

It reads left-to-right, or right-to-left if ending in :.

It requires a receiving object, e.g.

```
Console.println 10
```

```
not println 10
```

Multi-arg calling is done with parens:

```
s.indexOf ('0', 5)
```

+, -, !, ~

You can prepend unary_ to any of the four.

- 'cymbal' is expanded to `Symbol("cymbal")`.
- They must be alphanumeric.
- Values are interned. Symbol literal twice will refer to the exact same symbol object.

Absolutely nothing.

They are just methods that take no arguments when called without a dot or parens.

```
println("""| Some really long  
          | string""".stripMargin)
```

It's always left-to-right, regardless of associativity.

By-name parameters.

Method precedence is done by the first letter of the method name. Assignment operators are done by the last letter (ending in =).

.

+ -

:

= !

< >

&

^

|

(all letters)

(all assignment operators)

,