

new("multiverse")
creates a new, empty object which will contain the multiverse analysis

M

code = `_empty_`
parameters = `_empty_list_`
conditions = `_empty_list_`
m. table = `empty_data.frame_`
current assgn = `_empty_`

branch()
If a certain step in the analysis can be performed in multiple different ways, use this to assign a parameter to this step, and define the different analysis options for this parameter.

inside()
is used to add code to the multiverse. This code can include different analysis pathways at any step of the analysis (see ``branch``). This code is captured, not executed.

M

code = `<expression>`
parameters = `_empty_list_`
conditions = `_empty_list_`
m. table = `empty_data.frame_`
current assgn = `_empty_`

parse_multiverse()
inspects the code, and identifies the parameters and options declared. It then updates the corresponding slots of the multiverse object.

list of all **conditions** declared in the code. When value(s) of parameter are dependent on the value(s) of another parameter, it is captured as conditions.

list of all **parameters** declared in the code (say, **n**). Each parameter, p_i ($i = 1 \dots n$) has k_i options associated with it

get_code()
takes as input a multiverse object, and a list of options for each parameter. It returns the code for a single analysis, which can then be evaluated.

This function can be called after `parse_multiverse()`. It is also called internally by `execute_multiverse()`.

M

code = `<expression>`
parameters = `<list[n]>`
conditions = `<list>`
m. table = `<data.frame [n+1 x q]>`
current assgn = `<list[n]>`

list of one set of options for each parameter; corresponds to the first row of the multiverse table

execute_multiverse()
execute_default()

M

code = `<expression>`
parameters = `<list[n]>`
conditions = `<list>`
m. table = `<data.frame [n+3 x q]>`
current assgn = `<list[n]>`

executes default analysis code

creates code for single analysis using default options for each parameter; executes the code (in global environment, so that output can be inspected).

multiverse table
A table, where each row corresponds to one analysis in the multiverse. Each column is a parameter, and each row is a unique combination of the options for each parameter; one column stores as a list the options for each parameter. Thus, $\text{dim}(\text{m. table}) = n + 1$ columns and q rows, where, $q = \prod_{i=1}^n k_i$

two columns are added: ``universe code`` and ``universe environment``.

	parameters ($i = 1 \dots n$) [n columns]					parameter assignment list	universe code	universe environment
1	1	Each row is a list of values for each parameter.	Each row is an expression for a single analysis.	Each row is an environment where the analysis can be run, and the result inspected.
2	1			
.			
.			
k_1	1			
1	2			
2	2			
.			
.			
.			
.			
.			
.			
.			
k_2			