


# sort

Syntax	<div>sort anArray using &lt;expression with element and nextElement&gt;</div>													
Semantics	<div>sort allows you to sort an array by using an expression to compare array elements with each other. The sort algorithm will call the expression each time two elements need to be compared. The expression can return:</div> <table><tr><td rowspan="3">an integer</td><td>&lt; 0</td><td>element precedes nextElement</td></tr><tr><td>= 0</td><td>element and nextElement are equal</td></tr><tr><td>&gt; 0</td><td>element succeeds nextElement</td></tr><tr><td rowspan="2">a boolean</td><td>true</td><td>element precedes nextElement</td></tr><tr><td>false</td><td>element succeeds nextElement</td></tr></table> <div>It is not determined, what sort algorithm will actually be used, though a Quicksort derivative is the most common.</div> <div>Be aware that sort algorithms are not linear (typically <math>n \log n</math> to <math>n^2</math> iterations for <math>n</math> elements). Therefore, you should keep the expression as compact as possible.</div> <div>When using a boolean expression, make sure that equality always returns <b>true</b> (e.g. by using <b>&lt;=</b> instead of <b>&lt;</b>). Otherwise, the implemented sort algorithm may take longer than necessary or not terminate at all.</div>		an integer	< 0	element precedes nextElement	= 0	element and nextElement are equal	> 0	element succeeds nextElement	a boolean	true	element precedes nextElement	false	element succeeds nextElement
an integer	< 0	element precedes nextElement												
	= 0	element and nextElement are equal												
	> 0	element succeeds nextElement												
a boolean	true	element precedes nextElement												
	false	element succeeds nextElement												
Substitutables	anArray	Can be any variable or object attribute having the type <b>Array</b> .												
	<expression with element and next element>	This expression must use the currently evaluated array element and its next neighbor and indicates if the current order is ok or not.												
	element and nextElement are keywords and allow you to use relative references (as opposed to normal, absolute indices) to single elements of an array (see <a href="#">Get Array Element Operator [ ]</a> ).													
Examples	<pre>// sort an array of integers in ascending order set sorted = sort myList using element - nextElement;  //sort an array of strings in ascending order set sorted = sort myList using element.text &lt; nextElement.text;  //sort an array using dates in descending set sorted = sort myList using element.date &gt; nextElement.date;  // sort an array using boolean return values set sorted = sort myList using element &lt;= nextElement;  // sort an array with complex elements over 2 sort key set sorted = sort inputContainer.elements using (element.aString = nextElement.aString and element.anInteger &lt; nextElement.anInteger) or element.aString &lt; nextElement.aString;</pre>													

Example File (Builder project E2E Action Language/Array):



<your example path>\E2E Action Language\Array\uml\arraySort.xml