

CELL REFERENCING AND NAMING

N5 ✕ ✓ <i>fx</i> =J5*M5*N2					
	J	K	L	M	N
1					
2	Location:			Penalty Rate:	0.36%
3					
4	Amount	Invoice Month	Invoice Day	Over Due By	Late Charge
5	\$742.50	Mar	2	10	\$26.73
6	\$1,021.02	Apr	2	0	\$0.00
7	\$409.53	Mar	16	0	#VALUE!
8	-\$234.96	Mar	25	0	\$0.00
9	-\$450.12	Mar	17	0	\$0.00
10	\$114.18	Apr	9	0	#VALUE!

We want to calculate the **LATE CHARGE** using the values of **AMOUNT**, **OVERDUE BY**, and a fixed **PENALTY RATE** (which is in cell N2).

The formula used in cell N5 is: =J5*M5*N2

This works correctly for N5, but when you copy the formula down to other rows (like N6), the reference to N2 also changes relatively to N3, N4, and so on. This is because Excel uses **relative referencing** by default, which adjusts the cell references based on the position where the formula is copied.

However, in our case, we want the **PENALTY RATE** to always point to **cell N2**, no matter where the formula is copied. To do that, we use **absolute referencing** by adding dollar signs like this:

N5 ✕ ✓ <i>fx</i> =J5*M5*\$N\$2					
	J	K	L	M	N
1					
2	Location:			Penalty Rate:	0.36%
3					
4	Amount	Invoice Month	Invoice Day	Over Due By	Late Charge
5	\$742.50	Mar	2	10	\$26.73
6	\$1,021.02	Apr	2	0	\$0.00
7	\$409.53	Mar	16	0	\$0.00
8	-\$234.96	Mar	25	0	\$0.00
9	-\$450.12	Mar	17	0	\$0.00
10	\$114.18	Apr	9	0	\$0.00

- \$N\$2 locks both the column (N) and the row (2), so it won't change when copied.
- You can quickly toggle between relative and absolute references by selecting the cell reference (e.g., N2) in the formula bar and pressing **F4** on your keyboard.

This ensures your formula works correctly in all rows by always referring to the fixed PENALTY RATE in N2.

But there is a better way than this; you can use a named range. I'll show you how:

name box

	J	K	L	M	N	O
1						
2	Location:			Penalty Rate:	0.36%	
3						
4	Amount	Invoice Month	Invoice Day	Over Due By	Late Charge	0.0036
5	\$742.50	Mar	2	10	=J5*M5*Penalty_Rate	
6	\$1,021.02	Apr	2	0	\$0.00	
7	\$409.53	Mar	16	0	\$0.00	
8	-\$234.96	Mar	25	0	\$0.00	
9	-\$450.12	Mar	17	0	\$0.00	
10	\$114.18	Apr	9	0	\$0.00	

To make formulas easier to read and maintain, you can assign a **name** to a specific cell.

For example, I selected cell N2, then went to the **Name Box** (located to the left of the formula bar) and renamed it to: **Penalty_Rate**

Now, instead of using the cell reference N2, we can use the name Penalty_Rate in our formulas.

So, the formula in cell N5 becomes: **=J5*M5*Penalty_Rate**

And if I check the formula in cell N10, it updates the row references accordingly:

=J10*M10*Penalty_Rate

Notice that Penalty_Rate remains constant in all rows, just like an absolute reference (like \$N\$2). Using named ranges not only locks the reference but also makes your formulas more readable and meaningful.

Note: There is a rule for assigning a name to a specific cell. Those rules are already mentioned in the 00_Named_Ranges PDF, which is provided by the course instructor.

Note: There are also other ways to name a cell.

COUNTIFS():

Purpose: Counts the number of cells that meet **multiple criteria**.

Syntax: =COUNTIFS(range1, criteria1, [range2, criteria2], ...)

Ex.

Use: =COUNTIFS(A2:A10, ">30", B2:B10, "Male")

Result: Counts how many rows have Age > 30 and Gender = Male

SUMIFS():

Purpose: Adds the values that meet **multiple criteria**.

Syntax: =SUMIFS(sum_range, criteria_range1, criteria1, [criteria_range2, criteria2], ...)

Ex.

Use: =SUMIFS(C2:C10, A2:A10, "North", B2:B10, ">500")

Result: Sums Sales (C2:C10) where Region is North and Amount > 500

AVERAGEIFS():

Purpose: Calculates the **average** of cells that meet **multiple criteria**.

Syntax: =AVERAGEIFS(average_range, criteria_range1, criteria1, [criteria_range2, criteria2], ...)

MAXIFS():

Purpose: Finds the **maximum value** among cells that meet **multiple criteria**.

Syntax: =MAXIFS(max_range, criteria_range1, criteria1, [criteria_range2, criteria2], ...)

MINIFS():

Purpose: Finds the **minimum value** among cells that meet **multiple criteria**.

Syntax: =MINIFS(min_range, criteria_range1, criteria1, [criteria_range2, criteria2], ...)

Note: All these **IFS** functions allow you to apply **more than one condition**.