

# Specifying parameters

The number of parameters a script operation can have depends on the script operation language:

- An OCL expression can have a single parameter, which by default specifies the [Contextual element](#). Though the parameter cannot be removed, you can easily reset the [Contextual element](#), if necessary.
- A binary expression inherits parameters from the Java class to which the expression body references. You cannot create additional parameters.
- Other script operations, such as JavaScript or Groovy, can have as many parameters as you need (see the following figure).

The screenshot shows a configuration window for a script operation. The 'Language' dropdown is set to 'StructuredExpression'. The 'Body' section on the left contains a tree view with a 'Script in JavaScript' node. Below it, a 'Create parameter...' button is highlighted with an orange circle and labeled '1'. The main area on the right shows the configuration for 'Script::arg1'. It includes buttons for 'Edit', 'Use as...', 'Reset', and 'Remove' (the 'Remove' button is highlighted with an orange circle and labeled '2'). Below these buttons are input fields for 'Operation Name' (containing 'String1') and 'Parameter Name' (containing 'arg1'). A 'Value' field contains 'Hello World!'. At the bottom, there are 'OK', 'Cancel', and 'Evaluation Mode' buttons.

Language: StructuredExpression

Body:

- Body
  - Script in JavaScript
    - arg1 = "Hello World!"
    - Create parameter... (1)
    - Create operation...

Script::arg1

Edit Use as... Reset Remove (2)

Operation Name: String1

Parameter Name: arg1

Value: Hello World!

Standard OK Cancel Evaluation Mode