



▶ Dynamic Functions

▶ Multi-Line Functions

## Multi-Line Dynamic Functions

The single expression which provides the result of the Dynamic Function may be preceded by any number of assignment statements. Each such statement introduces a name which is local to the function.

For example in the following, the expressions `sum←` and `num←` create **local** variables `sum` and `num`.

```
mean←{      A Arithmetic mean
  sum←+/%  A Sum of elements
  num←P%   A Number of elements
  sum÷num  A Mean
}
```

Note that Dynamic Functions may be commented in the usual way using `␣`.

When the interpreter encounters a local definition, a new local name is created. The name is shadowed dynamically exactly as if the assignment had been preceded by: `␣shadow name ⋄`.

It is **important** to note the distinction between the two types of statement above. There can be **many** assignment statements, each introducing a new local variable, but only a **single** expression where the result is not assigned. As soon as the interpreter encounters such an expression, it is evaluated and the result returned immediately as the result of the function.

For example, in the following,

```
mean←{      A Arithmetic mean
  sum←+/%  A Sum of elements
  num←P%   A Number of elements

  sum,num  A Attempt to show sum,num (wrong)!

  sum÷num  A ... and return result.
}
```

As soon as the interpreter encounters the expression `sum, num`, the function terminates with the two element result (`sum, num`) and the following line is not evaluated.

### Dyalog Ltd

South Barn  
Minchens Court  
Minchens Lane  
Bramley  
Hampshire  
RG26 5BH  
United Kingdom

**Phone:**

+ 44 (0) 1256 830 030

**Fax:**

+ 44 (0) 1256 830 031

**e-mail:**

sales@dyalog.com



Dynamic Functions

Multi-Line Functions

## Multi-Line Dynamic Functions

To display arrays to the session from within a Dynamic function, you can use the explicit display forms `□←` or `□←` as in:

```
mean←{           A Arithmetic mean
  sum←+/w        A Sum of elements
  num←Pw         A Number of elements

  □←sum,num      A show sum,num

  sum÷num        A ... and return result.
}
```

Note that local definitions can be used to specify local nested Dynamic Functions:

```
rms←{           A Root Mean Square
  root←(w=0.5)  A √ Square root
  mean←((+/w)÷Pw) A √ Mean
  square←(w×w)  A √ Square

  root mean square w
}
```

## Default Left Argument

The special syntax: `α←expr` is used to give a default value to the left argument if a Dynamic Function is called monadically. For example:

```
root←{           A αth root
  α←2            A default to sqrt
  w=÷α
}
```

The expression to the right of `α←` is evaluated *only* if its Dynamic Function is called with no left argument.

### Dyalog Ltd

South Barn  
Minchens Court  
Minchens Lane  
Bramley  
Hampshire  
RG26 5BH  
United Kingdom

**Phone:**

+ 44 (0) 1256 830 030

**Fax:**

+ 44 (0) 1256 830 031

**e-mail:**

sales@dyalog.com