:mod:`curses` --- Terminal handling for character-cell displays

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 4)

Unknown directive type "module".

```
.. module:: curses
   :synopsis: An interface to the curses library, providing portable
              terminal handling.
   :platform: Unix
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 9)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Moshe Zadka <moshez@zadka.site.co.il>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 10)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Eric Raymond <esr@thyrsus.com>
```

The :mod:`curses` module provides an interface to the curses library, the de-facto standard for portable advanced terminal handling.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 14); *backlink*

Unknown interpreted text role "mod".

While curses is most widely used in the Unix environment, versions are available for Windows, DOS, and possibly other systems as well. This extension module is designed to match the API of ncurses, an open-source curses library hosted on Linux and the BSD variants of Unix.

Note

Whenever the documentation mentions a *character* it can be specified as an integer, a one-character Unicode string or a one-byte byte string.

Whenever the documentation mentions a *character string* it can be specified as a Unicode string or a byte string.

Note

Since version 5.4, the ncurses library decides how to interpret non-ASCII data using the `nl_langinfo` function. That means that you have to call :func:`locale.setlocale` in the application and encode Unicode strings using one of the system's available encodings. This example uses the system's default encoding:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 32); *backlink*

Unknown interpreted text role "func".

```
import locale
locale.setlocale(locale.LC_ALL, '')
code = locale.getpreferredencoding()
```

Then use *code* as the encoding for :meth:`str.encode` calls.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 42); *backlink*

Unknown interpreted text role "meth".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 44)

Unknown directive type "seealso".

```
.. seealso::

   Module :mod:`curses.ascii`
      Utilities for working with ASCII characters, regardless of your locale settings.

   Module :mod:`curses.panel`
      A panel stack extension that adds depth to  curses windows.

   Module :mod:`curses.textpad`
      Editable text widget for curses supporting  :program:`Emacs`\ -like bindings.

   :ref:`curses-howto`
      Tutorial material on using curses with Python, by Andrew Kuchling and Eric
      Raymond.

   The :source:`Tools/demo/` directory in the Python source distribution contains
   some example programs using the curses bindings provided by this module.
```

Functions

The module :mod:`curses` defines the following exception:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 68); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 71)

Unknown directive type "exception".

```
.. exception:: error

   Exception raised when a curses library function returns an error.
```

Note

Whenever *x* or *y* arguments to a function or a method are optional, they default to the current cursor location. Whenever *attr* is optional, it defaults to :const:`A_NORMAL`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 77); *backlink*

Unknown interpreted text role "const".

The module :mod:`curses` defines the following functions:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 81); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 84)

Unknown directive type "function".

```
.. function:: baudrate()

   Return the output speed of the terminal in bits per second.  On software
   terminal emulators it will have a fixed high value. Included for historical
   reasons; in former times, it was used to  write output loops for time delays and
   occasionally to change interfaces depending on the line speed.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 92)

Unknown directive type "function".

```
.. function:: beep()

   Emit a short attention sound.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 97)

Unknown directive type "function".

```
.. function:: can_change_color()

   Return ``True`` or ``False``, depending on whether the programmer can change the colors
   displayed by the terminal.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 103)

Unknown directive type "function".

```
.. function:: cbreak()

   Enter cbreak mode.  In cbreak mode (sometimes called "rare" mode) normal tty
   line buffering is turned off and characters are available to be read one by one.
   However, unlike raw mode, special characters (interrupt, quit, suspend, and flow
   control) retain their effects on the tty driver and calling program.  Calling
   first :func:`raw` then :func:`cbreak` leaves the terminal in cbreak mode.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 112)

Unknown directive type "function".

```
.. function:: color_content(color_number)

   Return the intensity of the red, green, and blue (RGB) components in the color
   *color_number*, which must be between ``0`` and ``COLORS - 1``.  Return a 3-tuple,
   containing the R,G,B values for the given color, which will be between
   ``0`` (no component) and ``1000`` (maximum amount of component).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 120)

Unknown directive type "function".

```
.. function:: color_pair(pair_number)

   Return the attribute value for displaying text in the specified color pair.
   Only the first 256 color pairs are supported. This
   attribute value can be combined with :const:`A_STANDOUT`, :const:`A_REVERSE`,
   and the other :const:`A_\*` attributes.  :func:`pair_number` is the counterpart
   to this function.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 129)

Unknown directive type "function".

```
.. function:: curs_set(visibility)

   Set the cursor state.  *visibility* can be set to ``0``, ``1``, or ``2``, for invisible,
   normal, or very visible.  If the terminal supports the visibility requested, return the
   previous cursor state; otherwise raise an exception.  On many
   terminals, the "visible" mode is an underline cursor and the "very visible" mode
   is a block cursor.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 138)

Unknown directive type "function".

```
.. function:: def_prog_mode()

   Save the current terminal mode as the "program" mode, the mode when the running
   program is using curses.  (Its counterpart is the "shell" mode, for when the
```

```
   program is not in curses.)  Subsequent calls to :func:`reset_prog_mode` will
   restore this mode.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 146)

Unknown directive type "function".

```
.. function:: def_shell_mode()

   Save the current terminal mode as the "shell" mode, the mode when the running
   program is not using curses.  (Its counterpart is the "program" mode, when the
   program is using curses capabilities.) Subsequent calls to
   :func:`reset_shell_mode` will restore this mode.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 154)

Unknown directive type "function".

```
.. function:: delay_output(ms)

   Insert an *ms* millisecond pause in output.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 159)

Unknown directive type "function".

```
.. function:: doupdate()

   Update the physical screen.  The curses library keeps two data structures, one
   representing the current physical screen contents and a virtual screen
   representing the desired next state.  The :func:`doupdate` ground updates the
   physical screen to match the virtual screen.

   The virtual screen may be updated by a :meth:`~window.noutrefresh` call after write
   operations such as :meth:`~window.addstr` have been performed on a window.  The normal
   :meth:`~window.refresh` call is simply :meth:`!noutrefresh` followed by :func:`!doupdate`;
   if you have to update multiple windows, you can speed performance and perhaps
   reduce screen flicker by issuing :meth:`!noutrefresh` calls on all windows,
   followed by a single :func:`!doupdate`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 174)

Unknown directive type "function".

```
.. function:: echo()

   Enter echo mode.  In echo mode, each character input is echoed to the screen as
   it is entered.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 180)

Unknown directive type "function".

```
.. function:: endwin()

   De-initialize the library, and return terminal to normal status.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 185)

Unknown directive type "function".

```
.. function:: erasechar()

   Return the user's current erase character as a one-byte bytes object.  Under Unix operating systems th
   is a property of the controlling tty of the curses program, and is not set by
   the curses library itself.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 192)

Unknown directive type "function".

```
.. function:: filter()

   The :func:`.filter` routine, if used, must be called before :func:`initscr` is
   called.  The effect is that, during those calls, :envvar:`LINES` is set to ``1``; the
   capabilities ``clear``, ``cup``, ``cud``, ``cud1``, ``cuu1``, ``cuu``, ``vpa`` are disabled; and the
   string is set to the value of ``cr``. The effect is that the cursor is confined to
   the current line, and so are screen updates.  This may be used for enabling
   character-at-a-time  line editing without touching the rest of the screen.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 202)

Unknown directive type "function".

```
.. function:: flash()

   Flash the screen.  That is, change it to reverse-video and then change it back
   in a short interval.  Some people prefer such as 'visible bell' to the audible
   attention signal produced by :func:`beep`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 209)

Unknown directive type "function".

```
.. function:: flushinp()

   Flush all input buffers.  This throws away any  typeahead  that  has been typed
   by the user and has not yet been processed by the program.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 215)

Unknown directive type "function".

