

# SaveTimeStamps

## %SaveTimeStamps function

Function	<a href="#">Controlling function</a> , which enable/disable change of the timestamp at setting the parameters of local values or objects in D2000.		
Declaration	<pre>%SaveTimeStamps(     BOOL in _bSaveTimeStamps )</pre>		
Parameters	<table><tr><td><b>bSaveTimeStamps</b></td><td><a href="#">@TRUE</a> - allows <a href="#">@FALSE</a>- disallows  Change of timestamp at assignment.</td></tr></table>	<b>bSaveTimeStamps</b>	<a href="#">@TRUE</a> - allows <a href="#">@FALSE</a> - disallows  Change of timestamp at assignment.
<b>bSaveTimeStamps</b>	<a href="#">@TRUE</a> - allows <a href="#">@FALSE</a> - disallows  Change of timestamp at assignment.		
Description	<p><a href="#">Controlling function</a>, which enable/disable change of the timestamp at setting the parameters of local values or objects in D2000.</p> <p>After script start preservation of timestamp is not activated.</p> <p>E.g. assignment executing:</p>		
Example	<pre>INT _i _i := 1 + 1</pre> <p>value 2 is set for local variable <code>_i</code> whereby real time is set for timestamp (e.g. <code>10:10:11 22-4-2008</code>).</p> <p>Following row disables timestamp actualization:</p> <pre>%SaveTimeStamps (@FALSE)</pre> <p>Following assignment sets new value (3) to the local variable without timestamp change.</p> <pre>_i := 2 + 1</pre> <p><b>This attribute is applied within inferences!!!</b> E.g. by reading values from archive (action <a href="#">GETARCHARR</a>) the read values from archive are assigned to the final column <b>without affecting their timestamps!!!</b></p>		



**Related pages:**

[ESL extended functions](#)  
[Function arguments - types](#)