

LISP - CHARACTERS

http://www.tutorialspoint.com/lisp/lisp_characters.htm

Copyright © tutorialspoint.com

In LISP, characters are represented as data objects of type **character**.

You can denote a character object preceding `#\` before the character itself. For example, `#\a` means the character a.

Space and other special characters can be denoted by preceding `#\` before the name of the character. For example, `#\SPACE` represents the space character.

The following example demonstrates this:

Example

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

```
(write 'a)
(terpri)
(write #\a)
(terpri)
(write-char #\a)
(terpri)
(write-char 'a)
```

When you execute the code, it returns the following result:

```
A
#\a
a
*** - WRITE-CHAR: argument A is not a character
```

Special Characters

Common LISP allows using the following special characters in your code. They are called the semi-standard characters.

- `#\Backspace`
- `#\Tab`
- `#\Linefeed`
- `#\Page`
- `#\Return`
- `#\Rubout`

Character Comparison Functions

Numeric comparison functions and operators, like, `<` and `>` do not work on characters. Common LISP provides other two sets of functions for comparing characters in your code.

One set is case-sensitive and the other case-insensitive.

The following table provides the functions:

Case Sensitive Functions	Case-insensitive Functions	Description
<code>char=</code>	<code>char-equal</code>	Checks if the values of the operands are all equal or not, if yes then condition becomes true.

char/=	char-not-equal	Checks if the values of the operands are all different or not, if values are not equal then condition becomes true.
char<#60;	char-lessp	Checks if the values of the operands are monotonically decreasing.
char>	char-greaterp	Checks if the values of the operands are monotonically increasing.
char<#60;=	char-not-greaterp	Checks if the value of any left operand is greater than or equal to the value of next right operand, if yes then condition becomes true.
char>=	char-not-lessp	Checks if the value of any left operand is less than or equal to the value of its right operand, if yes then condition becomes true.

Example

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

```
; case-sensitive comparison
(write (char= #\a #\b))
(terpri)
(write (char= #\a #\a))
(terpri)
(write (char= #\a #\A))
(terpri)

; case-insensitive comparison
(write (char-equal #\a #\A))
(terpri)
(write (char-equal #\a #\b))
(terpri)
(write (char-lessp #\a #\b #\c))
(terpri)
(write (char-greaterp #\a #\b #\c))
```

When you execute the code, it returns the following result:

```
NIL
T
NIL
T
NIL
T
NIL
```