

```
1  #!/bin/bash
2  # shellcheck source=kzcommon.sh
3  #-----#
4  # Webbestanden valideren.                                     #
5  #                                                            #
6  # Geschreven door Karel Zimmer <info@karelzimmer.nl>.      #
7  #                                                            #
8  # Auteursrecht (c) 2012-2020 Karel Zimmer.                  #
9  # GNU Algemene Publieke Licentie <https://www.gnu.org/licenses/gpl.html>. #
10 #                                                            #
11 # RelNum=11.00.20                                           #
12 # RelDat=2020-11-16                                         #
13 #-----#
14
15 #-----#
16 # Global constants                                           #
17 #-----#
18 source "$(dirname "$0")"/kzcommon.sh
19 readonly CSS_DIR=$HOME/uploads/karelzimmer.nl/httpdocs/css
20 readonly HTML_DIR=$HOME/uploads/karelzimmer.nl/httpdocs/html
21
22 readonly HOME_URL=https://karelzimmer.nl
23 readonly CSS_URL=$HOME_URL/css
24 readonly HTML_URL=$HOME_URL/html
25
26 readonly CSS_VALIDATOR_URL='http://jigsaw.w3.org/css-validator/validator?uri='
27 readonly HTML_VALIDATOR_URL='http://validator.w3.org/check?uri='
28
29 # Bij aanpassingen ook .completion aanpassen!
30 readonly OPTIONS_SHORT=$OPTIONS_SHORT_COMMON
31 readonly OPTIONS_LONG=$OPTIONS_LONG_COMMON
32 readonly USAGE="Gebruik: $PROGNAME $OPTIONS_USAGE_COMMON"
33
34 $OPTIONS_LONG_SHORT"
35 readonly HELP="Gebruik: $PROGNAME [OPTIE...]"
36
37 Webbestanden valideren.
38
39 Opties:
40     $OPTIONS_LONG_SHORT
41
42 $OPTIONS_HELP_COMMON"
43
44 #-----#
45 # Global variables                                           #
46 #-----#
47
48 #-----#
49 # Functions                                                  #
50 #-----#
51 check_input() {
52     local -i getopt_rc=0
53     local parsed=''
54
55     parsed=$(
56         getopt --alternative          \
57                --options              "$OPTIONS_SHORT"    \
58                --longoptions         "$OPTIONS_LONG"     \
59                --name                 "$PROGNAME"        \
60                -- "$@"
61     ) || getopt_rc=$?
62     if [[ $getopt_rc -ne 0 ]]; then
63         printf '%s\n' "$USAGELINE" >&2
64         QUIET=true exit $ERROR
65     fi
66     eval set -- "$parsed"
```

```
67     process_general_options "$@"
68
69     while true; do
70         case $1 in
71             --)
72                 shift
73                 break
74                 ;;
75             *)
76                 shift
77                 ;;
78         esac
79     done
80
81     if [[ "$*" ]]; then
82         printf "$PROGNAME: %s\n%s\n" 'geen argumenten opgeven' "$USAGELINE" >&2
83         QUIET=true exit $ERROR
84     fi
85
86     # Een non-gui script gestart met optie gui.
87     if $OPTION_GUI; then
88         OPTION_GUI=false
89         TERMINAL=true
90     fi
91
92     check_user
93 }
94
95 process_input() {
96     local dir=''
97
98     for dir in "$CSS_DIR" "$HTML_DIR"; do
99         if ! [[ -d $dir ]]; then
100             printf '%s\n' "Map '$dir' bestaat niet." >&2
101             QUIET=true exit $ERROR
102         fi
103     done
104
105     if ! pgrep 'chrome' &> /dev/null; then
106         warning 'Google Chrome webbrowser is nog niet gestart,
107 start nu eerst de Google Chrome webbrowser.'
108         read -rp "Druk op Enter-toets wanneer Google Chrome webbrowser is \
109 gestart [Enter]: " < /dev/tty
110         if ! pgrep 'chrome' &> /dev/null; then
111             error 'Google Chrome webbrowser is niet gestart.'
112             QUIET=true exit $ERROR
113         fi
114     fi
115
116     validate_css
117     validate_html
118 }
119
120 validate_css() {
121     local filename=''
122
123     cd "$CSS_DIR"
124
125     for filename in *.css; do
126         google-chrome "$CSS_VALIDATOR_URL$CSS_URL/$filename" |& $LOGCMD
127     done
128 }
129
130 validate_html() {
131     local filename=''
132
```

```
133     cd "$HTML_DIR"
134
135     for filename in *.html; do
136         google-chrome "$HTML_VALIDATOR_URL$HTML_URL/$filename" |& $LOGCMD
137     done
138 }
139
140 term_script() {
141     info 'Controleer de tabbladen in Google Chrome.'
142     QUIET=true exit $SUCCESS
143 }
144
145 #-----#
146 # Main line                                     #
147 #-----#
148 main() {
149     init_script
150     check_input "$@"
151     process_input
152     term_script
153 }
154
155 main "$@"
156
157 # EOF
```