

```

#!/bin/bash
# shellcheck source=kz_common.sh
#####
# Show WiFi data.
#
# Written in 2022 by Karel Zimmer <info@karelzimmer.nl>, Creative Commons
# Public Domain Dedication <https://creativecommons.org/publicdomain/zero/1.0>.
#####

set -o errexit
program_path=$(cd "$(dirname "$(realpath "$0")")" && pwd)
source "$program_path"/kz_common.sh

#####
# Constants
#####

declare program_name='kz-wifi'
declare program_desc
program_desc=$(gettext 'Show WiFi data')
declare display_name=${program_name/kz-/kz }

declare usage
usage="$(eval_gettext "Usage: \${display_name} \${options_usage}" \
[-g|--gui]"
declare help
help="$(eval_gettext "Usage: \${display_name} [OPTION...]"
)$program_desc.

$(gettext 'Options:')
$options_help
  -g, --gui      $(gettext 'starts in graphics mode')"
declare options_short+='g'
declare options_long+=",gui"

#####
# Variables
#####

declare option_gui=false

#####
# Functions
#####

function check_input {
  local -i getopt_rc=0
  local parsed=''

  parsed=$(
    getopt --alternative          \
           --options              "$options_short" \
           --longoptions         "$options_long"  \
           --name                 "$display_name" \
           --                     "$@"
  ) || getopt_rc=$?
  if [[ $getopt_rc -ne $ok ]]; then
    info "$usage_line"
    exit $error
  fi
}

```

```

eval set -- "$parsed"
kz_common.process_options "$@"

while true; do
    case $1 in
        -g|--gui)
            option_gui=true
            kz_common.reset_terminal_attributes
            shift
            ;;
        --)
            shift
            break
            ;;
        *)
            shift
            ;;
    esac
done

if [[ "$*" ]]; then
    printf "$display_name: $*: %s\n$usage_line\n" \
        "$($gettext 'arguments are not allowed')"
    exit $error
fi
}

function process_input {
    local nmcli_output=''
    local -i nmcli_rc=''
    local text
    text=$(gettext "Record this data to connect to the wireless network \
later.")

    if $option_gui; then
        if nmcli_output=$(nmcli device wifi show-password 2> >($logcmd)); then
            info "$($gettext 'WiFi data:')"

$nmcli_output

$text"
        else
            info "$($gettext 'No WiFi device found.')"
        fi
    else
        info "$($gettext 'WiFi data:')"
        "
        nmcli device wifi show-password 2> >($logcmd) || nmcli_rc=$?
        if [[ $nmcli_rc -eq 0 ]]; then
            info "
$text"
        else
            info "$($gettext 'No WiFi device found.')"
        fi
    fi
}

function term_script {
    exit $ok
}

```

```
#####  
# Script  
#####  
  
function main {  
    kz_common.init_script "$@"  
    check_input "$@"  
    process_input  
    term_script  
}  
  
main "$@"
```