```
.. function:: getmouse()

   After :meth:`~window.getch` returns :const:`KEY_MOUSE` to signal a mouse event, this
   method should be called to retrieve the queued mouse event, represented as a
   5-tuple ``(id, x, y, z, bstate)``. *id* is an ID value used to distinguish
   multiple devices, and *x*, *y*, *z* are the event's coordinates.  (*z* is
   currently unused.)  *bstate* is an integer value whose bits will be set to
   indicate the type of event, and will be the bitwise OR of one or more of the
   following constants, where *n* is the button number from 1 to 5:
   :const:`BUTTONn_PRESSED`, :const:`BUTTONn_RELEASED`, :const:`BUTTONn_CLICKED`,
   :const:`BUTTONn_DOUBLE_CLICKED`, :const:`BUTTONn_TRIPLE_CLICKED`,
   :const:`BUTTON_SHIFT`, :const:`BUTTON_CTRL`, :const:`BUTTON_ALT`.

   .. versionchanged:: 3.10
      The ``BUTTON5_*`` constants are now exposed if they are provided by the
      underlying curses library.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 233)

Unknown directive type "function".

```
.. function:: getsyx()

   Return the current coordinates of the virtual screen cursor as a tuple
   ``(y, x)``.  If :meth:`leaveok <window.leaveok>` is currently ``True``, then return ``(-1, -1)``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 239)

Unknown directive type "function".

```
.. function:: getwin(file)

   Read window related data stored in the file by an earlier :func:`putwin` call.
   The routine then creates and initializes a new window using that data, returning
   the new window object.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 246)

Unknown directive type "function".

```
.. function:: has_colors()

   Return ``True`` if the terminal can display colors; otherwise, return ``False``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 250)

Unknown directive type "function".

```
.. function:: has_extended_color_support()

   Return ``True`` if the module supports extended colors; otherwise, return
   ``False``. Extended color support allows more than 256 color pairs for
   terminals that support more than 16 colors (e.g. xterm-256color).

   Extended color support requires ncurses version 6.1 or later.

   .. versionadded:: 3.10
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 260)

Unknown directive type "function".

```
.. function:: has_ic()

   Return ``True`` if the terminal has insert- and delete-character capabilities.
   This function is included for historical reasons only, as all modern software
   terminal emulators have such capabilities.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 267)

Unknown directive type "function".

```
.. function:: has_il()

   Return ``True`` if the terminal has insert- and delete-line capabilities, or can
   simulate  them  using scrolling regions. This function is included for
   historical reasons only, as all modern software terminal emulators have such
   capabilities.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 275)

Unknown directive type "function".

```
.. function:: has_key(ch)

   Take a key value *ch*, and return ``True`` if the current terminal type recognizes
   a key with that value.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 281)

Unknown directive type "function".

```
.. function:: halfdelay(tenths)

   Used for half-delay mode, which is similar to cbreak mode in that characters
   typed by the user are immediately available to the program. However, after
   blocking for *tenths* tenths of seconds, raise an exception if nothing has
   been typed.  The value of *tenths* must be a number between ``1`` and ``255``.  Use
   :func:`nocbreak` to leave half-delay mode.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 290)

Unknown directive type "function".

```
.. function:: init_color(color_number, r, g, b)

   Change the definition of a color, taking the number of the color to be changed
   followed by three RGB values (for the amounts of red, green, and blue
   components).  The value of *color_number* must be between ``0`` and
   `COLORS - 1`.  Each of *r*, *g*, *b*, must be a value between ``0`` and
   ``1000``.  When :func:`init_color` is used, all occurrences of that color on the
   screen immediately change to the new definition.  This function is a no-op on
   most terminals; it is active only if :func:`can_change_color` returns ``True``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 301)

Unknown directive type "function".

```
.. function:: init_pair(pair_number, fg, bg)

   Change the definition of a color-pair.  It takes three arguments: the number of
   the color-pair to be changed, the foreground color number, and the background
   color number.  The value of *pair_number* must be between ``1`` and
   ``COLOR_PAIRS - 1`` (the ``0`` color pair is wired to white on black and cannot
   be changed).  The value of *fg* and *bg* arguments must be between ``0`` and
   ``COLORS - 1``, or, after calling :func:`use_default_colors`, ``-1``.
   If the color-pair was previously initialized, the screen is
   refreshed and all occurrences of that color-pair are changed to the new
   definition.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 314)

Unknown directive type "function".

```
.. function:: initscr()

   Initialize the library. Return a :ref:`window <curses-window-objects>` object
   which represents the whole screen.

   .. note::

      If there is an error opening the terminal, the underlying curses library may
      cause the interpreter to exit.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 325)

Unknown directive type "function".

```
.. function:: is_term_resized(nlines, ncols)

   Return ``True`` if :func:`resize_term` would modify the window structure,
   ``False`` otherwise.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 331)

Unknown directive type "function".

```
.. function:: isendwin()

   Return ``True`` if :func:`endwin` has been called (that is, the  curses library has
   been deinitialized).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 337)

Unknown directive type "function".

```
.. function:: keyname(k)

   Return the name of the key numbered *k* as a bytes object.  The name of a key generating printable
   ASCII character is the key's character.  The name of a control-key combination
   is a two-byte bytes object consisting of a caret (``b'^'``) followed by the corresponding
   printable ASCII character.  The name of an alt-key combination (128--255) is a
   bytes object consisting of the prefix ``b'M-'`` followed by the name of the corresponding
   ASCII character.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 347)

Unknown directive type "function".

```
.. function:: killchar()

   Return the user's current line kill character as a one-byte bytes object. Under Unix operating systems
   this is a property of the controlling tty of the curses program, and is not set
   by the curses library itself.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 354)

Unknown directive type "function".

```
.. function:: longname()

   Return a bytes object containing the terminfo long name field describing the current
   terminal.  The maximum length of a verbose description is 128 characters.  It is
   defined only after the call to :func:`initscr`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 361)

Unknown directive type "function".

```
.. function:: meta(flag)

   If *flag* is ``True``, allow 8-bit characters to be input.  If
   *flag* is ``False``,  allow only 7-bit chars.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 367)

Unknown directive type "function".

```
.. function:: mouseinterval(interval)

   Set the maximum time in milliseconds that can elapse between press and release
   events in order for them to be recognized as a click, and return the previous
   interval value.  The default value is 200 milliseconds, or one fifth of a second.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 374)

Unknown directive type "function".

```
.. function:: mousemask(mousemask)

   Set the mouse events to be reported, and return a tuple ``(availmask,
   oldmask)``.   *availmask* indicates which of the specified mouse events can be
   reported; on complete failure it returns ``0``.  *oldmask* is the previous value of
   the given window's mouse event mask.  If this function is never called, no mouse
   events are ever reported.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 383)

Unknown directive type "function".

```
.. function:: napms(ms)

   Sleep for *ms* milliseconds.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 388)

Unknown directive type "function".

```
.. function:: newpad(nlines, ncols)

   Create and return a pointer to a new pad data structure with the given number
   of lines and columns.  Return a pad as a window object.

   A pad is like a window, except that it is not restricted by the screen size, and
   is not necessarily associated with a particular part of the screen.  Pads can be
   used when a large window is needed, and only a part of the window will be on the
   screen at one time.  Automatic refreshes of pads (such as from scrolling or
   echoing of input) do not occur.  The :meth:`~window.refresh` and :meth:`~window.noutrefresh`
   methods of a pad require 6 arguments to specify the part of the pad to be
   displayed and the location on the screen to be used for the display. The
   arguments are *pminrow*, *pmincol*, *sminrow*, *smincol*, *smaxrow*, *smaxcol*; the *p*
   arguments refer to the upper left corner of the pad region to be displayed and
   the *s* arguments define a clipping box on the screen within which the pad region
   is to be displayed.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 406)

Unknown directive type "function".

```
.. function:: newwin(nlines, ncols)
              newwin(nlines, ncols, begin_y, begin_x)

   Return a new :ref:`window <curses-window-objects>`, whose left-upper corner
   is at  ``(begin_y, begin_x)``, and whose height/width is  *nlines*/*ncols*.

   By default, the window will extend from the  specified position to the lower
   right corner of the screen.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 416)

Unknown directive type "function".

```
.. function:: nl()

   Enter newline mode.  This mode translates the return key into newline on input,
   and translates newline into return and line-feed on output. Newline mode is
   initially on.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 423)

Unknown directive type "function".

