





Type Description Regular expression Specific values (2) 0.0, 0.1f, etc. \W(-)?[0-9]+(,[0-9]+)\*((\.[0-9]+)?[a-z]\*)\W Member value of objects, e.g., Location.x \W(^(java\.|javax\.|org\.))?([A-Za-z\_]+\w+\.)+[a-z\_]+[a-z0-9\_]\*[^\.A-Za-z0-9\_] Class methods and static members (4) class methods, e.g., ClassA.func(Param1) \W[A-Za-z\_]+[A-Za-z\_0-9]\*(\.[A-Za-z\_]+[A-Za-z\_0-9])\*(#[A-Za-z\_]+[A-Za-z\_0-9]\*)?\([^()]\*\)\W Static member, e.g., Desktop.Action#OPEN \W([A-Za-z\_]+[A-Za-z\_0-9]\*(\.[A-Za-z\_]+[A-Za-z\_0-9])\*)?(#[A-Za-z\_]+[A-Za-z\_0-9]\*)[^A-Za-z0-9\_()] All upper case \W(\w+\.)\*([A-Z]+\_)\*[A-Z]+\W Class name \W([A-Za-z\_]+\w+\.)\*[A-Za-z\_]\*[A-Z]+\w+[^\.A-Za-z0-9\_] Expressions (11) A - B \W\w+((\s+-)|(-\s+)|(\s+-\s+))\w+\W A + B \W\w+\s\*\+\s\*\w+\W A \* B \W\w+\s\*\\*\s\*\w+\W A..B \W\(?\s\*\w+\s\*\)?\s\*\.\s\*\.\s\*\(?\s\*\w+\s\*\)?\W [A, B] \W\[\s\*\w+\s\*,\s\*\w+\s\*\]\W

[A..B] \W\[\s\*\w+\s\*(\.\s\*\.\s\*)\s\*\w+\s\*\]\W A <\<= B <\<= C \W\w+\s\*<=?\s\*\w+\s\*<=?\s\*\w+\W A >\>= B >\>= C \W\w+\s\*>=?\s\*\w+\s\*>=?\s\*\w+\W From A to B \W(from\s+)?\w+\s+to\s+\w+\W A != B \W\w+\s\*!=\s\*\w+\W Enumeration expression \W (\s\*\w+\s\*)(,\s\*\w+\s\*)+,?\s\*or\s\*\w+\W

Type Rule FRAG statements (2) FRAG -> SBAR FRAG

–

> IN + NP SBAR statements (2) SBAR

–

> IN(if) + S SBAR

–

> SBAR + CC + SBAR S statements (2) S -> S + CC + S S -> NP + VP VP statements (3) VP-> VP+CC + VP VP-> VBZ+ADJP/UCP VP-> VBZ + ADJP +

？

NP statements (1) NP -> NP + CC + NP ADJP statements (1) ADJP -> ? + CC + ? UCP statements (1) UCP

–

> ? + CC + ?

Nullness not allowed

@exception; @throws 1.

[something] be/equals null 2.

[something] be equal/equivalent to null 3.

[something1] or [something2] be/equals null 4.

[something1] or [something2] be equal/equivalent to null 5.

[something]

parameter

be null 6.

The specified [something] be null 7.

Any/none of [something] be null 8.

Either/neither/any/all/both/none

parameter s)

be/equals null 9.

Either/neither/any/all/both/none

parameter s)

be equal/equivalent null 10.

Either/neither [something1] or/nor [something2] be/equals null 11.

Both [something1] and [something2] be null 12.

The [type] be null 13.

[parameter phrase] be/equals null 14.

[parameter phrase] be equal/equivalent null 15.

[something

]’

s value be/equals null 16.

[something]’s

value be equal/equivalent null 17.

Value of [something] be/equals null 18.

Value of [something] be equal/equivalent null

@param

19.

[something] can not be null 20.

Non-null [something]

Nullness allowed

@param 1.

[parameter] can/could be null 2.

[parameter] may be null 3.

[parameter] can/could be equivalent/equal to null 4.

[something]

parameter

can/could be null 5.

[something]

parameter

may be null 6.

[something]

parameter

can/could be equivalent/equal to null 7.

Value of [something] can/could be null 8.

Value of [something] may be null 9.

Value of [something] can/could be equal/equivalent to null 10.

[something] null be ignored 11.

[something] … or null

Type restriction

@exception; @throws 1.

[something] be {not} [SpecClassName] 2.

[something1] or/and [something2] be {not} [SpecClassName]

3.

[something] be {not} equivalent/equal to [SpecClassName] 4.

[something1] or/and [something2] be {not} equivalent/equal to [SpecClassName] 5.

[something] be {not} a class derived from [SpecClassName] 6.

[something1] or/and [something2] be {not} a class derived from [SpecClassName] 7.

[something] be {not} an instance of [type] 8.

[something1] or/and [something2] be {not} an instance of [type] 9.

[something] be {not} an instance of [type1] or [type2] 10.

[something1] or/and [something2] be {not} an instance of [type1] or [type2]

Range limitation

@exception; @throws 1.

[something] >/</= [value] 2.

[something] be {not} less/greater/larger/equal/equivalent than/to [value] 3.

[something] equals [value] 4.

[something1] or/and [something2] be {not} less/greater/larger/equal/equivalent than/to [value] 5.

Computing [expression] be {not} less/greater/larger/equal/equivalent than/to [value] 6.

Computing either [expression1] or [expression2] be {not} less/greater/larger/equal/equivalent than [value] 7.

Product/sum of [something1] and [something2] be {not} less/greater/larger/equal/equivalent than/to [value] 8.

[something] be {not} negative/positive/false/true

9.

[something1] or/and [something2] be {not} negative/positive/false/true 10.

[something1] and [something2] be {not} the same 11.

[something1] equals [something2] 12.

[something] be {not} in/out of/outside of range [range value] 13.

[something] be {not} in/out of bounds 14.

[something] be {not} [value] 15.

[range expression] (only the expression

，

like ) 16.

[something] be {not} between [value1] and [value2] 17.

[something] be {not} [value set] 18.

[something] be {not} one of [value set] 19.

[something] be {not} one of following: [value set] 20.

[something] be {not} one of supported data, which are [value set] @param: 21.

[something] can/must {not} be negative/positive/non-negative/non-positive 22.

[something] must be greater/less/larger than [value] 23.

[something] be greater/less/larger than [value]