

Re: Casting in a where clause

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.programming/2004-11/4616.html>

From: Aaron [SQL Server MVP] (*ten.xoc_at_dnartreb.noraa*)

Date: 11/20/04

Date: Sat, 20 Nov 2004 12:50:54 -0500

Two small issues.

- (a) you can't cast -999 as decimal(4,2); you get an overflow.
- (b) you need to be confident that the user will never be looking for -999, or else NULL values in the table will erroneously match.

"The answer" is to store numeric data in a numeric datatype.

"The Ultimate Ozz" <The Ultimate Ozz@discussions.microsoft.com> wrote in message news:9661B106-ED51-4DB2-AD3D-6E108582BF6F@microsoft.com...

> *Looks like you never got the answer.*

>

> *Here it should be:*

>

> *where ISNUMERIC(ISNULL(static_mos, '-999')) = 1 and CAST(
> isnull(static_mos, '-999') AS decimal(4,2)) = uservalue*

>

> *From the pro....*

> *- Azhar*

>

> *"scottrg" wrote:*

>

> > *I'm having trouble in a SQL statement that queries from a view that has
> > non-numeric, null, and zero-length data in a varchar column. I'm trying
to*

> > *cast numeric data as a real number to compare against a user-entered
value.*

> > *Part of my SQL looks like:*

> >

> > *where ISNUMERIC(static_mos) = 1 and CAST(static_mos AS decimal(4,2)) =
> > uservalue*

> >

> > *For some reason the CAST always takes precedence and the query blows up
on*

> > *null values. Aren't "and" clauses supposed to be evaluated*

left-to-right? Is

> > *there any way I can get the ISNUMERIC to be evaluated first?*

microsoft.public.sqlserver.programming: Re: Casting in a where clause

> >