```
.. function:: nocbreak()

   Leave cbreak mode.  Return to normal "cooked" mode with line buffering.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 428)

Unknown directive type "function".

```
.. function:: noecho()

   Leave echo mode.  Echoing of input characters is turned off.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 433)

Unknown directive type "function".

```
.. function:: nonl()

   Leave newline mode.  Disable translation of return into newline on input, and
   disable low-level translation of newline into newline/return on output (but this
   does not change the behavior of ``addch('\n')``, which always does the
   equivalent of return and line feed on the virtual screen).  With translation
   off, curses can sometimes speed up vertical motion a little; also, it will be
   able to detect the return key on input.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 443)

Unknown directive type "function".

```
.. function:: noqiflush()

   When the :func:`!noqiflush` routine is used, normal flush of input and output queues
   associated with the ``INTR``, ``QUIT`` and ``SUSP`` characters will not be done.  You may
   want to call :func:`!noqiflush` in a signal handler if you want output to
   continue as though the interrupt had not occurred, after the handler exits.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 451)

Unknown directive type "function".

```
.. function:: noraw()

   Leave raw mode. Return to normal "cooked" mode with line buffering.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 456)

Unknown directive type "function".

```
.. function:: pair_content(pair_number)

   Return a tuple ``(fg, bg)`` containing the colors for the requested color pair.
   The value of *pair_number* must be between ``0`` and ``COLOR_PAIRS - 1``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 462)

Unknown directive type "function".

```
.. function:: pair_number(attr)

   Return the number of the color-pair set by the attribute value *attr*.
   :func:`color_pair` is the counterpart to this function.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 468)

Unknown directive type "function".

```
.. function:: putp(str)

   Equivalent to ``tputs(str, 1, putchar)``; emit the value of a specified
   terminfo capability for the current terminal.  Note that the output of :func:`putp`
   always goes to standard output.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 475)

Unknown directive type "function".

```
.. function:: qiflush([flag])

   If *flag* is ``False``, the effect is the same as calling :func:`noqiflush`. If
   *flag* is ``True``, or no argument is provided, the queues will be flushed when
   these control characters are read.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 482)

Unknown directive type "function".

```
.. function:: raw()

   Enter raw mode.  In raw mode, normal line buffering and  processing of
   interrupt, quit, suspend, and flow control keys are turned off; characters are
   presented to curses input functions one by one.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 489)

Unknown directive type "function".

```
.. function:: reset_prog_mode()

   Restore the  terminal  to "program" mode, as previously saved  by
   :func:`def_prog_mode`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 495)

Unknown directive type "function".

```
.. function:: reset_shell_mode()

   Restore the  terminal  to "shell" mode, as previously saved  by
   :func:`def_shell_mode`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 501)

Unknown directive type "function".

```
.. function:: resetty()

   Restore the state of the terminal modes to what it was at the last call to
   :func:`savetty`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 507)

Unknown directive type "function".

```
.. function:: resize_term(nlines, ncols)

   Backend function used by :func:`resizeterm`, performing most of the work;
   when resizing the windows, :func:`resize_term` blank-fills the areas that are
   extended.  The calling application should fill in these areas with
   appropriate data.  The :func:`!resize_term` function attempts to resize all
   windows.  However, due to the calling convention of pads, it is not possible
   to resize these without additional interaction with the application.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 517)

Unknown directive type "function".

```
.. function:: resizeterm(nlines, ncols)

   Resize the standard and current windows to the specified dimensions, and
   adjusts other bookkeeping data used by the curses library that record the
   window dimensions (in particular the SIGWINCH handler).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 524)

Unknown directive type "function".

```
.. function:: savetty()

   Save the current state of the terminal modes in a buffer, usable by
   :func:`resetty`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 529)

Unknown directive type "function".

```
.. function:: get_escdelay()

   Retrieves the value set by :func:`set_escdelay`.
```

```
   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 535)

Unknown directive type "function".

```
.. function:: set_escdelay(ms)

   Sets the number of milliseconds to wait after reading an escape character,
   to distinguish between an individual escape character entered on the
   keyboard from escape sequences sent by cursor and function keys.

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 543)

Unknown directive type "function".

```
.. function:: get_tabsize()

   Retrieves the value set by :func:`set_tabsize`.

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 549)

Unknown directive type "function".

```
.. function:: set_tabsize(size)

   Sets the number of columns used by the curses library when converting a tab
   character to spaces as it adds the tab to a window.

   .. versionadded:: 3.9
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 556)

Unknown directive type "function".

```
.. function:: setsyx(y, x)

   Set the virtual screen cursor to *y*, *x*. If *y* and *x* are both ``-1``, then
   :meth:`leaveok <window.leaveok>` is set ``True``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 562)

Unknown directive type "function".

```
.. function:: setupterm(term=None, fd=-1)

   Initialize the terminal.  *term* is a string giving
   the terminal name, or ``None``; if omitted or ``None``, the value of the
   :envvar:`TERM` environment variable will be used.  *fd* is the
   file descriptor to which any initialization sequences will be sent; if not
   supplied or ``-1``, the file descriptor for ``sys.stdout`` will be used.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 571)

Unknown directive type "function".

```
.. function:: start_color()

   Must be called if the programmer wants to use colors, and before any other color
   manipulation routine is called.  It is good practice to call this routine right
   after :func:`initscr`.

   :func:`start_color` initializes eight basic colors (black, red,  green, yellow,
   blue, magenta, cyan, and white), and two global variables in the :mod:`curses`
   module, :const:`COLORS` and :const:`COLOR_PAIRS`, containing the maximum number
   of colors and color-pairs the terminal can support.  It also restores the colors
   on the terminal to the values they had when the terminal was just turned on.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 584)

Unknown directive type "function".

```
.. function:: termattrs()

   Return a logical OR of all video attributes supported by the terminal.  This
   information is useful when a curses program needs complete control over the
   appearance of the screen.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 591)

Unknown directive type "function".

```
.. function:: termname()

   Return the value of the environment variable :envvar:`TERM`, as a bytes object,
   truncated to 14 characters.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 597)

Unknown directive type "function".

```
.. function:: tigetflag(capname)

   Return the value of the Boolean capability corresponding to the terminfo
   capability name *capname* as an integer.  Return the value ``-1`` if *capname* is not a
   Boolean capability, or ``0`` if it is canceled or absent from the terminal
   description.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 605)

Unknown directive type "function".

```
.. function:: tigetnum(capname)

   Return the value of the numeric capability corresponding to the terminfo
   capability name *capname* as an integer.  Return the value ``-2`` if *capname* is not a
   numeric capability, or ``-1`` if it is canceled or absent from the terminal
   description.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 613)

Unknown directive type "function".

```
.. function:: tigetstr(capname)

   Return the value of the string capability corresponding to the terminfo
   capability name *capname* as a bytes object.  Return ``None`` if *capname*
   is not a terminfo "string capability", or is canceled or absent from the
   terminal description.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 621)

Unknown directive type "function".

```
.. function:: tparm(str[, ...])

   Instantiate the bytes object *str* with the supplied parameters, where *str* should
   be a parameterized string obtained from the terminfo database.  E.g.
   ``tparm(tigetstr("cup"), 5, 3)`` could result in ``b'\033[6;4H'``, the exact
   result depending on terminal type.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 629)

Unknown directive type "function".

```
.. function:: typeahead(fd)

   Specify that the file descriptor *fd* be used for typeahead checking.  If *fd*
   is ``-1``, then no typeahead checking is done.

   The curses library does "line-breakout optimization" by looking for typeahead
   periodically while updating the screen.  If input is found, and it is coming
   from a tty, the current update is postponed until refresh or doupdate is called
   again, allowing faster response to commands typed in advance. This function
   allows specifying a different file descriptor for typeahead checking.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 641)

Unknown directive type "function".

```
.. function:: unctrl(ch)

   Return a bytes object which is a printable representation of the character *ch*.
   Control characters are represented as a caret followed by the character, for
   example as ``b'^C'``. Printing characters are left as they are.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 648)

Unknown directive type "function".

```
.. function:: ungetch(ch)

   Push *ch* so the next :meth:`~window.getch` will return it.

   .. note::

      Only one *ch* can be pushed before :meth:`!getch` is called.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 657)

Unknown directive type "function".

