

# Map operator

Performs a mapping of incoming attributes to out-coming attributes using map functions. Odysseus also provides a wide range of [mapping functions](#).

*Hint: Map is stateless. To used Map in a statebased fashion see: [StateMap](#)*

## Parameter

- **expressions**: A list of expressions to map multiple incoming attribute values to out-coming attributes. Optional each expression can have a name (in this case use ['expression', 'expressionName'])
- **threads**: Number of threads used to process the expressions simultaneous. A positive number greater than 1 indicates the fixed number of threads, a value of 0 or 1 disables threading, and a negative number estimates the number of threads based on the number of expressions and the available processors.
- **allownull**: If set to true (default) and an error occurs in calculation a null value is added to the element. Else the element is skipped and no output is produced. Default is true.
- **evaluateOnPunctuation**: If set to true, map will also create an output (with the last read element) when it receives a punctuation.
- **supressErrors**: If set to true calculation errors will not appear in log or console. Could be helpful in scenarios where null values are allowed.
- **keepAttributes**: If set to true, the input object will be copied to the output object. Remark: Not usable in keyvalue, here you should use \$ instead
- **removeAttribtues**: A List of attributes, which will be removed from the input object. Only valid together with keepAttributes.

## Example

### PQL

#### Map Operator

```
output = MAP({
  expressions = [
    ['auction_id * 5', 'AuctionMult5'],
    ['sqrt(auction_id)']
  ], input)
```

### CQL

#### Map Operator

```
SELECT auction_id * 5, sqrt(auction_id) FROM input
```