

# LISP - MACROS

Macros allow you to extend the syntax of standard LISP.

Technically, a macro is a function that takes an s-expression as arguments and returns a LISP form, which is then evaluated.

## Defining a Macro

In LISP, a named macro is defined using another macro named **defmacro**. Syntax for defining a macro is:

```
(defmacro macro-name (parameter-list))
"Optional documentation string."
body-form
```

The macro definition consists of the name of the macro, a parameter list, an optional documentation string, and a body of Lisp expressions that defines the job to be performed by the macro.

## Example

Let us write a simple macro named `setTo10`, which will take a number and set its value to 10.

Create new source code file named `main.lisp` and type the following code in it.

```
(defmacro setTo10(num)
  (setq num 10)(print num))
  (setq x 25)
  (print x)
  (setTo10 x)
```

When you click the Execute button, or type `Ctrl+E`, LISP executes it immediately and the result returned is:

```
25
10
```