```
.. function:: update_lines_cols()

   Update :envvar:`LINES` and :envvar:`COLS`. Useful for detecting manual screen resize.

   .. versionadded:: 3.5
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 664)

Unknown directive type "function".

```
.. function:: unget_wch(ch)

   Push *ch* so the next :meth:`~window.get_wch` will return it.

   .. note::

      Only one *ch* can be pushed before :meth:`!get_wch` is called.

   .. versionadded:: 3.3
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 675)

Unknown directive type "function".

```
.. function:: ungetmouse(id, x, y, z, bstate)

   Push a :const:`KEY_MOUSE` event onto the input queue, associating the given
   state data with it.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 681)

Unknown directive type "function".

```
.. function:: use_env(flag)

   If used, this function should be called before :func:`initscr` or newterm are
   called.  When *flag* is ``False``, the values of lines and columns specified in the
   terminfo database will be used, even if environment variables :envvar:`LINES`
   and :envvar:`COLUMNS` (used by default) are set, or if curses is running in a
   window (in which case default behavior would be to use the window size if
   :envvar:`LINES` and :envvar:`COLUMNS` are not set).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 691)

Unknown directive type "function".

```
.. function:: use_default_colors()

   Allow use of default values for colors on terminals supporting this feature. Use
   this to support transparency in your application.  The default color is assigned
   to the color number ``-1``. After calling this function,  ``init_pair(x,
   curses.COLOR_RED, -1)`` initializes, for instance, color pair *x* to a red
   foreground color on the default background.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 700)

Unknown directive type "function".

```
.. function:: wrapper(func, /, *args, **kwargs)

   Initialize curses and call another callable object, *func*, which should be the
   rest of your curses-using application.  If the application raises an exception,
   this function will restore the terminal to a sane state before re-raising the
   exception and generating a traceback.  The callable object *func* is then passed
   the main window 'stdscr' as its first argument, followed by any other arguments
   passed to :func:`!wrapper`.  Before calling *func*, :func:`!wrapper` turns on
   cbreak mode, turns off echo, enables the terminal keypad, and initializes colors
   if the terminal has color support.  On exit (whether normally or by exception)
   it restores cooked mode, turns on echo, and disables the terminal keypad.
```

Window Objects

Window objects, as returned by :func:`initscr` and :func:`newwin` above, have the following methods and attributes:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 718); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 718); *backlink*

Unknown interpreted text role "func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 722)

Unknown directive type "method".

```
.. method:: window.addch(ch[, attr])
            window.addch(y, x, ch[, attr])

   Paint character *ch* at ``(y, x)`` with attributes *attr*, overwriting any
   character previously painted at that location.  By default, the character
   position and attributes are the current settings for the window object.

   .. note::

      Writing outside the window, subwindow, or pad raises a :exc:`curses.error`.
      Attempting to write to the lower right corner of a window, subwindow,
      or pad will cause an exception to be raised after the character is printed.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-

main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 736)

Unknown directive type "method".

```
.. method:: window.addnstr(str, n[, attr])
            window.addnstr(y, x, str, n[, attr])

   Paint at most *n* characters of the character string *str* at
   ``(y, x)`` with attributes
   *attr*, overwriting anything previously on the display.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 744)

Unknown directive type "method".

```
.. method:: window.addstr(str[, attr])
            window.addstr(y, x, str[, attr])

   Paint the character string *str* at ``(y, x)`` with attributes
   *attr*, overwriting anything previously on the display.

   .. note::

      * Writing outside the window, subwindow, or pad raises :exc:`curses.error`.
        Attempting to write to the lower right corner of a window, subwindow,
        or pad will cause an exception to be raised after the string is printed.

      * A `bug in ncurses <https://bugs.python.org/issue35924>`_, the backend
        for this Python module, can cause SegFaults when resizing windows. This
        is fixed in ncurses-6.1-20190511.  If you are stuck with an earlier
        ncurses, you can avoid triggering this if you do not call :func:`addstr`
        with a *str* that has embedded newlines.  Instead, call :func:`addstr`
        separately for each line.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 764)

Unknown directive type "method".

```
.. method:: window.attroff(attr)

   Remove attribute *attr* from the "background" set applied to all writes to the
   current window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 770)

Unknown directive type "method".

```
.. method:: window.attron(attr)

   Add attribute *attr* from the "background" set applied to all writes to the
   current window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 776)

Unknown directive type "method".

```
.. method:: window.attrset(attr)

   Set the "background" set of attributes to *attr*.  This set is initially
   ``0`` (no attributes).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 782)

Unknown directive type "method".

```
.. method:: window.bkgd(ch[, attr])

   Set the background property of the window to the character *ch*, with
   attributes *attr*.  The change is then applied to every character position in
   that window:

   * The attribute of every character in the window  is changed to the new
     background attribute.
```

```
   * Wherever  the  former background character appears, it is changed to the new
     background character.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 795)

Unknown directive type "method".

```
.. method:: window.bkgdset(ch[, attr])

   Set the window's background.  A window's background consists of a character and
   any combination of attributes.  The attribute part of the background is combined
   (OR'ed) with all non-blank characters that are written into the window.  Both
   the character and attribute parts of the background are combined with the blank
   characters.  The background becomes a property of the character and moves with
   the character through any scrolling and insert/delete line/character operations.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 805)

Unknown directive type "method".

```
.. method:: window.border([ls[, rs[, ts[, bs[, tl[, tr[, bl[, br]]]]]]]])

   Draw a border around the edges of the window. Each parameter specifies  the
   character to use for a specific part of the border; see the table below for more
   details.

   .. note::

      A ``0`` value for any parameter will cause the default character to be used for
      that parameter.  Keyword parameters can *not* be used.  The defaults are listed
      in this table:

   +-----------+---------------------+-----------------------+
   | Parameter | Description         | Default value         |
   +===========+=====================+=======================+
   | *ls*      | Left side           | :const:`ACS_VLINE`    |
   +-----------+---------------------+-----------------------+
   | *rs*      | Right side          | :const:`ACS_VLINE`    |
   +-----------+---------------------+-----------------------+
   | *ts*      | Top                 | :const:`ACS_HLINE`    |
   +-----------+---------------------+-----------------------+
   | *bs*      | Bottom              | :const:`ACS_HLINE`    |
   +-----------+---------------------+-----------------------+
   | *tl*      | Upper-left corner   | :const:`ACS_ULCORNER` |
   +-----------+---------------------+-----------------------+
   | *tr*      | Upper-right corner  | :const:`ACS_URCORNER` |
   +-----------+---------------------+-----------------------+
   | *bl*      | Bottom-left corner  | :const:`ACS_LLCORNER` |
   +-----------+---------------------+-----------------------+
   | *br*      | Bottom-right corner | :const:`ACS_LRCORNER` |
   +-----------+---------------------+-----------------------+
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 838)

Unknown directive type "method".

```
.. method:: window.box([vertch, horch])

   Similar to :meth:`border`, but both *ls* and *rs* are *vertch* and both *ts* and
   *bs* are *horch*.  The default corner characters are always used by this function.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 844)

Unknown directive type "method".

```
.. method:: window.chgat(attr)
            window.chgat(num, attr)
            window.chgat(y, x, attr)
            window.chgat(y, x, num, attr)

   Set the attributes of *num* characters at the current cursor position, or at
   position ``(y, x)`` if supplied. If *num* is not given or is ``-1``,
   the attribute will be set on all the characters to the end of the line.  This
   function moves cursor to position ``(y, x)`` if supplied. The changed line
   will be touched using the :meth:`touchline` method so that the contents will
```

```
   be redisplayed by the next window refresh.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 857)

Unknown directive type "method".

```
.. method:: window.clear()

   Like :meth:`erase`, but also cause the whole window to be repainted upon next
   call to :meth:`refresh`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 863)

Unknown directive type "method".

```
.. method:: window.clearok(flag)

   If *flag* is ``True``, the next call to :meth:`refresh` will clear the window
   completely.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 869)

Unknown directive type "method".

