where

Last Modified on 25/05/2020 1:45 pm AEST

The **where** keyword filters a record list by applying a yes/no condition.

Keyword

```
list where condition
```

Arguments

Argument	Data Type
list	record list
condition	single-value bool (true/false)
result	record list

Comments

A list calculation (such as requesting all records of a particular type, or following a to-many relationship) is provided to the left of the **where** keyword. A filtering calculation, typically some sort of comparison, is then provided to the right.

The filtering calculation must be a single-value calculation, not a list calculation. For example, you can filter a list of Person records by testing whether their name is equal to some value, but not (directly) by testing whether they have a task that is named some value, because a person could have multiple tasks, which would therefore result in multiple true/false results for each person. Note, however, that the any and every functions can combine multiple yes/no results back into a single result.

Use of true/false fields

The **where** keyword is typically used with a comparison such as [Field] = 'value'. However, any calculation that returns a true/false function can be provided to the right-hand side of a the **where** keyword, including the name of a bool true/false field. For example:

```
all([Task]) where [Is completed]
```

Context

The right-hand sorting calculations are evaluated with respect to the records returned by the left-hand side. For example, if you refer to the [Name] field on the right-hand side, then it refers to the name field of the record list being filtered, not the name field of some other parent record that may be applicable to the calculation as a whole.

If it is necessary to refer to the parent record, then first assign the context to a variable and then refer to the variable within the **where** condition. For example:

```
let parentRecordDueDate = [Project due date]
select
count( [Tasks] where [Estimated task completion] > parentRecordDueDate )
```

Examples

```
[Incidents] where State='NSW'
```

```
all([Person]) where any(not [Assigned Tasks].[Is complete])
```