Object

As this is a very simplified documentation, we may retreat to the following statement: while 'primitive' values - like strings, singles etc. - contain only one value, objects are more complex. A very basic example is given in the overview on types: imagine an address which is composed maybe of a street name, a number, a zip code, and a city - then the whole thing is the address object, and the individual values are the properties of this object.

In the context of Titan macros, you'll most likely see objects like handles, timestamps, and some more. They are explained in more detail in separate articles - see the indivitual link further down.

Also, there seems to happen implicit parsing/conversion. E.g. the function Math.Min expects two objects as input - but in the example Chase - Double speed it also works with a number like 3600.0:

```
<step>Math.Min("Playbacks.Editor.Times.ChaseSpeed",
Playbacks.Editor.Times.ChaseSpeed * 2, 3600.0)</step>
```

Example for the use of objects: ActionScript.SetProperty:

```
ActionScript.SetProperty(String id, Object value)
```

This sets a particular property, identified by its id, to an object value.

More specifically, as mentioned there,

```
<step>ActionScript.SetProperty("Handles.SourceHandle",
Playbacks.Editor.SelectedPlayback)</step>
```

sets the property named "Handles.SourceHandle" to be the multidimensional object value Playbacks.Editor.SelectedPlayback.

You can find some details about the properties which are used in the overview about properties.

object types in this Wiki

- AcwColour
- AcwProgrammerIdPair
- AcwRecordMask
- AcwTimecodeSource
- AcwTimeSpan
- AcwUserNumber
- DmxAssignment
- FrameRate
- Leveladjust
- TimecodeTime
- Timestamp

However the most important object type in Titan is more in-depth explained as Handle.

Objects are also mentioned in

- The Syntax of Functions
- ActionScript.SetProperty
- Math.Cast.ToInteger
- Math.Cast.ToSingle
- Math.IsEqual
- Math.Min
- Math.ToString
- Timecode.ParseFrameRate
- Triggers.AddMidiTrigger
- Handles.PendingLegend
- Programmer.CurrentProgrammerIds
- Triggers.PendingTriggerTarget
- Types

From:
https://avosupport.de/wiki/ - **AVOSUPPORT**

Permanent link:
https://avosupport.de/wiki/macros/type/object

Last update: **2018/06/25 13:24**