

```
1 #!/bin/bash
2 # shellcheck source=kzcommon.sh
3 #-----#
4 # Apps installeren. #
5 # #
6 # Geschreven door Karel Zimmer <info@karelzimmer.nl>. #
7 # #
8 # Auteursrecht (c) 2009-2020 Karel Zimmer. #
9 # GNU Algemene Publieke Licentie <https://www.gnu.org/licenses/gpl.html>. #
10 # #
11 # RelNum=55.08.04 #
12 # RelDat=2020-11-23 #
13 #-----#
14
15 #-----#
16 # Global constants #
17 #-----#
18 source "$(dirname "$0")"/kzcommon.sh
19 readonly BEGIN_INFO="\
20 $LINE1
21 $LINEV Volg onderstaande aanvullende instructies na het installeren van deze a\
22 pps $LINEV
23 $LINEH2"
24 readonly END_INFO="
25 $LINE1
26 $LINEV Volg bovenstaande aanvullende instructies na het installeren van deze a\
27 pps $LINEV
28 $LINEH2"
29 readonly BEGIN_REMOVE_INFO="\
30 $LINE1
31 $LINEV Volg onderstaande instructies voor het verwijderen van deze apps \
32 $LINEV
33 $LINEH2"
34 readonly END_REMOVE_INFO="
35 $LINE1
36 $LINEV Volg bovenstaande instructies voor het verwijderen van deze apps \
37 $LINEV
38 $LINEH2"
39 readonly EDITION_DEFAULT=desktop
40 readonly RUN_AS_SUPERUSER=true
41
42 # Bij aanpassingen ook .completion aanpassen!
43 readonly OPTIONS_SHORT=${OPTIONS_SHORT_COMMON}'ace:firs'
44 readonly OPTIONS_LONG=${OPTIONS_LONG_COMMON}',apps,cat,edition:,files,info,\
45 remove-info,simulate"
46 readonly USAGE="Gebruik: $PROGRAMME [-a|--apps] [-c|--cat] \
47 [-e|--edition=EDITIE] [-f|--files]
48 [-i|--info] [-r|--remove-info] [-s|--simulate]
49 $OPTIONS_USAGE_COMMON
50 [APP...] [BESTAND...]"
51
52 $OPTIONS_LONG_SHORT
53
54 $ARGUMENT_LONG_SHORT"
55 readonly HELP="Gebruik: $PROGRAMME [OPTIE...] [--] [APP...] [BESTAND...]"
56
57 Apps installeren.
58
59 Opties:
60 $OPTIONS_LONG_SHORT
61
62 $ARGUMENT_LONG_SHORT
63
64 -a --apps Toon lijst van te installeren apps
65 -c --cat Toon inhoud van installatiebestanden
66 -e --edition=EDITIE Gebruik opgegeven editie
```

```
67     -f --files           Toon lijst van alle installatiebestanden
68     -i --info           Toon aanvullende instructies na het installeren van de apps
69     -r --remove-info   Toon instructies voor het verwijderen van de apps
70     -s --simulate      Geen actie, simuleer installatie, en toon opdracht die
71                       uitgevoerd zou worden
72 $OPTIONS_HELP_COMMON
73
74 Argumenten:
75     APP                Opgegeven apps verwerken
76     BESTAND           Opgegeven bestanden verwerken"
77
78 #-----#
79 # Global variables                                     #
80 #-----#
81 declare -a APP_ARGUMENT=''
82 declare -a FILE_ARGUMENT=''
83 declare -a INPUTFILE=''
84 declare ARGUMENT_APP=false
85 declare ARGUMENT_FILE=false
86 declare COMMANDS_FOUND=false
87 declare DFLTFILE_1=''
88 declare DFLTFILE_2=''
89 declare DFLTFILE_NOTFOUND=''
90 declare DISTR0=''
91 declare EDITION_ARGUMENT=''
92 declare EDITION=$EDITION_DEFAULT
93 declare EXECUTE_COMMANDS=true
94 declare -i MAXRC=0
95 declare INFO_FOUND=false
96 declare OPTION_APPS=false
97 declare OPTION_CAT=false
98 declare OPTION_EDITION=false
99 declare OPTION_FILES=false
100 declare OPTION_INFO=false
101 declare OPTION_REMOVE_INFO=false
102 declare OPTION_SIMULATE=false
103 declare REMOVE_INFO_FOUND=false
104 declare TEMP_TEXT_FILE=''
105
106 #-----#
107 # Functions                                           #
108 #-----#
109 check_input() {
110     local file_notfound=false
111     local -i app_arg_num=0
112     local id=''
113     local -i file_arg_num=0
114     local -i file_num=0
115     local -i getopt_rc=0
116     local parsed=''
117     local release=''
118
119     parsed=$(
120         getopt --alternative          \
121                --options             "$OPTIONS_SHORT" \
122                --longoptions        "$OPTIONS_LONG"  \
123                --name                "$PROGNAME"     \
124                -- "$@"
125     ) || getopt_rc=$?
