
| **WEEKDAY** | `=WEEKDAY(serial_number, [return_type])` | सप्ताह का दिन नंबर के रूप में (1=Sun, 2=Mon) बताता है। | `=WEEKDAY(TODAY())` |

## 7. FINANCIAL & ADVANCED ARRAY FUNCTIONS (वित्तीय एवं आधुनिक)

| FORMULA | SYNTAX | विवरण (DESCRIPTION) | EXAMPLE |
|---|---|---|---|
| **PMT** | `=PMT(rate, nper, pv, [fv], [type])` | ऋण (Loan) के लिए मासिक किश्त (EMI) की गणना करता है। | `=PMT(8%/12, 60, -500000)` |
| **FV / PV** | `=FV(rate, nper, pmt, [pv])` | निवेश का भविष्य मान (Future Value) या वर्तमान मान। | `=FV(6%/12, 120, -5000)` |
| **UNIQUE** | `=UNIQUE(array)` | लिस्ट से डुप्लीकेट हटाकर यूनिक वैल्यू निकालता है (Excel 365)। | `=UNIQUE(A1:A100)` |
| **SORT** | `=SORT(array, [sort_index], [sort_order])` | डेटा को स्वचालित रूप से आरोही/अवरोही क्रम में लगाता है। | `=SORT(A1:B10, 1, 1)` |
| **FILTER** | `=FILTER(array, include, [if_empty])` | शर्तों के आधार पर डेटा को फ़िल्टर करके डायनामिकली दिखाता है। | `=FILTER(A1:C10, B1:B10="IT")` |