select

Syntax

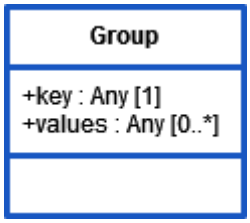
```
select each from anArray where <where-expression> group
by <group-by-expression>
select first from anArray where <where-expression> group
by <group-by-expression>
select last from anArray where <where-expression> group
by <group-by-expression>
select distinct <distinct-expression> from anArray where
<where-expression>
```

Semantics

Selects array items by evaluating a boolean where-expression for each array element and optionally grouping it. In all expressions the array element is identified by the keyword element. The optional keyword `each` indicates that the select statement returns an **Array** containing the filtered and grouped elements. If only the first or last array element are to be selected use `first` respectively `last`. If an array of unique values is to be returned use the keyword `distinct`.

> ⚠ Please note that `select` only selects the element references (not the content). The elements of the target array will reference to the same elements as the selected elements from the source array. Changes to an element of the target array will also change the related element in the source array.

Substitutables

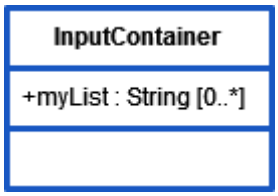
`anArray`	Can be any variable or object attribute having the type **Array**.
`<where-expression>`	This is can be any expression evaluating to **Boolean** and containing the keyword `element`. The where-clause (`where <where-expression>`) is mandatory for all `select` statements except for `select distinct`.
`<group-by-expression>`	This is can be any expression containing the keyword element and returning any simple type except **Blob**. The resulting (grouped) array is an array of type **Group**, where **values** contains the array elements that belong into that group.  The group-by clause (`group by <group-by-expression>`) is optional. The group-by clause must not be used in conjunction with the distinct clause (`distinct <distinct-expression>`).
`<distinct-expression>`	This is can be any expression containing the keyword element returning any simple type except **Blob**. The distinct-clause (`distinct <distinct-expression>`) must not be used in conjunction with the group-by clause.

Related Pages:

- apply Statement
- reduce Statement
- sort Statement

Examples

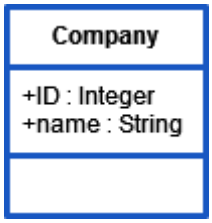
The examples below use an instance of **InputContainer** as input




```
set filteredElements = select each from inputContainer.
myList where element = "a";
set firstFilteredElement = select first from
inputContainer.myList where element = "a";
set filteredElements = select each from inputContainer.
myList where element like "a.+";
set lastFilteredElement = select last from inputContainer.
myList where element like "a.+";
```

Please note that `select` only selects the element references (not the content), so e.g. the elements of **filteredElements** will reference to the same elements as the selected elements from **inputContainer.myList** . Changes to an element of **filteredElements** will also change the related element in **inputContainer.myList** .

The examples below use an array of **Company** objects as input:




```
set distinctArray = select distinct element.ID from
companyArray;
set filteredDistinctArray = select distinct element.ID.
convertToString().concat(":", element.Name)
        from companyArray
        where element.Name.startsWith("M");
set groupedArray = select each from companyArray group by
element.ID;
set filteredAndGroupedArray =
select each from companyArray
        where element.Name.startsWith("M")
        group by element.ID.convertToString().concat(":",
element.Name);
```