#

**Article:** SELECT \* and Column Issue in View

**Date**: 19/12/2011

**Posted by**: HeelpBook Staff

**Source**: [Link](http://blog.sqlauthority.com/2010/09/15/sql-server-select-and-adding-column-issue-in-view-limitation-of-the-view%C2%A04/)

**Permalink**: [Link](http://heelpbook.altervista.org/2011/sql-server-select-and-adding-column-issue-in-view/)

This is very well known limitation of the **View**.

Once the **view** is created and if the basic table has any column added or removed, it is not usually reflected in the view till it is refreshed.

To test this, we will create a view where we will use **SELECT \*** and select everything from the table. Once the view is created, we will add a column to the view.

We will test that even though we have used **SELECT \***, the view does not retrieve the newly added column. Once we refresh the view using **SP\_REFRESHVIEW**, it will start retrieving the newly added column.

Run the following **T-SQL** script in SQL Server Management Studio **New Query Window**:

USE AdventureWorks
GO
IF EXISTS (SELECT \* FROM sys.views WHERE OBJECT\_ID = OBJECT\_ID(N'[dbo].[LimitView4]'))
DROP VIEW [dbo].[LimitView4]
GO

-- Create View
CREATE VIEW LimitView4
AS
SELECT \*
FROM HumanResources.Shift
GO

-- Select from original table
SELECT \*
FROM HumanResources.Shift
GO

-- Select from View
SELECT \*
FROM LimitView4
GO

-- Add Column to original Table
ALTER TABLE HumanResources.Shift
ADD AdditionalCol INT
GO

-- Select from original table
SELECT \*
FROM HumanResources.Shift
GO

-- Select from View
SELECT \*
FROM LimitView4
GO

-- Refresh the view
EXEC sp\_refreshview 'LimitView4'
GO

-- Select from original table
SELECT \*
FROM HumanResources.Shift
GO

-- Select from View
SELECT \*
FROM LimitView4
GO

-- Clean up
ALTER TABLE HumanResources.Shift
DROP COLUMN AdditionalCol
GO

Above query will return following resultset….



The same *limitation exits in the case of deleting the column* as well. This is a very well-known issue with the **Views**. The resolutions of these issues are as follows:

1. Refresh the views using **sp\_refreshview** stored procedure
2. Do not use **SELECT \*** but use **SELECT** column-names ;
3. Create view with **SCHEMABINDING**; this way, the underlying table will not get modified.