Version 15.17R, X3J13/94-101R.

Fri 12-Aug-1994 6:35pm EDT

**Description:**

If it is *non-nil*, it specifies the right margin (as *integer* number of *ems*) to use when the *pretty printer* is making layout decisions.

If it is **nil**, the right margin is taken to be the maximum line length such that output can be displayed without wraparound or truncation. If this cannot be determined, an *implementation-dependent* value is used.

**Notes:**

This measure is in units of *ems* in order to be compatible with *implementation-defined* variable-width fonts while still not requiring the language to provide support for fonts.

**print-not-readable** *Condition Type*

**Class Precedence List:**

**print-not-readable**, **error**, **serious-condition**, **condition**, **t**

**Description:**

The *type* **print-not-readable** consists of error conditions that occur during output while **\*print-readably\*** is *true*, as a result of attempting to write a printed representation with the *Lisp printer* that would not be correctly read back with the *Lisp reader* . The object which could not be printed is initialized by the :object initialization argument to **make-condition**, and is *accessed* by the *function* **print-not-readable-object**.

**See Also:**

**print-not-readable-object**

**print-not-readable-object** *Function*

**Syntax:**

**print-not-readable-object** *condition ! object*

**Arguments and Values:**

*condition*—a *condition* of *type* **print-not-readable**.

*object*—an *object*.

**Description:**

Returns the *object* that could not be printed readably in the situation represented by *condition*. Printer **22–79**

Version 15.17R, X3J13/94-101R.

Fri 12-Aug-1994 6:35pm EDT

**See Also:**

**print-not-readable**, Chapter 9 (Conditions)

**format** *Function*

**Syntax:**

**format** *destination control-string* &rest *args ! result*

**Arguments and Values:**

*destination*—**nil**, **t**, a *stream*, or a *string* with a *fill pointer* .

*control-string*—a *format control*.

*args*—*format arguments* for *control-string*.

*result*—if *destination* is *non-nil*, then **nil**; otherwise, a *string*.

**Description:**

**format** produces formatted output by outputting the characters of *control-string* and observing that a *tilde* introduces a directive. The character after the tilde, possibly preceded by prefix parameters and modifiers, specifies what kind of formatting is desired. Most directives use one or more elements of *args* to create their output.

If *destination* is a *string*, a *stream*, or **t**, then the *result* is **nil**. Otherwise, the *result* is a *string* containing the ‘output.’

**format** is useful for producing nicely formatted text, producing good-looking messages, and so on. **format** can generate and return a *string* or output to *destination*.

For details on how the *control-string* is interpreted, see Section 22.3 (Formatted Output).

**Aected By:**

**\*standard-output\***, **\*print-escape\***, **\*print-radix\***, **\*print-base\***, **\*print-circle\***, **\*print-pretty\***, **\*print-level\***, **\*print-length\***, **\*print-case\***, **\*print-gensym\***, **\*print-array\***.

**Exceptional Situations:**

If *destination* is a *string* with a *fill pointer* , the consequences are undefined if destructive modifications are performed directly on the *string* during the *dynamic extent* of the call.

**See Also:**

**write**, Section 13.1.10 (Documentation of Implementation-Defined Scripts)

**22–80** Programming Language—Common Lisp