

# Storage.Delete

<b>Method</b>	<b>Delete</b>
<b>Module</b>	Storage
<b>Library</b>	Core
<b>Action</b>	Removes an existing user defined persistent storage table.
<b>Parameters</b>	Name (string): The name of the persistent storage table. Case is not significant.
<b>Return values</b>	Name (string): The table's name if the storage table is deleted from persistent storage, else nothing.
<b>Example</b>	<pre>@storageName = SELECT "LoggedInAudit" AS Name;  @result = Storage.Delete(Name:@storageName); IF (@result)     SELECT @storageName.Name    " Table deleted" AS Status FROM @storageName; ELSE     SELECT @storageName.Name    " Table not found" AS Status FROM @storageName; ENDIF;</pre>
<b>Platforms</b>	<ul style="list-style-type: none"><li>• Windows</li><li>• Linux</li><li>• MacOS</li><li>• Solaris Intel</li><li>• Solaris Sparc</li><li>• Android</li></ul>
<b>Notes</b>	<p>This method can be used with the <code>SCALE evaluate</code> statement to abandon processing if the persistent storage table is not present.</p> <p>See <a href="#">User Defined Persistent Storage</a> page for guidance using persistent storage tables.</p> <p>The following methods existing in the Agent Storage module:</p> <ul style="list-style-type: none"><li>• <a href="#">Storage.Check</a> — Tests the existence of a user defined persistent storage table.</li><li>• <a href="#">Storage.Delete</a> — Removes an existing user defined persistent storage table.</li><li>• <a href="#">Storage.Get</a> — Indicate whether a persistent storage table of the specified name is present and return its contents if present.</li><li>• <a href="#">Storage.GetRemote</a> — Retrieve the requested datum directly from the Platform central repository.</li><li>• <a href="#">Storage.Set</a> — Set or change the value of the named persistent storage table.</li></ul>