# Memory

## Stereotype Attributes

### Memory Adapter

| Attribute | Description | Allowed Values | |
|---|---|---|---|
| **scope** | Supply the scope of the operation. | **gl ob al** | The stored data is accessible globally to the service, and available across service calls (default). |
| | | se ssi on | The stored data is accessible within a service call. |
| **action** | ⓘ The adapter action derives from the used operation. Do not configure this. | | |

## Memory Adapter Operations

### store Operation

Store a value to memory.

- `store ( key : String, value : Any, hashMapKey : String, oldValue : Any )`

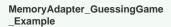| Name | Type | Direction | Description |
|---|---|---|---|
| **key** | String | in | Set a key that can be used to access the stored value with the retrieve operation. |
| **value** | Any | in | Provide the data that should be stored to the memory. |
| **hashMa pKey** | String | in | In case the stored value is a hash map, you can use **hashMapKey** to provide the key of a hash map entry you want to override.<br>Refer to Using the Memory Adapter with Maps for more information on this. |
| **oldValue** | Any | out | If the provided key is already present in the memory, it gets overwritten. In this case, **oldValue** returns the previous value. If no old value is present, **oldV alue** is NULL.<br>In case **value** contains a hash map, **oldValue** returns the complete previously stored hash map. |

### retrieve Operation

Retrieve a value from memory.

- `retrieve ( key : String, hashMapKey : String, value : Any )`

| Name | Type | Direction | Description |
|---|---|---|---|
| **key** | String | in | Provide the key of the value that has been stored to memory with store. If no value with this key can be found, the Memory adapter throws error `MEMADSM /9`. |
| **hashMa pKey** | String | in | In case the stored value is a hash map, you can use **hashMapKey** to retrieve a dedicated value from the map.<br>Refer to Using the Memory Adapter with Maps for more information on this. |
| **value** | Any | out | Returns the either the stored value, or the map entry. |

### remove Operation

**MemoryAdapter_GuessingGame _Example**

Click the icon to download a simple example model that shows the usage of the Memory adapter in **Scheer PAS** *Desig ner*.

**MemoryAdapter_HashMap_Exa mple**

Click the icon to download a simple example model that shows the usage of the Memory adapter with hash maps in **Sc heer PAS** *Designer*.

Remove a value from memory.

- remove ( key : String, hashMapKey : String, oldValue : Any )

| Name | Type | Direction | Description |
|------|------|-----------|-------------|
| **key** | String | in | Provide the key of the value that has been stored to memory with store. If no value with this key can be found, the Memory adapter throws error `MEMADSM/12`. |
| **hashMapKey** | String | in | In case the stored value is a hash map, you can use **hashMapKey** to provide the key of a hash map entry you want to remove.<br>Refer to Using the Memory Adapter with Maps for more information on this. |
| **oldValue** | Any | out | Returns the value of the memory item that has been removed. In case **value** contains a hash map, **oldValue** returns the complete previously stored hash map. |

## clear Operation

Clear all stored data. No parameters.