certbot.display.util module

- certbot.display.util module

certbot.display.util module

Certbot display.

This module (`certbot.display.util`) or its companion `certbot.display.ops` should be used whenever:

- Displaying status information to the user on the terminal
- Collecting information from the user via prompts

Other messages can use the `logging` module. See `log.py`.

certbot.display.util.OK = 'ok'
Display exit code indicating user acceptance.

certbot.display.util.CANCEL = 'cancel'
Display exit code for a user canceling the display.

certbot.display.util.notify(*msg: str*) → None[source]
Display a basic status message.

Parameters:

msg (*str*) – message to display

certbot.display.util.notification(*message: str*, *pause: bool = True*, *wrap: bool = True*, *force_interactive: bool = False*, *decorate: bool = True*) → None[source]
Displays a notification and waits for user acceptance.

Parameters:

- **message** (*str*) – Message to display
- **pause** (*bool*) – Whether or not the program should pause for the user's confirmation
- **wrap** (*bool*) – Whether or not the application should wrap text
- **force_interactive** (*bool*) – True if it's safe to prompt the user because it won't cause any workflow regressions
- **decorate** (*bool*) – Whether to surround the message with a decorated frame

certbot.display.util.menu(*message: str*, *choices: list[str] | list[tuple[str, str]]*, *default: int | None = None*, *cli_flag: str | None = None*, *force_interactive: bool = False*) → tuple[str, int][source]
Display a menu.

Parameters:

- **message** (*str*) – title of menu
- **choices** (*list of tuples (tag, item) or list of descriptions (tags will be enumerated)*) – Menu lines, len must be > 0
- **default** – default value to return, if interaction is not possible
- **cli_flag** (*str*) – option used to set this value with the CLI
- **force_interactive** (*bool*) – True if it's safe to prompt the user because it won't cause any workflow regressions

Returns:
tuple of (`code`, `index`) where `code` - str display exit code `index` - int index of the user's selection

Return type:

tuple

certbot.display.util.input_text(*message: str*, *default: str | None = None*, *cli_flag: str | None = None*, *force_interactive: bool = False*) → tuple[str, str][source]
Accept input from the user.

Parameters:

- **message** (*str*) – message to display to the user
- **default** – default value to return, if interaction is not possible
- **cli_flag** (*str*) – option used to set this value with the CLI
- **force_interactive** (*bool*) – True if it's safe to prompt the user because it won't cause any workflow regressions

Returns:
tuple of (`code`, `input`) where `code` - str display exit code `input` - str of the user's input

Return type:

tuple

certbot.display.util.yesno(*message: str*, *yes_label: str = 'Yes'*, *no_label: str = 'No'*, *default: bool | None = None*, *cli_flag: str | None = None*, *force_interactive: bool = False*) → bool[source]
Query the user with a yes/no question.

Yes and No label must begin with different letters, and must contain at least one letter each.

Parameters:

- **message** (*str*) – question for the user
- **yes_label** (*str*) – Label of the "Yes" parameter
- **no_label** (*str*) – Label of the "No" parameter
- **default** – default value to return, if interaction is not possible
- **cli_flag** (*str*) – option used to set this value with the CLI
- **force_interactive** (*bool*) – True if it's safe to prompt the user because it won't cause any workflow regressions

Returns:
True for "Yes", False for "No"

Return type:

bool

certbot.display.util.checklist(*message: str*, *tags: list[str]*, *default: list[str] | None = None*, *cli_flag: str | None = None*, *force_interactive: bool = False*) → tuple[str, list[str]][source]
Display a checklist.

Parameters:

- **message** (*str*) – Message to display to user
- **tags** (*list*) – `str` tags to select, len(tags) > 0
- **default** – default value to return, if interaction is not possible
- **cli_flag** (*str*) – option used to set this value with the CLI
- **force_interactive** (*bool*) – True if it's safe to prompt the user because it won't cause any workflow regressions

Returns:
tuple of (`code`, `tags`) where `code` - str display exit code `tags` - list of selected tags

Return type:

tuple

certbot.display.util.directory_select(*message: str*, *default: str | None = None*, *cli_flag: str | None = None*, *force_interactive: bool = False*) → tuple[str, str][source]
Display a directory selection screen.

Parameters:

- **message** (*str*) – prompt to give the user
- **default** – default value to return, if interaction is not possible
- **cli_flag** (*str*) – option used to set this value with the CLI
- **force_interactive** (*bool*) – True if it's safe to prompt the user because it won't cause any workflow regressions

Returns:
tuple of the form (`code`, `string`) where `code` - display exit code `string` - input entered by the user

certbot.display.util.assert_valid_call(*prompt: str*, *default: str*, *cli_flag: str*, *force_interactive: bool*) → None[source]
Verify that provided arguments is a valid display call.

Parameters:

- **prompt** (*str*) – prompt for the user
- **default** – default answer to prompt
- **cli_flag** (*str*) – command line option for setting an answer to this question
- **force_interactive** (*bool*) – if interactivity is forced

---

□□□ #3
□ □□□ □□□ 2026-10-31 13:02:08 UTC
□ cocobook-sync □□□ 2026-11-05 03:32:33 UTC