

getpass — Portable password input

Source code: [Lib/getpass.py](#)

Availability: not WASI.

This module does not work or is not available on WebAssembly. See [WebAssembly platforms](#) for more information.

The `getpass` module provides two functions:

`getpass.getpass(prompt='Password: ', stream=None, *, echo_char=None)`

Prompt the user for a password without echoing. The user is prompted using the string *prompt*, which defaults to 'Password: '. On Unix, the prompt is written to the file-like object *stream* using the replace error handler if needed. *stream* defaults to the controlling terminal (`/dev/tty`) or if that is unavailable to `sys.stderr` (this argument is ignored on Windows).

The *echo_char* argument controls how user input is displayed while typing. If *echo_char* is `None` (default), input remains hidden. Otherwise, *echo_char* must be a single printable ASCII character and each typed character is replaced by it. For example, *echo_char*='*' will display asterisks instead of the actual input.

If echo free input is unavailable `getpass()` falls back to printing a warning message to *stream* and reading from `sys.stdin` and issuing a [GetPassWarning](#).

Note: If you call `getpass` from within IDLE, the input may be done in the terminal you launched IDLE from rather than the idle window itself.

Note: On Unix systems, when *echo_char* is set, the terminal will be configured to operate in [noncanonical mode](#). In particular, this means that line editing shortcuts such as `Ctrl+U` will not work and may insert unexpected characters into the input.

Changed in version 3.14: Added the *echo_char* parameter for keyboard feedback.

exception `getpass.GetPassWarning`

A [UserWarning](#) subclass issued when password input may be echoed.

`getpass.getuser()`

Return the “login name” of the user.

This function checks the environment variables `LOGNAME`, `USER`, `LNAME` and `USERNAME`, in order, and returns the value of the first one which is set to a non-empty string. If none are set, the login name from the password database is returned on systems which support the [pwd](#) module, otherwise, an [OSError](#) is raised.

In general, this function should be preferred over [os.getlogin\(\)](#).

Changed in version 3.13: Previously, various exceptions beyond just [OSError](#) were raised.