<http://codebetter.com/raymondlewallen/2005/05/09/the-difference-in-truncate-and-delete-in-sql-server/>

**The difference in TRUNCATE and DELETE in Sql Server**

I’ve answered this question many times, and answered it again this weekend.  What is the difference when doing aDELETE TableA instead of TRUNCATE TableA?   
A common misconception is that they do the same thing.  Not  
so.  In fact, there are many differences between the two.

DELETE is a logged operation on a per row basis.  This means  
that the deletion of each row gets logged and physically deleted.

You can DELETE any row that will not violate a constraint, while leaving the foreign key or any other contraint in place.

TRUNCATE is also a logged operation, but in a different way.   
TRUNCATE logs the deallocation of the data pages in which the data  
exists.  The deallocation of data pages means that your data  
rows still actually exist in the data pages, but the  
extents have been marked as empty for reuse.  This is what  
makes TRUNCATE a faster operation to perform over DELETE.

You cannot TRUNCATE a table that has any foreign key  
constraints.  You will have to remove the contraints, TRUNCATE the  
table, and reapply the contraints.

TRUNCATE will reset any identity columns to the default seed  
value.  This means if you have a table with an identity column and  
you have 264 rows with a seed value of 1, your last record will have  
the value 264 (assuming you started with value 1) in its identity  
columns.  After TRUNCATEing your table, when you insert a new  
record into the empty table, the identity column will have a value of  
1.  DELETE will not do this.  In the same scenario, if you  
DELETEd your rows, when inserting a new row into the empty table, the  
identity column will have a value of 265.