**Suggested Videos**  
[Part 61 - Creating a large table with random data for performance testing](http://csharp-video-tutorials.blogspot.com/2013/01/creating-large-table-with-random-data.html)  
[Part 62 - What to choose for performance - SubQuery or Joins](http://csharp-video-tutorials.blogspot.com/2013/01/what-to-choose-for-performance.html)  
[Part 63 - Cursors in sql server](http://csharp-video-tutorials.blogspot.com/2013/01/cursors-in-sql-server-part-63.html)   
  
In [Part 63](http://csharp-video-tutorials.blogspot.com/2013/01/cursors-in-sql-server-part-63.html), we have discussed about cursors. The example, in Part 63, took around 45 seconds on my machine. [Please watch Part 63](http://csharp-video-tutorials.blogspot.com/2013/01/cursors-in-sql-server-part-63.html), before proceeding with this video. In this video we will re-write the example, using a join.   
  
   
  
Update tblProductSales  
set UnitPrice =   
 Case   
 When Name = 'Product - 55' Then 155  
 When Name = 'Product - 65' Then 165  
 When Name like 'Product - 100%' Then 10001  
 End   
from tblProductSales  
join tblProducts  
on tblProducts.Id = tblProductSales.ProductId  
Where Name = 'Product - 55' or Name = 'Product - 65' or   
Name like 'Product - 100%'   
  
   
  
**When I executed this query,** on my machine it took less than a second. Where as the same thing using a cursor took 45 seconds. Just imagine the amount of impact cursors have on performance. Cursors should be used as your last option. Most of the time cursors can be very easily replaced using joins.   
  
**To check the result of the UPDATE statement, use the following query.**  
Select  Name, UnitPrice from   
tblProducts join  
tblProductSales on tblProducts.Id = tblProductSales.ProductId  
where (Name='Product - 55' or Name='Product - 65' or   
Name like 'Product - 100%')