```
.. method:: window.clrtobot()

   Erase from cursor to the end of the window: all lines below the cursor are
   deleted, and then the equivalent of :meth:`clrtoeol` is performed.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 875)

Unknown directive type "method".

```
.. method:: window.clrtoeol()

   Erase from cursor to the end of the line.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 880)

Unknown directive type "method".

```
.. method:: window.cursyncup()

   Update the current cursor position of all the ancestors of the window to
   reflect the current cursor position of the window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 886)

Unknown directive type "method".

```
.. method:: window.delch([y, x])

   Delete any character at ``(y, x)``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 891)

Unknown directive type "method".

```
.. method:: window.deleteln()

   Delete the line under the cursor. All following lines are moved up by one line.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 896)

Unknown directive type "method".

```
.. method:: window.derwin(begin_y, begin_x)
            window.derwin(nlines, ncols, begin_y, begin_x)

   An abbreviation for "derive window", :meth:`derwin` is the same as calling
   :meth:`subwin`, except that *begin_y* and *begin_x* are relative to the origin
   of the window, rather than relative to the entire screen.  Return a window
   object for the derived window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 905)

Unknown directive type "method".

```
.. method:: window.echochar(ch[, attr])

   Add character *ch* with attribute *attr*, and immediately  call :meth:`refresh`
   on the window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 911)

Unknown directive type "method".

```
.. method:: window.enclose(y, x)

   Test whether the given pair of screen-relative character-cell coordinates are
   enclosed by the given window, returning ``True`` or ``False``.  It is useful for
   determining what subset of the screen windows enclose the location of a mouse
   event.

   .. versionchanged:: 3.10
      Previously it returned ``1`` or ``0`` instead of ``True`` or ``False``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 922)

Unknown directive type "attribute".

```
.. attribute:: window.encoding

   Encoding used to encode method arguments (Unicode strings and characters).
   The encoding attribute is inherited from the parent window when a subwindow
   is created, for example with :meth:`window.subwin`. By default, the locale
   encoding is used (see :func:`locale.getpreferredencoding`).

   .. versionadded:: 3.3
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 932)

Unknown directive type "method".

```
.. method:: window.erase()

   Clear the window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 937)

Unknown directive type "method".

```
.. method:: window.getbegyx()

   Return a tuple ``(y, x)`` of co-ordinates of upper-left corner.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 942)

Unknown directive type "method".

```
.. method:: window.getbkgd()

   Return the given window's current background character/attribute pair.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 947)

Unknown directive type "method".

```
.. method:: window.getch([y, x])

   Get a character. Note that the integer returned does *not* have to be in ASCII
   range: function keys, keypad keys and so on are represented by numbers higher
   than 255.  In no-delay mode, return ``-1`` if there is no input, otherwise
   wait until a key is pressed.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 955)

Unknown directive type "method".

```
.. method:: window.get_wch([y, x])

   Get a wide character. Return a character for most keys, or an integer for
   function keys, keypad keys, and other special keys.
   In no-delay mode, raise an exception if there is no input.

   .. versionadded:: 3.3
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 964)

Unknown directive type "method".

```
.. method:: window.getkey([y, x])

   Get a character, returning a string instead of an integer, as :meth:`getch`
   does. Function keys, keypad keys and other special keys return a multibyte
   string containing the key name.  In no-delay mode, raise an exception if
   there is no input.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 972)

Unknown directive type "method".

```
.. method:: window.getmaxyx()

   Return a tuple ``(y, x)`` of the height and width of the window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 977)

Unknown directive type "method".

```
.. method:: window.getparyx()

   Return the beginning coordinates of this window relative to its parent window
   as a tuple ``(y, x)``.  Return ``(-1, -1)`` if this window has no
   parent.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 984)

Unknown directive type "method".

```
.. method:: window.getstr()
            window.getstr(n)
            window.getstr(y, x)
            window.getstr(y, x, n)

   Read a bytes object from the user, with primitive line editing capacity.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 992)

Unknown directive type "method".

```
.. method:: window.getyx()

   Return a tuple ``(y, x)`` of current cursor position  relative to the window's
   upper-left corner.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 998)

Unknown directive type "method".

```
.. method:: window.hline(ch, n)
            window.hline(y, x, ch, n)

   Display a horizontal line starting at ``(y, x)`` with length *n* consisting of
   the character *ch*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1005)

Unknown directive type "method".

```
.. method:: window.idcok(flag)

   If *flag* is ``False``, curses no longer considers using the hardware insert/delete
   character feature of the terminal; if *flag* is ``True``, use of character insertion
   and deletion is enabled.  When curses is first initialized, use of character
   insert/delete is enabled by default.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1013)

Unknown directive type "method".

```
.. method:: window.idlok(flag)

   If *flag* is ``True``, :mod:`curses` will try and use hardware line
   editing facilities. Otherwise, line insertion/deletion are disabled.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1019)

Unknown directive type "method".

```
.. method:: window.immedok(flag)

   If *flag* is ``True``, any change in the window image automatically causes the
   window to be refreshed; you no longer have to call :meth:`refresh` yourself.
   However, it may degrade performance considerably, due to repeated calls to
   wrefresh.  This option is disabled by default.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1027)

Unknown directive type "method".

```
.. method:: window.inch([y, x])

   Return the character at the given position in the window. The bottom 8 bits are
   the character proper, and upper bits are the attributes.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1033)

Unknown directive type "method".

```
.. method:: window.insch(ch[, attr])
            window.insch(y, x, ch[, attr])

   Paint character *ch* at ``(y, x)`` with attributes *attr*, moving the line from
```

```
   position *x* right by one character.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1040)

Unknown directive type "method".

```
.. method:: window.insdelln(nlines)

   Insert *nlines* lines into the specified window above the current line.  The
   *nlines* bottom lines are lost.  For negative *nlines*, delete *nlines* lines
   starting with the one under the cursor, and move the remaining lines up.  The
   bottom *nlines* lines are cleared.  The current cursor position remains the
   same.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1049)

Unknown directive type "method".

```
.. method:: window.insertln()

   Insert a blank line under the cursor. All following lines are moved down by one
   line.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1055)

Unknown directive type "method".

```
.. method:: window.insnstr(str, n[, attr])
            window.insnstr(y, x, str, n[, attr])

   Insert a character string (as many characters as will fit on the line) before
   the character under the cursor, up to *n* characters.   If *n* is zero or
   negative, the entire string is inserted. All characters to the right of the
   cursor are shifted right, with the rightmost characters on the line being lost.
   The cursor position does not change (after moving to *y*, *x*, if specified).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1065)

Unknown directive type "method".

```
.. method:: window.insstr(str[, attr])
            window.insstr(y, x, str[, attr])

   Insert a character string (as many characters as will fit on the line) before
   the character under the cursor.  All characters to the right of the cursor are
   shifted right, with the rightmost characters on the line being lost.  The cursor
   position does not change (after moving to *y*, *x*, if specified).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1074)

Unknown directive type "method".

```
.. method:: window.instr([n])
            window.instr(y, x[, n])

   Return a bytes object of characters, extracted from the window starting at the
   current cursor position, or at *y*, *x* if specified. Attributes are stripped
   from the characters.  If *n* is specified, :meth:`instr` returns a string
   at most *n* characters long (exclusive of the trailing NUL).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1083)

Unknown directive type "method".

```
.. method:: window.is_linetouched(line)

   Return ``True`` if the specified line was modified since the last call to
   :meth:`refresh`; otherwise return ``False``.  Raise a :exc:`curses.error`
```

```
   exception if *line* is not valid for the given window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1090)

Unknown directive type "method".

```
.. method:: window.is_wintouched()

   Return ``True`` if the specified window was modified since the last call to
   :meth:`refresh`; otherwise return ``False``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1096)

Unknown directive type "method".

