

Suppose you have a *table* (**tblDemo**) and a view (**vwDemo**) over that table that joins **tblDemo** to another table to retrieve their results.

The view might resemble the following:

```
CREATE VIEW vwDemo
AS

SELECT d.*, o.OtherField01, o.OtherField02

FROM tblDemo d

INNER JOIN tblOther o ON d.ID = o.ID
```

While I've never advocated selecting * from anything, it does have its uses; but it turns out there's potentially a nasty gotcha. Personally, I like to err on the side of higher specificity, though that usually requires a bit more manual tweaking as time goes on.

What happens if the schema of **tblDemo** changes? Well, if you don't update or recreate your view, you may very well get some unexpected side effects.

For instance, if a new field were added to **tblDemo** (e.g., *ALTER TABLE tblDemo ADD NewField int*), the view, when inspected in **SQL** shows the following:

```
SELECT d.ID, d.Field01, d.Field02, d.Field03, d.NewField AS OtherField01,
o.OtherField01 AS OtherField02, o.OtherField02

FROM tblDemo d

INNER JOIN tblOther o ON d.ID = o.ID
```

Notice that the '*NewField*' got aliased with the field name at the new fields position! That field, in turn, received the next field's alias, and so on. Understandably, this can have some very negative side effects in your applications.

For this reason I would strongly recommend AGAINST using **SELECT *** in a view in **SQL Server**. Of course, if your view does nothing but **SELECT *** on a single table or the '*' is the last part of

the **SELECT** clause (such as *SELECT o.OtherField01, o.OtherField02, d.* FROM ...*) then you circumvent the issue.

When used in conjunction with other columns, however, there are indeed repercussions.

Use **sp_refreshview** to fix this “issue”

To refresh the view's metadata information, run the **sp_refreshview** stored procedure against a typical view called, for example **V1**:

```
EXEC sp_refreshview 'dbo.V1';
```

A typical example is where a schema change in the underlying objects is not reflected in the views' metadata information. You might find it to be a good practice to refresh all views' metadata information after applying schema changes to objects in the database.

To avoid the tedious process of writing the **sp_refreshview** statements you can use the following query:

```
SELECT N'EXEC sp_refreshview `

+ QUOTENAME (VIEW_NAME, '''') + `;' AS cmd

FROM (SELECT QUOTENAME (TABLE_SCHEMA)

+ N'.' + QUOTENAME (TABLE_NAME) AS VIEW_NAME

FROM INFORMATION_SCHEMA.VIEWS) AS V

WHERE OBJECTPROPERTY (OBJECT_ID (VIEW_NAME) , 'IsSchemaBound') = 0;
```

The query will generate as its output the lines of code with the **sp_refreshview** statements against all views in the database that are not schema-bound.