



Rocket Uniface Library 10.4

Conditions

A condition is a logical expression that evaluates to a boolean value. It is typically used in conditional processing initiated by ProcScript statements such as **if**, **repeat**, **selectcase**, and **while**.

For example:

```
if ($STR$ != "&" & $NR$)
...
endif
```



Note: Parentheses around the conditional expression are optional.

In Uniface, every expression has a boolean representation. As a consequence, any expression can be used as a condition. Uniface performs implicit data type conversion if the expression is of a data type other than Boolean.

When testing conditions, you should not rely on implicit type conversion to use strings as if they were Booleans. For example, in the following conditions, it is not obvious how *str1* and *str2* should be interpreted:

- `if (str1) ...`
- `if (bTrue & str2) ...`

Instead, you should always explicitly state what you are testing for. For example:

- `if (str1 = "true")`
- `if (bTrue & str2 != $syntax("f*"))`

Example: Conditions

The following ProcScript examples show how conditions can be used in ProcScript:

- A condition in combination with the **if** ProcScript statement, where *myCondition* should resolve to a Boolean value:

```
if (myCondition) call doIt
```

- Two expressions that are used in combination with a logical operator:

```
if (myVar = "My expression") call doIt
```

- A condition in combination with a **if-else** ProcScript block:

```
if (myVar1 > myVar2)
  call doIt
else
  call doSomethingElse
endif
```

Related concepts

[Type Conversion](#)

[Operators](#)

[if](#)

[repeat](#)

[selectcase](#)

[while](#)