```
.. method:: window.keypad(flag)

   If *flag* is ``True``, escape sequences generated by some keys (keypad,  function keys)
   will be interpreted by :mod:`curses`. If *flag* is ``False``, escape sequences will be
   left as is in the input stream.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1103)

Unknown directive type "method".

```
.. method:: window.leaveok(flag)

   If *flag* is ``True``, cursor is left where it is on update, instead of being at "cursor
   position."  This reduces cursor movement where possible. If possible the cursor
   will be made invisible.

   If *flag* is ``False``, cursor will always be at "cursor position" after an update.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1112)

Unknown directive type "method".

```
.. method:: window.move(new_y, new_x)

   Move cursor to ``(new_y, new_x)``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1117)

Unknown directive type "method".

```
.. method:: window.mvderwin(y, x)

   Move the window inside its parent window.  The screen-relative parameters of
   the window are not changed.  This routine is used to display different parts of
   the parent window at the same physical position on the screen.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1124)

Unknown directive type "method".

```
.. method:: window.mvwin(new_y, new_x)

   Move the window so its upper-left corner is at ``(new_y, new_x)``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1129)

Unknown directive type "method".

```
.. method:: window.nodelay(flag)
```

```
   If *flag* is ``True``, :meth:`getch` will be non-blocking.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1134)

Unknown directive type "method".

```
.. method:: window.notimeout(flag)

   If *flag* is ``True``, escape sequences will not be timed out.

   If *flag* is ``False``, after a few milliseconds, an escape sequence will not be
   interpreted, and will be left in the input stream as is.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1142)

Unknown directive type "method".

```
.. method:: window.noutrefresh()

   Mark for refresh but wait.  This function updates the data structure
   representing the desired state of the window, but does not force an update of
   the physical screen.  To accomplish that, call  :func:`doupdate`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1149)

Unknown directive type "method".

```
.. method:: window.overlay(destwin[, sminrow, smincol, dminrow, dmincol, dmaxrow, dmaxcol])

   Overlay the window on top of *destwin*. The windows need not be the same size,
   only the overlapping region is copied. This copy is non-destructive, which means
   that the current background character does not overwrite the old contents of
   *destwin*.

   To get fine-grained control over the copied region, the second form of
   :meth:`overlay` can be used. *sminrow* and *smincol* are the upper-left
   coordinates of the source window, and the other variables mark a rectangle in
   the destination window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1162)

Unknown directive type "method".

```
.. method:: window.overwrite(destwin[, sminrow, smincol, dminrow, dmincol, dmaxrow, dmaxcol])

   Overwrite the window on top of *destwin*. The windows need not be the same size,
   in which case only the overlapping region is copied. This copy is destructive,
   which means that the current background character overwrites the old contents of
   *destwin*.

   To get fine-grained control over the copied region, the second form of
   :meth:`overwrite` can be used. *sminrow* and *smincol* are the upper-left
   coordinates of the source window, the other variables mark a rectangle in the
   destination window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1175)

Unknown directive type "method".

```
.. method:: window.putwin(file)

   Write all data associated with the window into the provided file object.  This
   information can be later retrieved using the :func:`getwin` function.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1181)

Unknown directive type "method".

```
.. method:: window.redrawln(beg, num)

   Indicate that the *num* screen lines, starting at line *beg*, are corrupted and
   should be completely redrawn on the next :meth:`refresh` call.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1187)

Unknown directive type "method".

```
.. method:: window.redrawwin()

   Touch the entire window, causing it to be completely redrawn on the next
   :meth:`refresh` call.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1193)

Unknown directive type "method".

```
.. method:: window.refresh([pminrow, pmincol, sminrow, smincol, smaxrow, smaxcol])

   Update the display immediately (sync actual screen with previous
   drawing/deleting methods).

   The 6 optional arguments can only be specified when the window is a pad created
   with :func:`newpad`.  The additional parameters are needed to indicate what part
   of the pad and screen are involved. *pminrow* and *pmincol* specify the upper
   left-hand corner of the rectangle to be displayed in the pad.  *sminrow*,
   *smincol*, *smaxrow*, and *smaxcol* specify the edges of the rectangle to be
   displayed on the screen.  The lower right-hand corner of the rectangle to be
   displayed in the pad is calculated from the screen coordinates, since the
   rectangles must be the same size.  Both rectangles must be entirely contained
   within their respective structures.  Negative values of *pminrow*, *pmincol*,
   *sminrow*, or *smincol* are treated as if they were zero.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1210)

Unknown directive type "method".

```
.. method:: window.resize(nlines, ncols)

   Reallocate storage for a curses window to adjust its dimensions to the
   specified values.  If either dimension is larger than the current values, the
   window's data is filled with blanks that have the current background
   rendition (as set by :meth:`bkgdset`) merged into them.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1218)

Unknown directive type "method".

```
.. method:: window.scroll([lines=1])

   Scroll the screen or scrolling region upward by *lines* lines.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1223)

Unknown directive type "method".

```
.. method:: window.scrollok(flag)

   Control what happens when the cursor of a window is moved off the edge of the
   window or scrolling region, either as a result of a newline action on the bottom
   line, or typing the last character of the last line.  If *flag* is ``False``, the
   cursor is left on the bottom line.  If *flag* is ``True``, the window is scrolled up
   one line.  Note that in order to get the physical scrolling effect on the
   terminal, it is also necessary to call :meth:`idlok`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1233)

Unknown directive type "method".

```
.. method:: window.setscrreg(top, bottom)

   Set the scrolling region from line *top* to line *bottom*. All scrolling actions
   will take place in this region.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1239)

Unknown directive type "method".

```
.. method:: window.standend()

   Turn off the standout attribute.  On some terminals this has the side effect of
   turning off all attributes.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1245)

Unknown directive type "method".

```
.. method:: window.standout()

   Turn on attribute *A_STANDOUT*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1250)

Unknown directive type "method".

```
.. method:: window.subpad(begin_y, begin_x)
            window.subpad(nlines, ncols, begin_y, begin_x)

   Return a sub-window, whose upper-left corner is at ``(begin_y, begin_x)``, and
   whose width/height is *ncols*/*nlines*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1257)

Unknown directive type "method".

```
.. method:: window.subwin(begin_y, begin_x)
            window.subwin(nlines, ncols, begin_y, begin_x)

   Return a sub-window, whose upper-left corner is at ``(begin_y, begin_x)``, and
   whose width/height is *ncols*/*nlines*.

   By default, the sub-window will extend from the specified position to the lower
   right corner of the window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1267)

Unknown directive type "method".

```
.. method:: window.syncdown()

   Touch each location in the window that has been touched in any of its ancestor
   windows.  This routine is called by :meth:`refresh`, so it should almost never
   be necessary to call it manually.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1274)

Unknown directive type "method".

```
.. method:: window.syncok(flag)

   If *flag* is ``True``, then :meth:`syncup` is called automatically
   whenever there is a change in the window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1280)

Unknown directive type "method".

```
.. method:: window.syncup()

   Touch all locations in ancestors of the window that have been changed in  the
   window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1286)

Unknown directive type "method".

```
.. method:: window.timeout(delay)

   Set blocking or non-blocking read behavior for the window.  If *delay* is
   negative, blocking read is used (which will wait indefinitely for input).  If
   *delay* is zero, then non-blocking read is used, and :meth:`getch` will
   return ``-1`` if no input is waiting.  If *delay* is positive, then
   :meth:`getch` will block for *delay* milliseconds, and return ``-1`` if there is
   still no input at the end of that time.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1296)

Unknown directive type "method".

```
.. method:: window.touchline(start, count[, changed])

   Pretend *count* lines have been changed, starting with line *start*.  If
   *changed* is supplied, it specifies whether the affected lines are marked as
   having been changed (*changed*\ ``=True``) or unchanged (*changed*\ ``=False``).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1303)

Unknown directive type "method".

```
.. method:: window.touchwin()

   Pretend the whole window has been changed, for purposes of drawing
   optimizations.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1309)

Unknown directive type "method".

