```
#!/bin/bash
# shellcheck source=common.sh
# ##############################################################################
# Bestand:  ckname
# Doel:     Controleer bestandsnaam(lengte)
# Gebruik:  In het terminalvenster:
#           /opt/kzscripts/ckname
#           of:
#           kzckname
# Auteur:   Karel Zimmer (https://karelzimmer.nl, info@karelzimmer.nl)
# ------------------------------------------------------------------------------
# Auteursrecht (c) 2012-2020 Karel Zimmer.
#
# Dit programma is vrije software: u mag het herdistribueren en/of wijzigen
# onder de voorwaarden van de GNU Algemene Publieke Licentie zoals gepubliceerd
# door de Free Software Foundation, onder versie 3 van de Licentie of (naar uw
# keuze) elke latere versie.
#
# Dit programma is gedistribueerd in de hoop dat het nuttig zal zijn maar ZONDER
# ENIGE GARANTIE; zelfs zonder de impliciete garanties die GEBRUIKELIJK ZIJN IN
# DE HANDEL of voor BRUIKBAARHEID VOOR EEN SPECIFIEK DOEL.
# Zie de GNU Algemene Publieke Licentie voor meer details.
#
# U hoort een kopie van de GNU Algemene Publieke Licentie te hebben ontvangen
# samen met dit programma. Als dat niet het geval is, zie
# http://www.gnu.org/licenses/.
# ##############################################################################
readonly REL_NUM=11.03.02
readonly REL_DAT=2020-04-03
readonly REL_MSG='Teksten gewijzigd'

# ##############################################################################
# Instellingen
# ##############################################################################
source "$(dirname "$(readlink --canonicalize "$0")")"/common.sh

# ------------------------------------------------------------------------------
# Globale constanten
# ------------------------------------------------------------------------------
declare -ir FILENAME_MAXLEN=142
readonly START_DFLT=$PWD

readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON
readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON
readonly OPTIONS_TAB_COMPLETION=$OPTIONS_TAB_COMPLETION_COMMON
readonly USAGE="Gebruik: $PROGNAME2 $OPTIONS_USAGE_COMMON
$OPTIONS_USAGE_FILLER[--] [MAP]

$OPTIONS_LONG_SHORT"
readonly HELP="Gebruik: $PROGNAME2 [OPTIE...] [--] [MAP]

Controleer bestandsnaam(lengte).

Opties:
  $OPTIONS_LONG_SHORT

$OPTIONS_HELP_COMMON

Argument:
  MAP                 Begin te controleren vanaf map MAP"

# ------------------------------------------------------------------------------
# Globale variabelen
# ------------------------------------------------------------------------------
declare ARGUMENT_MAP=false
declare MAP_ARGUMENT=''
declare START_MAP=''

# ##############################################################################
```

```
70  # Functies
71  # ##############################################################################
72  controleer_invoer() {
73      local -i getopt_rc=0
74      local parsed=''
75
76      set +o errexit
77      parsed=$(getopt --alternative                       \
78                      --options       "$OPTIONS_SHORT"    \
79                      --longoptions   "$OPTIONS_LONG"     \
80                      --name          "$PROGNAME2"        \
81                      -- "$@")
82      getopt_rc=$?
83      set -o errexit
84      if [[ $getopt_rc -ne 0 ]]; then
85          printf '%s\n' "$HELPLINE" >&2
86          quiet; exit $ERROR
87      fi
88      eval set -- "$parsed"
89      verwerk_algemene_opties "$@"
90
91      while true; do
92          case $1 in
93              --)
94                  shift
95                  break
96                  ;;
97              *)
98                  shift
99                  ;;
100         esac
101     done
102
103     while [[ "$*" ]]; do
104         if $ARGUMENT_MAP; then
105             printf "$PROGNAME2: %s\n%s\n" 'maximaal één argument opgeven' \
106 "$HELPLINE" >&2
107             quiet; exit $ERROR
108         else
109             ARGUMENT_MAP=true
110             MAP_ARGUMENT=$1
111         fi
112         shift
113     done
114
115     controleer_gebruiker
116 }
117
118 verwerk_invoer() {
119     if $ARGUMENT_MAP; then
120         START_MAP=$MAP_ARGUMENT
121     else
122         START_MAP=$START_DFLT
123     fi
124
125     if [[ ! -d $START_MAP ]]; then
126         printf "$PROGNAME2: %s\n%s\n" "map '$START_MAP' bestaat niet" \
127 "$HELPLINE" >&2
128         quiet; exit $ERROR
129     fi
130
131     zoek_bestanden_en_mappen
132 }
133
134 zoek_bestanden_en_mappen() {
135     local dirname
136     local file
137     local basename
138     local good
```

```
139       local -i count=0
140
141       # Het maakt niet uit of in de bestandsnaam speciale tekens voorkomen zoals
142       # tab, spatie, enz.  Hiervoor zorgt de find met print0,en de read met IFS=
143       # en als delimiter de null character die niet mag voorkomen in een
144       # bestandsnaam.  N.B.: In Linux is alles een bestand!
145       while IFS= read -r -d $'\0' file; do
146
147           dirname=$(dirname "$file")
148           basename=$(basename "$file")
149
150           # Verwijder de slechte tekens, \ is escape voor " en :
151           good=$(echo "$file" | tr --delete '?"\\<>*|:')
152
153           if [[ $file != "$good" ]]; then
154               (( ++count ))
155               if [[ -d "$file" ]]; then
156                   toon_tekst "BadName  DIR '$file'."
157               elif [[ -f "$file" ]]; then
158                   toon_tekst "BadName FILE '$basename' in map '$dirname'."
159               else
160                   toon_tekst "BadName SYML '$file'."
161               fi
162           fi
163
164           if [[ ${#basename} -gt $FILENAME_MAXLEN ]]; then
165               (( ++count ))
166               if [[ -d "$file" ]]; then
167                   toon_tekst "BadLen.  DIR '$basename'."
168               else
169                   toon_tekst "BadLen. FILE '$basename' in map '$dirname'."
170               fi
171           fi
172
173       done < <(find "$START_MAP" -type f -print0 -or -type d -print0 -or -type l \
174   -print0)
175
176       if [[ $count -eq 0 ]]; then
177           toon_succestekst "Er zijn geen fouten gevonden."
178       elif [[ $count -eq 1 ]]; then
179           toon_waarschuwingstekst 'Er is één fout gevonden.'
180       else
181           toon_waarschuwingstekst "Er zijn $count fouten gevonden."
182       fi
183   }
184
185   term_script() {
186       quiet; exit $SUCCESS
187   }
188
189   verwerk_script() {
190       controleer_invoer "$@"
191       verwerk_invoer
192   }
193
194   # ############################################################################
195   # Hoofdlijn
196   # ############################################################################
197   main() {
198       init_script
199       verwerk_script "$@"
200       term_script
201   }
202
203   main "$@"
204
205   # Einde script
```