# Documentation on index

the **INDEX** function as a way to find a specific item in a table by its address. Instead of searching by a name, you tell it the exact row and column numbers where the data is located. It's like finding a book in a library by its shelf and book number. 📚

**Basic Syntax**

The most common syntax for the INDEX function is:

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

* **array**: This is your data table or range of cells. This is the area where you'll be looking for your value.
* **row\_num**: This is the row number within your chosen array. **Important:** This is a relative number, not the row number on the Excel sheet itself. For example, if your table starts on row 5, the first row of your table is 1, the second is 2, and so on.
* **[column\_num]**: This is the column number within your array. This is optional if your array is a single column. Like the row number, it's relative to your selected array.

**Simple Example**

Let's say you have a table in cells **A1:C5** with the following data:

|  | **A** | **B** | **C** |
| --- | --- | --- | --- |
| **1** | Product | Price | Stock |
| **2** | Apple | $1.20 | 50 |
| **3** | Banana | $0.80 | 30 |
| **4** | Orange | $1.50 | 25 |
| **5** | Grapes | $2.00 | 40 |

Export to Sheets

If you want to find the price of Bananas, you can use the INDEX function.

1. The array is the whole data range, **A2:C5**.
2. "Banana" is in the **2nd row** of this array. So, row\_num is **2**.
3. The "Price" is in the **2nd column** of this array. So, column\_num is **2**.

Your formula would be: =INDEX(A2:C5, 2, 2)

The result will be **$0.80**.

# Documentation on match

**MATCH** function as a simple search tool that tells you an item's position in a list. It doesn't return the item itself, but its location—for example, "the third item in the list."

**Basic Syntax**

The MATCH function has this syntax:

=MATCH(lookup\_value, lookup\_array, [match\_type])

* **lookup\_value**: The item you're looking for. This can be text in quotes (e.g., "Banana") or a cell reference (e.g., A5).
* **lookup\_array**: This is the single row or single column where you want to search. You can't use a whole table.
* **[match\_type]**: This is optional, but it's very important. It tells Excel how to search for the value.
  + **0 (zero)**: This is the most common and recommended option for beginners. It finds an **exact match**. The data in your list doesn't need to be sorted.
  + **1** (or omitted): Finds the largest value that is less than or equal to the lookup\_value. Your list **must** be sorted in ascending order.
  + **-1**: Finds the smallest value that is greater than or equal to the lookup\_value. Your list **must** be sorted in descending order.

**Simple Example**

Let's use a list of names to find a person's position in the list.

|  | **A** |
| --- | --- |
| **1** | Sarah |
| **2** | Mike |
| **3** | Jane |
| **4** | David |

Export to Sheets

If you want to find the position of "Jane" in this list, your formula would be:

=MATCH("Jane", A1:A4, 0)

* The lookup\_value is "Jane".
* The lookup\_array is the range **A1:A4**.
* The match\_type is **0** to ensure an exact match.

The formula will return **3**, because "Jane" is the third item in the list.

# Documentation on index-match

INDEX-MATCH is a powerful and very popular combination of two Excel functions that, when used together, can look up data more flexibly and efficiently than VLOOKUP. Think of it as a two-step process: **MATCH** finds the position, and then **INDEX** gets the item at that position.

**Why use INDEX-MATCH?**

Most beginners learn VLOOKUP first, but it has a major limitation: it can only look for data to the **right** of your lookup column. INDEX-MATCH overcomes this limitation, allowing you to search for data anywhere in your table.

**The Formula in Two Simple Steps**

The basic idea is to nest the MATCH function inside the INDEX function. The MATCH function finds the **row number**, which is then used by INDEX to return the value.

The formula looks like this:

=INDEX(data\_to\_return, MATCH(lookup\_value, lookup\_column, 0))

1. **MATCH(lookup\_value, lookup\_column, 0)**: This part of the formula runs first. It finds the position of your lookup\_value within a single column (lookup\_column). The 0 at the end ensures an **exact match**. The result of this part is a number—the row position.
2. **INDEX(data\_to\_return, [result from MATCH])**: The INDEX function then uses the row number returned by MATCH to go to the correct row in your data\_to\_return column and gets the value from that cell.

<br>

**Example Walkthrough**

Let's use a simple example to see this in action. Imagine a table where you want to find the **"ID"** for a specific **"Name"**. The "ID" is to the left of the "Name" column.

|  | **A** | **B** | **C** |
| --- | --- | --- | --- |
| **1** | ID | Name | Department |
| **2** | 101 | Sarah | Sales |
| **3** | 102 | Mike | Marketing |
| **4** | 103 | Jane | HR |
| **5** | 104 | David | IT |

Export to Sheets

You want to find Jane's ID.

**Step 1: Find Jane's Row with MATCH** First, write the MATCH part of the formula to find the row number for "Jane."

=MATCH("Jane", B2:B5, 0)

* lookup\_value: "Jane"
* lookup\_array: The Name column, **B2:B5**
* match\_type: 0 for an exact match

This formula will return the number **3** because "Jane" is the third name in the list.

**Step 2: Use INDEX to Get the ID** Now, use the INDEX function to get the value from the ID column (**A2:A5**) on the third row.

=INDEX(A2:A5, 3)

* array: The ID column, **A2:A5**
* row\_num: The number **3** from the MATCH function

This will return **103**, which is Jane's ID.

**Putting It All Together** By combining these two steps, you can create a single formula:

=INDEX(A2:A5, MATCH("Jane", B2:B5, 0))

This powerful combination performs a flexible lookup that VLOOKUP can't, making it an essential tool for any Excel user.

# Documentation on Xlookup

**XLOOKUP** function is a modern and more straightforward way to find information in your spreadsheet. It's designed to be an improvement on older functions like VLOOKUP and HLOOKUP, making lookups easier and more powerful.

**The Main Idea**

Think of XLOOKUP as a smart search tool. You tell it:

1. **What** you want to find.
2. **Where** to look for it.
3. **Where** to get the corresponding result from.

The great thing is, unlike VLOOKUP, XLOOKUP doesn't care if your lookup column is on the left or the right of the data you want to return. It can look in any direction.

**Basic Syntax**

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

* **lookup\_value**: The value you're searching for. This is a required argument.
* **lookup\_array**: The range of cells that contains the lookup\_value. This is a required argument.
* **return\_array**: The range of cells that contains the result you want to get back. This is also required.
* **[if\_not\_found]**: *(Optional)* The message or value to show if a match isn't found. This is a great feature that prevents the dreaded #N/A error.
* **[match\_mode]**: *(Optional)* 0 (zero) for an exact match. This is the default, which is perfect for beginners. You don't even need to type it!
* **[search\_mode]**: *(Optional)* 1 for a search starting at the first item, -1 to search from the last item.

<br>

<br>

**Simple Example**

Let's say you have a list of employee names and their IDs, and you want to find an employee's ID using their name. The employee IDs are in a column to the left of the names.

|  | **A** | **B** |
| --- | --- | --- |
| **1** | Employee ID | Name |
| **2** | 101 | Sarah |
| **3** | 102 | Mike |
| **4** | 103 | Jane |
| **5** | 104 | David |

Export to Sheets

To find Mike's ID, you can use XLOOKUP.

1. **lookup\_value**: "Mike"
2. **lookup\_array**: The column with the names, **B2:B5**
3. **return\_array**: The column with the IDs, **A2:A5**

Your formula would be: =XLOOKUP("Mike", B2:B5, A2:A5)

This formula will return **102**. You can see how easy it is to look "left" in the table.

This video provides a helpful, step-by-step tutorial on the XLOOKUP function for beginners, explaining the required and optional arguments.