```
.. method:: window.untouchwin()

   Mark all lines in  the  window  as unchanged since the last call to
   :meth:`refresh`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1315)

Unknown directive type "method".

```
.. method:: window.vline(ch, n)
            window.vline(y, x, ch, n)

   Display a vertical line starting at ``(y, x)`` with length *n* consisting of the
   character *ch*.
```

Constants

The :mod:`curses` module defines the following data members:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1325); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1328)

Unknown directive type "data".

```
.. data:: ERR

   Some curses routines  that  return  an integer, such as :meth:`~window.getch`, return
   :const:`ERR` upon failure.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1334)

Unknown directive type "data".

```
.. data:: OK

   Some curses routines  that  return  an integer, such as  :func:`napms`, return
   :const:`OK` upon success.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1340)

Unknown directive type "data".

```
.. data:: version

   A bytes object representing the current version of the module.  Also available as
   :const:`__version__`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1346)

Unknown directive type "data".

```
.. data:: ncurses_version

   A named tuple containing the three components of the ncurses library
   version: *major*, *minor*, and *patch*.  All values are integers.  The
   components can also be accessed by name,  so ``curses.ncurses_version[0]``
   is equivalent to ``curses.ncurses_version.major`` and so on.

   Availability: if the ncurses library is used.

   .. versionadded:: 3.8
```

Some constants are available to specify character cell attributes. The exact constants available are system dependent.

Attribute	Meaning
A_ALTCHARSET	Alternate character set mode
A_BLINK	Blink mode
A_BOLD	Bold mode
A_DIM	Dim mode
A_INVIS	Invisible or blank mode
A_ITALIC	Italic mode
A_NORMAL	Normal attribute
A_PROTECT	Protected mode
A_REVERSE	Reverse background and foreground colors
A_STANDOUT	Standout mode
A_UNDERLINE	Underline mode
A_HORIZONTAL	Horizontal highlight
A_LEFT	Left highlight
A_LOW	Low highlight
A_RIGHT	Right highlight
A_TOP	Top highlight
A_VERTICAL	Vertical highlight
A_CHARTEXT	Bit-mask to extract a character

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1403)

Unknown directive type "versionadded".

```
.. versionadded:: 3.7
   ``A_ITALIC`` was added.
```

Several constants are available to extract corresponding attributes returned by some methods.

Bit-mask	Meaning
A_ATTRIBUTES	Bit-mask to extract attributes
A_CHARTEXT	Bit-mask to extract a character
A_COLOR	Bit-mask to extract color-pair field information

Keys are referred to by integer constants with names starting with KEY_. The exact keycaps available are system dependent.

Key constant	Key
KEY_MIN	Minimum key value
KEY_BREAK	Break key (unreliable)
KEY_DOWN	Down-arrow
KEY_UP	Up-arrow
KEY_LEFT	Left-arrow
KEY_RIGHT	Right-arrow
KEY_HOME	Home key (upward+left arrow)
KEY_BACKSPACE	Backspace (unreliable)
KEY_F0	Function keys. Up to 64 function keys are supported.
KEY_Fn	Value of function key *n*
KEY_DL	Delete line
KEY_IL	Insert line
KEY_DC	Delete character
KEY_IC	Insert char or enter insert mode
KEY_EIC	Exit insert char mode
KEY_CLEAR	Clear screen
KEY_EOS	Clear to end of screen
KEY_EOL	Clear to end of line
KEY_SF	Scroll 1 line forward
KEY_SR	Scroll 1 line backward (reverse)
KEY_NPAGE	Next page
KEY_PPAGE	Previous page
KEY_STAB	Set tab
KEY_CTAB	Clear tab
KEY_CATAB	Clear all tabs
KEY_ENTER	Enter or send (unreliable)
KEY_SRESET	Soft (partial) reset (unreliable)
KEY_RESET	Reset or hard reset (unreliable)
KEY_PRINT	Print
KEY_LL	Home down or bottom (lower left)
KEY_A1	Upper left of keypad
KEY_A3	Upper right of keypad
KEY_B2	Center of keypad
KEY_C1	Lower left of keypad
KEY_C3	Lower right of keypad
KEY_BTAB	Back tab
KEY_BEG	Beg (beginning)
KEY_CANCEL	Cancel
KEY_CLOSE	Close
KEY_COMMAND	Cmd (command)
KEY_COPY	Copy
KEY_CREATE	Create
KEY_END	End
KEY_EXIT	Exit
KEY_FIND	Find
KEY_HELP	Help
KEY_MARK	Mark
KEY_MESSAGE	Message
KEY_MOVE	Move
KEY_NEXT	Next
KEY_OPEN	Open
KEY_OPTIONS	Options
KEY_PREVIOUS	Prev (previous)
KEY_REDO	Redo
KEY_REFERENCE	Ref (reference)

Key constant	Key
KEY_REFRESH	Refresh
KEY_REPLACE	Replace
KEY_RESTART	Restart
KEY_RESUME	Resume
KEY_SAVE	Save
KEY_SBEG	Shifted Beg (beginning)
KEY_SCANCEL	Shifted Cancel
KEY_SCOMMAND	Shifted Command
KEY_SCOPY	Shifted Copy
KEY_SCREATE	Shifted Create
KEY_SDC	Shifted Delete char
KEY_SDL	Shifted Delete line
KEY_SELECT	Select
KEY_SEND	Shifted End
KEY_SEOL	Shifted Clear line
KEY_SEXIT	Shifted Exit
KEY_SFIND	Shifted Find
KEY_SHELP	Shifted Help
KEY_SHOME	Shifted Home
KEY_SIC	Shifted Input
KEY_SLEFT	Shifted Left arrow
KEY_SMESSAGE	Shifted Message
KEY_SMOVE	Shifted Move
KEY_SNEXT	Shifted Next
KEY_SOPTIONS	Shifted Options
KEY_SPREVIOUS	Shifted Prev
KEY_SPRINT	Shifted Print
KEY_SREDO	Shifted Redo
KEY_SREPLACE	Shifted Replace
KEY_SRIGHT	Shifted Right arrow
KEY_SRSUME	Shifted Resume
KEY_SSAVE	Shifted Save
KEY_SSUSPEND	Shifted Suspend
KEY_SUNDO	Shifted Undo
KEY_SUSPEND	Suspend
KEY_UNDO	Undo
KEY_MOUSE	Mouse event has occurred
KEY_RESIZE	Terminal resize event
KEY_MAX	Maximum key value

On VT100s and their software emulations, such as X terminal emulators, there are normally at least four function keys (:const:`KEY_F1`, :const:`KEY_F2`, :const:`KEY_F3`, :const:`KEY_F4`) available, and the arrow keys mapped to :const:`KEY_UP`, :const:`KEY_DOWN`, :const:`KEY_LEFT` and :const:`KEY_RIGHT` in the obvious way. If your machine has a PC keyboard, it is safe to expect arrow keys and twelve function keys (older PC keyboards may have only ten function keys); also, the following keypad mappings are standard:

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)curses.rst`, **line 1620**); ***backlink***

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)curses.rst`, **line 1620**); ***backlink***

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)curses.rst`, **line 1620**); ***backlink***

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)curses.rst`, **line 1620**); ***backlink***

Unknown interpreted text role "const".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main) (Doc) (library)curses.rst`, **line 1620**); ***backlink***

Unknown interpreted text role "const".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1620); *backlink*

Unknown interpreted text role "const".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1620); *backlink*

Unknown interpreted text role "const".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1620); *backlink*

Unknown interpreted text role "const".

Keycap	Constant
:kbd:`Insert` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1632); *backlink* Unknown interpreted text role "kbd".	KEY_IC
:kbd:`Delete` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1634); *backlink* Unknown interpreted text role "kbd".	KEY_DC
:kbd:`Home` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1636); *backlink* Unknown interpreted text role "kbd".	KEY_HOME
:kbd:`End` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1638); *backlink* Unknown interpreted text role "kbd".	KEY_END
:kbd:`Page Up` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1640); *backlink* Unknown interpreted text role "kbd".	KEY_PPAGE
:kbd:`Page Down` System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1642); *backlink* Unknown interpreted text role "kbd".	KEY_NPAGE

The following table lists characters from the alternate character set. These are inherited from the VT100 terminal, and will generally be available on software emulations such as X terminals. When there is no graphic available, curses falls back on a crude printable ASCII approximation.

Note

These are available only after :func:`initscr` has been called.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst`, line 1651); *backlink*

Unknown interpreted text role "func".

ACS code	Meaning
ACS_BBSS	alternate name for upper right corner
ACS_BLOCK	solid square block
ACS_BOARD	board of squares
ACS_BSBS	alternate name for horizontal line
ACS_BSSB	alternate name for upper left corner
ACS_BSSS	alternate name for top tee
ACS_BTEE	bottom tee
ACS_BULLET	bullet
ACS_CKBOARD	checker board (stipple)
ACS_DARROW	arrow pointing down
ACS_DEGREE	degree symbol
ACS_DIAMOND	diamond
ACS_GEQUAL	greater-than-or-equal-to
ACS_HLINE	horizontal line
ACS_LANTERN	lantern symbol
ACS_LARROW	left arrow
ACS_LEQUAL	less-than-or-equal-to
ACS_LLCORNER	lower left-hand corner
ACS_LRCORNER	lower right-hand corner
ACS_LTEE	left tee
ACS_NEQUAL	not-equal sign
ACS_PI	letter pi
ACS_PLMINUS	plus-or-minus sign
ACS_PLUS	big plus sign
ACS_RARROW	right arrow
ACS_RTEE	right tee
ACS_S1	scan line 1
ACS_S3	scan line 3
ACS_S7	scan line 7
ACS_S9	scan line 9
ACS_SBBS	alternate name for lower right corner
ACS_SBSB	alternate name for vertical line
ACS_SBSS	alternate name for right tee
ACS_SSBB	alternate name for lower left corner
ACS_SSBS	alternate name for bottom tee
ACS_SSSB	alternate name for left tee
ACS_SSSS	alternate name for crossover or big plus
ACS_STERLING	pound sterling
ACS_TTEE	top tee
ACS_UARROW	up arrow
ACS_ULCORNER	upper left corner
ACS_URCORNER	upper right corner
ACS_VLINE	vertical line

The following table lists the predefined colors:

Constant	Color
COLOR_BLACK	Black
COLOR_BLUE	Blue
COLOR_CYAN	Cyan (light greenish blue)
COLOR_GREEN	Green
COLOR_MAGENTA	Magenta (purplish red)
COLOR_RED	Red
COLOR_WHITE	White
COLOR_YELLOW	Yellow

:mod:`curses.textpad` --- Text input widget for curses programs

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1766); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1769)

Unknown directive type "module".

