

# How can I add days to a specific date in VBA, and what are some examples of using this feature?

Authored by  
**stats writer**

June 24, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I add days to a specific date in VBA, and what are some examples of using this feature?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=150405>

The process of adding days to a specific date in VBA involves using the DateAdd function to perform the calculation. This function takes in three parameters - the interval, the number of days to add, and the specific date. The interval can be specified as "d" for days, "m" for months, or "y" for years. This feature is useful for tasks such as calculating future due dates or tracking project timelines. For example, a VBA code can be written to automatically add 7 days to the current date and display the result. Another example is using this feature to calculate the number of days between two dates by subtracting the earlier date from the later one. This feature can also be utilized in creating dynamic forms or reports that require frequent updating of dates.

## Add Days to Date in VBA (With Examples)

**You can use the DateAdd function in VBA to add a specific number of days to a date.**

**Here is one common way to use this function in practice:**

```
Sub AddDays()
```

```
Dim i As Integer
```

```
For i = 2 To 10
```

```
Range("B" & i) = DateAdd("d", 4, Range("A" & i))
```

```
Next i
```

```
End Sub
```

**This particular macro will add four days to each date in the range A2:A10 and display the new dates in the**

**range B2:B10.**

**Note that the "d" argument in the DateAdd function specifies that we would like to add *days* to the dates as opposed to another unit of time.**

**Refer to the for a complete list of units you can use in the DateAdd function.**

**The following example shows how to use this syntax in practice.**

**Example: Add Days to Date in VBA**

**Suppose we have the following list of dates in Excel:**

	A	B	C	D	E	F
1	<b>Date</b>					
2	1/1/2023					
3	1/5/2023					
4	2/14/2023					
5	3/15/2023					
6	4/12/2023					
7	5/22/2023					
8	6/1/2023					
9	7/30/2023					
10	10/31/2023					
11						
12						
13						
14						
15						
16						
17						
18						
19						

**Suppose we would like to add four days to each date and display the new dates in column B.**

**We can create the following macro to do so:**

```
Sub AddDays()
```

```
Dim i As Integer
```

```
For i = 2 To 10
```

```
Range("B" & i) = DateAdd("d", 4, Range("A" & i))
```

```
Next i
```

## End Sub

When we run this macro, we receive the following output:

	A	B	C	D	E
1	<b>Date</b>	<b>Date + 4 Days</b>			
2	1/1/2023	1/5/2023			
3	1/5/2023	1/9/2023			
4	2/14/2023	2/18/2023			
5	3/15/2023	3/19/2023			
6	4/12/2023	4/16/2023			
7	5/22/2023	5/26/2023			
8	6/1/2023	6/5/2023			
9	7/30/2023	8/3/2023			
10	10/31/2023	11/4/2023			
11					
12					
13					
14					
15					
16					
17					
18					

Notice that column B contains each of the dates in column A with four days added to them.

Feel free to change the numeric value in the DateAdd function to add a different number of days to each date.