weeJava Specification

weeJava is defined as a subset of the Java programming language we learned. It allows the following operations, symbols, keywords and identifiers etc., which keep their original meaning as in Java.

Operators

weeJava supports the following operators.

Name	Common name	Character sequence
OP_MULTIPLY	multiply	*
OP_DIVIDE	divide	/
OP_MOD	mod	%
OP_ADD	plus	+
OP_SUBTRACT	minus	-
OP_LESS	less than	<
OP_LESSEQUAL	less than or equal	<=
OP_GREATER	greater than	>
OP_GREATEREQUAL	greater than or equal	>=
OP_EQUAL	equal	==
OP_NOTEQUAL	not equal	!=
OP_NOT	unary not	!
OP_ASSIGN	assignment	=
OP_AND	logical and	&&
OP_OR	logical or	||
OP_DOT	Dot operator	.

Symbols

Here are symbols that are allowed in **weeJava**.

Name	Common name	Character
LEFT_PAREN	left parenthesis	(
RIGHT_PAREN	right parenthesis	)
LEFT_BRACE	left brace	{
RIGHT_BRACE	right brace	}
LEFT_BRACKET	left bracket	[
RIGHT_BRACKET	right bracket	]
SEMICOLON	semi-colon	;
COLON	colon	:
COMMA	comma	,

Keywords

The following table lists all reserved words in **weeJava**. Note that **weeJava** does not support data types such as float, short or enum.

Name	String
KEYWORD_IF	if
KEYWORD_ELSE	else
KEYWORD_WHILE	while
KEYWORD_RETURN	return
KEYWORD_MAIN	main
KEYWORD_INT	int
KEYWORD_DOUBLE	double
KEYWORD_BOOLEAN	boolean
KEYWORD_STRING	String
KEYWORD_PUBLIC	public
KEYWORD_CLASS	class
KEYWORD_VOID	void
KEYWORD_FOR	for
KEYWORD_CASE	case
KEYWORD_SWITCH	switch
KEYWORD_STATIC	static
KEYWORD_BREAK	break
KEYWORD_CONTINUE	continue
KEYWORD_DEFAULT	default

Identifiers and literals

Identifiers can be variable/class names and method names. Literals can be e.g. 19, 30.6, “abcdefg”, true.

Name	Common name	Value
IDENTIFIER	Variable names	as is
INTEGER	integer literal	as is, interpreted as a number
DOUBLE	double literal	e.g. 1.234
STRING	string literal	the characters with the double quotes
BOOLEAN	boolean	true/false