

## Re: Need to avoid "field X cannot contain a Null value"

---

*Source:* <http://www.tech-archive.net/Archive/Access/microsoft.public.access.forms/2006-07/msg00128.html>

---

- *From:* [sjg1314@xxxxxxxxxx](mailto:sjg1314@xxxxxxxxxx)
  - *Date:* 4 Jul 2006 07:02:56 -0700
- 

Allen, thanks so much for taking the time to reply.

Allen Browne wrote:

At the most basic level, the user can get out of the field if they either make no entry, or undo the entry (press Esc.)

The problem with this is that you "just have to know" this trick (Ctrl-Z works too). For maximum usability, this is unacceptable.

If you want to trap the error and replace it with your own error message, you can do that in the Error event of the form.

I thought of that last night and will try it out today. The main problem with the error traps is that all you get is a number. You can convert this number to a string. I wish that, instead, the error handler would have an event structure that would indicate the object in which the error occurred. Any ideas?

If you want to delay the message until the record is being validated instead of the field, open the table in design view, and set the Required property to No for each field. Still in table design, open the Properties box (View menu), and enter a Validation Rule there for the table. Example: ([Surname] Is Not Null) AND ([City] Is Not Null)

I've seen this trick mentioned in a number of places. As an experienced database programmer (who is an MS Access/Visual Basic beginner), I find this horrifying. Instead of concentrating the knowledge of what fields are required in one place (the Table), this information has to be spread out into all the Forms that use the Table.

I realize MS Access is used for small databases designed by a single programmer, but I can't bring myself to do this. If a field is

Re: Need to avoid "field X cannot contain a Null value"

required, this information should be part of the Schema.

I have not tried to programmatically create the event procedures and modify existing event procedures in the way you describe. With the form open in design view, you can determine whether an event procedure or macro is handling the event of a control by examining its property, e.g.:  
? Forms!Form1!Text0.BeforeUpdate  
which returns the string "[Event Procedure]" if you have one. See help on CreateEventProc if you want to follow that approach.

I'm aware of this. I'm also aware that what I'm asking for is fairly difficult. I was hoping either someone already had working code or that I would be told: "You idiot! Just put a comma in the event field and add the next event handler." No such luck, it appears.

The important thing here is that if you, a Microsoft MVP, don't have any better suggestions, then I know I can stop looking.

Thanks again, Allen, and thanks also for your Web page of tips.

One more question, master of VBA: is there any way to tell if an object has a property? All my attempts caused a run-time or compile error to occur. Some of my code would be greatly simplified if I could just say "If IsProperty(x.a) Then ...".

I'm stuck with MS Access 2000, so if this feature has been added later, it won't help much. If it can be done with some trickery with error handlers, that would be great, but I wasn't able to get the error handler to catch this error.

.