

# Re: DateAdd Function

---

*Source:* <http://www.tech-archive.net/Archive/Access/microsoft.public.access.forms/2006-08/msg01424.html>

---

- *From:* "Rick Brandt" <[rickbrandt2@xxxxxxxxxxxx](mailto:rickbrandt2@xxxxxxxxxxxx)>
  - *Date:* Wed, 09 Aug 2006 12:39:38 GMT
- 

Christ4 wrote:

I am VERY new at this. I am trying to set a default "Date Due" to be 7 days from the "Date Out" unless the "Student" option is selected, then I want the "Date Due" to be 1 day from the "Date Out." Please help!

You can't use the actual DefaultValue property for this. That is applied as soon as the new record is painted on the screen before your [Date Out] field has been populated. What you can do is "push" a value into the [Date Due] field in the AfterUpdate event of the TextBox used for [Date Out].

```
If Me![Student] = True Then
Me![Date Due] = DateAdd("d", 1, Me![Date Out])
Else
Me![Date Due] = DateAdd("d", 7, Me![Date Out])
End If
```

You might need to put similar code in the AfterUpdate of the [Student] control in case they change that after setting the [Date Out].

However; if this rule is *\*always\** true (no exceptions), then [Date Due] should not even be a field stored in your table. It should just be calculated on-the-fly based on the other two fields. You can do that in queries, forms, and reports making actual storage of the result unnecessary and unwise.

Example of that on-the-fly expression in a query would be...

```
Date Due: IIf([Student]=True, DateAdd("d", 1, [Date Out]), DateAdd("d", 7, [Date Out]))
```

By calculating instead of storing you are guaranteed to always have the correct result for the Date Due. As soon as you save it in your table it could be changed there or either of the other two fields could be changed making the record incorrect.

---

Re: DateAdd Function

Rick Brandt, Microsoft Access MVP  
Email (as appropriate) to...  
RBrandt at Hunter dot com

—

Rick Brandt, Microsoft Access MVP  
Email (as appropriate) to...  
RBrandt at Hunter dot com

.