Can I negate the value in an attribute that uses an "Eval"?

Asked 15 years, 9 months ago   Modified 8 years, 5 months ago   Viewed 13k times

▲

15

▼

🔖

🕐

I'd like to set a button's enabled state to be the negation of a value.

So given this code:

```
<asp:CheckBox ID="DefaultChecked"
              Checked='<%# Bind("IsDefaultMessage") %>'
              Enabled="false"
              runat="server" />

<asp:LinkButton ID="MakeDefaultButton"
                runat="server"
                CommandName="MakeDefault'
                CommandArgument='<%# Bind("ResidentialInfoID") %>'
                Text="Make Default" />
```

How can I make the LinkButton `Enabled` attribute `false` if `IsDefaultMessage == true` ?

`asp.net`   `asp.net-3.5`

Share

Improve this question

Follow

edited Oct 9, 2013 at 10:04

Jeroen
63.5k ● 46 ● 227 ● 365

asked Mar 4, 2009 at 15:23

KevDog
5,803 ● 9 ● 45 ● 74

6 Answers

Sorted by: [Highest score (default) ⇕]

▲

24

▼

🔖

✓

🕐

Use Eval instead of Bind. Bind is for two-way binding, i.e. for cases where you need to be able to save the data back to your data source.

When you use Bind, the compiled page will actually have generated code that uses Eval to set the value, plus some code to read out the value for saving. Because Bind is replaced with generated code, you can't use any extra logic with Bind.

```
<asp:CheckBox ID="DefaultChecked" Checked='<%# !(bool)Eval("IsDefaultMessage")
%>' Enabled="false" runat="server" />
<asp:LinkButton ID="MakeDefaultButton" runat="server" CommandName="MakeDefault'
CommandArgument='<%#Bind("ResidentialInfoID") %>' Text="Make Default"/>
```

▲

6

▼

🔖

🕘

If you can use Eval, it is just a method of the `Control` class. It's only special in that it needs to be in the context of a data bound block `<%# ... %>`. Other than that, you can basically treat the block like a regular `<%= %>` expression block:

```
<%# !(bool)Eval("IsDefaultMessage") %>
```

If you want to still Bind it (Eval isn't round-trip), than you'll need to negate it back and forth during databinding. You may not need to do this though, if you can just re-word the control. For example, if a check box, instead of labelling it "Is Not Default Message" to the user and negating it back and forth, than lable it "Is Default Message". Contrived example, but you get the idea.

> Doing this throws an "invalid preprocessor directive node" error. Any ideas on how to solve? – KevDog Mar 4, 2009 at 15:47

> Also, "Cannot apply operator ! to operand of type object" – KevDog Mar 4, 2009 at 15:49

> You're right--Bind can't be used like I had it. I switched it to Eval and described how to workaround Bind usage. – Jon Adams Mar 5, 2009 at 16:29

> Also, "Cannot apply operator ! to operand of type object" – ch2o Apr 25, 2013 at 14:20

> @ch2o: Fixed by casting to the proper type – Jon Adams Apr 25, 2013 at 17:24

▲

2

▼

🔖

🕘

In case someone is looking for an option with VB.Net

```
<asp:CheckBox ID="DefaultChecked" Checked='<%# NOT (Eval("IsDefaultMessage")) %>' Enabled="false" runat="server" />
```

I've never used Bind but my understanding is that it is similar to Databinder.Eval. Either way, both methods return objects so you need to cast it to a boolean before evaluating it.

```
<%# !Convert.ToBoolean(Bind("IsDefaultMessage") %>
```

Edit: Looks like this can't be done and using a SqlDataSource on the page would solve the problem. http://forums.asp.net/t/1009497.aspx.

Share
Improve this answer
Follow

edited Mar 4, 2009 at 20:06

answered Mar 4, 2009 at 15:58

rmw
1,253 ●1 ●12 ●20

> This throws the error "The name 'Bind' does not exist in the current context" – KevDog Mar 4, 2009 at 16:09

> Is there a reason you want to use Bind? Have you tried using DataBinder.Eval? <%# !Convert.ToBoolean(DataBinder.Eval(Container.DataItem, "IsDefaultMessage")) %> – rmw Mar 4, 2009 at 18:38

> Here is a forum about this issue: forums.asp.net/t/1009497.aspx. Can you add an SqlDataSource to your page? It looks like that might make it work. – rmw Mar 4, 2009 at 18:51

As I recall (It's been a while), there's no particular magic in `<%#Bind(`. It's just `#Bind(` inside <%....%>. Which means you'd want:

```
<%  ! #Bind("IsDefaultMessage") %>'
```

Share Improve this answer Follow

answered Mar 4, 2009 at 15:28

James Curran
103k ●37 ●185 ●262

> this won't work... the <%#...%> lets ASP.NET know it is taking place during databinding... the ! must go after the #. – Jeff Martin Mar 4, 2009 at 15:41

The code

```
Checked='<%# Eval("IsDefaultMessage").ToString().Length() > 4 %>'
```

will return true if IsDefaultMessage is false Since "False".Length = 5 and "True".Length = 4

edited Mar 28, 2012 at 15:37

answered Mar 28, 2012 at 15:09

Manuel Alves
4,013 ●2 ●32 ●24