

Re: Difference between = and IN

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.mseq/2005-02/0037.html>

From: Andy (Andy_at_discussions.microsoft.com)

Date: 02/07/05

Date: Mon, 7 Feb 2005 06:17:04 -0800

Hi Hugo

Thanks for this – I wasn't aware there was an issue with IN/NOT IN and Nulls! I was under the impression that Nulls would just be excluded from the results. In this particular instance it will always be a single value or delimited list. However, the app also uses a sub query so I will certainly be having a look at that as well.

Thanks
Andy

"Hugo Kornelis" wrote:

> On Mon, 7 Feb 2005 02:21:01 -0800, Andy wrote:
>
> >Thanks Steve
> >So presumably if there is only one item it is more efficient
> >to use = as the query processor does not need to convert it whereas the IN
> >statement will be converted to = anyway?
> >
> >The main reason I am interested is that I am writing code that will build an
> >SQL statement and wanted to know if I would be better off using an IN
> >operator all the time or checking for instances where an = would suffice and
> >using that instead.
> >
> >Thanks
> >Andy
>
> Hi Andy,
>
> If you actually have to perform the subquery and count the number of rows
> returned to choose between IN and =, then it's best to use IN, so that the
> subquery is executed only once. If you have more efficient ways to
> determine the number of rows returned, use =.
>
> BTW, you might also consider using EXISTS instead of IN – it is generally
> more efficient, but more important: NOT IN has very unpleasant behaviour

microsoft.public.sqlserver.mseq: Re: Difference between = and IN

- > with *NULLS* and should be avoided (replace by *NOT EXISTS*); for uniformity,
- > I generally prefer to replace *IN* with *EXISTS* as well. I use *IN* and *NOT IN*
- > only with a delimited list of values, never with a subquery.
- >
- > *Best, Hugo*
- > --
- >
- > (Remove *_NO_* and *_SPAM_* to get my e-mail address)
- >