

NAME

localeconv - natural language formatting for C

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

#include <locale.h>

struct lconv *

localeconv(*void*);

#include <xlocale.h>

struct lconv *

localeconv_l(*locale_t locale*);

DESCRIPTION

The **localeconv**() function returns a pointer to a structure which provides parameters for formatting numbers, especially currency values:

```
struct lconv {
    char    *decimal_point;
    char    *thousands_sep;
    char    *grouping;
    char    *int_curr_symbol;
    char    *currency_symbol;
    char    *mon_decimal_point;
    char    *mon_thousands_sep;
    char    *mon_grouping;
    char    *positive_sign;
    char    *negative_sign;
    char    int_frac_digits;
    char    frac_digits;
    char    p_cs_precedes;
    char    p_sep_by_space;
    char    n_cs_precedes;
    char    n_sep_by_space;
    char    p_sign_posn;
    char    n_sign_posn;
```

```
    char    int_p_cs_precedes;
    char    int_n_cs_precedes;
    char    int_p_sep_by_space;
    char    int_n_sep_by_space;
    char    int_p_sign_posn;
    char    int_n_sign_posn;
};
```

The individual fields have the following meanings:

<i>decimal_point</i>	The decimal point character, except for currency values, cannot be an empty string.
<i>thousands_sep</i>	The separator between groups of digits before the decimal point, except for currency values.
<i>grouping</i>	The sizes of the groups of digits, except for currency values. This is a pointer to a vector of integers, each of size <i>char</i> , representing group size from low order digit groups to high order (right to left). The list may be terminated with 0 or CHAR_MAX. If the list is terminated with 0, the last group size before the 0 is repeated to account for all the digits. If the list is terminated with CHAR_MAX, no more grouping is performed.
<i>int_curr_symbol</i>	The standardized international currency symbol.
<i>currency_symbol</i>	The local currency symbol.
<i>mon_decimal_point</i>	The decimal point character for currency values.
<i>mon_thousands_sep</i>	The separator for digit groups in currency values.
<i>mon_grouping</i>	Like <i>grouping</i> but for currency values.
<i>positive_sign</i>	The character used to denote nonnegative currency values, usually the empty string.
<i>negative_sign</i>	The character used to denote negative currency values, usually a minus sign.
<i>int_frac_digits</i>	The number of digits after the decimal point in an international-style currency value.

<i>frac_digits</i>	The number of digits after the decimal point in the local style for currency values.
<i>p_cs_precedes</i>	1 if the currency symbol precedes the currency value for nonnegative values, 0 if it follows.
<i>p_sep_by_space</i>	1 if a space is inserted between the currency symbol and the currency value for nonnegative values, 0 otherwise.
<i>n_cs_precedes</i>	Like <i>p_cs_precedes</i> but for negative values.
<i>n_sep_by_space</i>	Like <i>p_sep_by_space</i> but for negative values.
<i>p_sign_posn</i>	The location of the <i>positive_sign</i> with respect to a nonnegative quantity and the <i>currency_symbol</i> , coded as follows: <ul style="list-style-type: none"> 0 Parentheses around the entire string. 1 Before the string. 2 After the string. 3 Just before <i>currency_symbol</i>. 4 Just after <i>currency_symbol</i>.
<i>n_sign_posn</i>	Like <i>p_sign_posn</i> but for negative currency values.
<i>int_p_cs_precedes</i>	Same as <i>p_cs_precedes</i> , but for internationally formatted monetary quantities.
<i>int_n_cs_precedes</i>	Same as <i>n_cs_precedes</i> , but for internationally formatted monetary quantities.
<i>int_p_sep_by_space</i>	Same as <i>p_sep_by_space</i> , but for internationally formatted monetary quantities.
<i>int_n_sep_by_space</i>	Same as <i>n_sep_by_space</i> , but for internationally formatted monetary quantities.
<i>int_p_sign_posn</i>	Same as <i>p_sign_posn</i> , but for internationally formatted monetary quantities.
<i>int_n_sign_posn</i>	Same as <i>n_sign_posn</i> , but for internationally formatted monetary quantities.

Unless mentioned above, an empty string as a value for a field indicates a zero length result or a value that is not in the current locale. A CHAR_MAX result similarly denotes an unavailable value.

The **localeconv_l()** function takes an explicit locale parameter. For more information, see `xlocale(3)`.

RETURN VALUES

The **localeconv()** function returns a pointer to a static object which may be altered by later calls to `setlocale(3)` or **localeconv()**. The return value for **localeconv_l()** is stored with the locale. It will remain valid until a subsequent call to `freelocale(3)`. If a thread-local locale is in effect then the return value from **localeconv()** will remain valid until the locale is destroyed.

ERRORS

No errors are defined.

SEE ALSO

`setlocale(3)`, `strfmon(3)`

STANDARDS

The **localeconv()** function conforms to ISO/IEC 9899:1999 ("ISO C99").

HISTORY

The **localeconv()** function first appeared in 4.4BSD.