```
.. module:: curses.textpad
   :synopsis: Emacs-like input editing in a curses window.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1771)

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Eric Raymond <esr@thyrsus.com>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1772)

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Eric Raymond <esr@thyrsus.com>
```

The :mod:`curses.textpad` module provides a :class:`Textbox` class that handles elementary text editing in a curses window, supporting a set of keybindings resembling those of Emacs (thus, also of Netscape Navigator, BBedit 6.x, FrameMaker, and many other programs). The module also provides a rectangle-drawing function useful for framing text boxes or for other purposes.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1775); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1775); *backlink*

Unknown interpreted text role "class".

The module :mod:`curses.textpad` defines the following function:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1781); *backlink*

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1784)

Unknown directive type "function".

```
.. function:: rectangle(win, uly, ulx, lry, lrx)

   Draw a rectangle.  The first argument must be a window object; the remaining
   arguments are coordinates relative to that window.  The second and third
   arguments are the y and x coordinates of the upper left hand corner of the
   rectangle to be drawn; the fourth and fifth arguments are the y and x
   coordinates of the lower right hand corner. The rectangle will be drawn using
   VT100/IBM PC forms characters on terminals that make this possible (including
   xterm and most other software terminal emulators).  Otherwise it will be drawn
   with ASCII  dashes, vertical bars, and plus signs.
```

Textbox objects

You can instantiate a :class:`Textbox` object as follows:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1801); *backlink*

Unknown interpreted text role "class".

Return a textbox widget object. The *win* argument should be a curses :ref:`window <curses-window-objects>` object in which the textbox is to be contained. The edit cursor of the textbox is initially located at the upper left hand corner of the containing window,

with coordinates `(0, 0)`. The instance's :attr:`stripspaces` flag is initially on.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1806); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1806); *backlink*

Unknown interpreted text role "attr".

:class:`Textbox` objects have the following methods:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1812); *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1815)

Unknown directive type "method".

```
.. method:: edit([validator])

   This is the entry point you will normally use.  It accepts editing
   keystrokes until one of the termination keystrokes is entered.  If
   *validator* is supplied, it must be a function.  It will be called for
   each keystroke entered with the keystroke as a parameter; command dispatch
   is done on the result. This method returns the window contents as a
   string; whether blanks in the window are included is affected by the
   :attr:`stripspaces` attribute.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1826)

Unknown directive type "method".

```
.. method:: do_command(ch)

   Process a single command keystroke.  Here are the supported special
   keystrokes:

   +------------------+-----------------------------------------+
   | Keystroke        | Action                                  |
   +==================+=========================================+
   | :kbd:`Control-A` | Go to left edge of window.              |
   +------------------+-----------------------------------------+
   | :kbd:`Control-B` | Cursor left, wrapping to previous line if |
   |                  | appropriate.                            |
   +------------------+-----------------------------------------+
   | :kbd:`Control-D` | Delete character under cursor.          |
   +------------------+-----------------------------------------+
   | :kbd:`Control-E` | Go to right edge (stripspaces off) or end |
   |                  | of line (stripspaces on).               |
   +------------------+-----------------------------------------+
   | :kbd:`Control-F` | Cursor right, wrapping to next line when  |
   |                  | appropriate.                            |
   +------------------+-----------------------------------------+
   | :kbd:`Control-G` | Terminate, returning the window contents. |
   +------------------+-----------------------------------------+
   | :kbd:`Control-H` | Delete character backward.              |
   +------------------+-----------------------------------------+
   | :kbd:`Control-J` | Terminate if the window is 1 line,      |
   |                  | otherwise insert newline.               |
   +------------------+-----------------------------------------+
   | :kbd:`Control-K` | If line is blank, delete it, otherwise    |
   |                  | clear to end of line.                   |
   +------------------+-----------------------------------------+
   | :kbd:`Control-L` | Refresh screen.                         |
   +------------------+-----------------------------------------+
   | :kbd:`Control-N` | Cursor down; move down one line.        |
   +------------------+-----------------------------------------+
   | :kbd:`Control-O` | Insert a blank line at cursor location.   |
   +------------------+-----------------------------------------+
   | :kbd:`Control-P` | Cursor up; move up one line.            |
   +------------------+-----------------------------------------+

   Move operations do nothing if the cursor is at an edge where the movement
   is not possible.  The following synonyms are supported where possible:

   +------------------------+------------------+
   | Constant               | Keystroke        |
```

```
+========================+==================+
| :const:`KEY_LEFT`      | :kbd:`Control-B` |
+------------------------+------------------+
| :const:`KEY_RIGHT`     | :kbd:`Control-F` |
+------------------------+------------------+
| :const:`KEY_UP`        | :kbd:`Control-P` |
+------------------------+------------------+
| :const:`KEY_DOWN`      | :kbd:`Control-N` |
+------------------------+------------------+
| :const:`KEY_BACKSPACE` | :kbd:`Control-h` |
+------------------------+------------------+

All other keystrokes are treated as a command to insert the given
character and move right (with line wrapping).
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1887)

Unknown directive type "method".

```
.. method:: gather()

   Return the window contents as a string; whether blanks in the
   window are included is affected by the :attr:`stripspaces` member.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\(cpython-main)(Doc)(library)curses.rst, line 1893)

Unknown directive type "attribute".

```
.. attribute:: stripspaces

   This attribute is a flag which controls the interpretation of blanks in
   the window.  When it is on, trailing blanks on each line are ignored; any
   cursor motion that would land the cursor on a trailing blank goes to the
   end of that line instead, and trailing blanks are stripped when the window
   contents are gathered.
```