126     if [[ $getopt_rc -ne 0 ]]; then
127         printf '%s\n' "$USAGELINE" >&2
128         QUIET=true exit $ERROR
129     fi
130     eval set -- "$parsed"
131     process_general_options "$@"
132 }
```

```
133     while true; do
134         case $1 in
135             -a|--apps)
136                 OPTION_APPS=true
137                 shift
138                 ;;
139             -c|--cat)
140                 OPTION_CAT=true
141                 shift
142                 ;;
143             -e|--edition)
144                 if $OPTION_EDITION; then
145                     printf "$PROGNAME: %s\n%s\n" \
146                         "optie '$1' éénmaal opgeven" "$USAGELINE" >&2
147                     QUIET=true exit $ERROR
148                 else
149                     OPTION_EDITION=true
150                     EDITION_ARGUMENT=$2
151                     EDITION=$EDITION_ARGUMENT
152                 fi
153                 shift 2
154                 ;;
155             -f|--files)
156                 OPTION_FILES=true
157                 shift
158                 ;;
159             -i|--info)
160                 OPTION_INFO=true
161                 EXECUTE_COMMANDS=false
162                 shift
163                 ;;
164             -r|--remove-info)
165                 OPTION_REMOVE_INFO=true
166                 EXECUTE_COMMANDS=false
167                 shift
168                 ;;
169             -s|--simulate)
170                 OPTION_SIMULATE=true
171                 EXECUTE_COMMANDS=false
172                 shift
173                 ;;
174             --)
175                 shift
176                 break
177                 ;;
178             *)
179                 shift
180                 ;;
181         esac
182     done
183
184     id=$(lsb_release --id --short | tr '[:upper:]' '[:lower:]')
185     release=$(lsb_release --release --short)
186     DISTRO=$id-$release-$EDITION
187     DFLTFILE_1=$PROGDIR/$PROGNAME-$DISTRO.sh
188     DFLTFILE_2=$PROGDIR/$PROGNAME-$DISTRO-$HOSTNAME.sh
189     DFLTFILE_NOTFOUND="Geen bestanden zoals $PROGNAME-$DISTRO*.sh in $PROGDIR."
190
191     # Verwerk argumenten.
192     while [[ "$*" ]]; do
193         if [[ "$(basename "$1")" = $PROGNAME-*.sh ]]; then
194             ARGUMENT_FILE=true
195             FILE_ARGUMENT[$file_arg_num]=$1
196             (( ++file_arg_num ))
197             shift
198         else
```

```

199         ARGUMENT_APP=true
200         APP_ARGUMENT[$app_arg_num]=$1
201         (( ++app_arg_num ))
202         shift
203     fi
204 done
205 if $ARGUMENT_APP; then
206     process_argument_app
207 fi
208 if $ARGUMENT_FILE; then
209     process_argument_file
210 fi
211 # Als APP of FILE is opgegeven, de DFLTFILES niet verwerken.
212 if ! ($ARGUMENT_APP || $ARGUMENT_FILE); then
213     process_dfltfiles
214 fi
215 if $file_notfound; then
216     QUIET=true exit $SUCCESS
217 fi
218
219 # Verwerk opties.
220 if $OPTION_APPS; then
221     process_option_apps
222     QUIET=true exit $SUCCESS
223 elif $OPTION_FILES; then
224     process_option_files
225     QUIET=true exit $SUCCESS
226 # Optie cat is afhankelijk van argumenten APP en FILE.
227 elif $OPTION_CAT; then
228     process_option_cat
229     QUIET=true exit $SUCCESS
230 fi
231
232 # We gaan opdrachten uitvoeren, mogen we dat wel?
233 if $EXECUTE_COMMANDS; then
234     check_user
235     check_on_ac_power
236 fi
237 }
238
239 process_option_apps() {
240     local temp_text_file
241
242     temp_text_file=$(mktemp -t "$PROGNAME-XXXXXXXXXX.txt")
243
244     # shellcheck disable=SC2062
245     printf '%s\n' "De volgende apps kunnen geïnstalleerd worden:
246     NUMMER APP
247     $(
248         if ! grep --regexp='^#@' \
249             --no-messages \
250             "$PROGDIR/$PROGNAME-$DISTRO"*.sh |
251             awk -F#@ \
252             '{print $2}' |
253             sort --unique |
254             nl --number-width=6 \
255             --number-format=rn \
256             --number-separator=' ' \
257             --body-numbering=a; then
258             printf '%s\n' "      0 Geen apps gevonden.
259
260     $DFLTFILE_NOTFOUND"
261     else
262         printf '%s\n' "
263     De gevonden apps zijn te gebruiken als invoer voor dit script:
264     ${BLUE}$PROGNAME APP...${NORMAL}

```

```

265 Voor meer informatie voer uit:
266   ${BLUE}$PROGNAME --sim APP...${NORMAL}
267   ${BLUE}$PROGNAME --cat APP...${NORMAL}"
268   fi
269 )" > "$temp_text_file"
270
271   if $OPTION_GUI; then
272     remove_non_printables "$temp_text_file"
273     # Constructie '2> >($LOGCMD)' om stderr naar de log te krijgen.
274     # Voorbeeld: Unable to init server: Kon niet verbinden:
275     #               Verbinding is geweigerd
276     #               en: (zenity:47712): Gtk-WARNING **: 10:35:49.339:
277     #               cannot open display:
278     if ! zenity --text-info                                \
279               --width      550                            \
280               --height     600                            \
281               --title      'Overzicht apps'               \
282               --filename   "$temp_text_file"              \
283               --font       'Ubuntu Mono 12'              \
284               --ok-label   'Oké'                          \
285               --cancel-label 'Annuleren'                 \
286               2> >($LOGCMD); then
287       true
288     fi
289   else
290     less "$LESS_OPTIONS" "$temp_text_file"
291   fi
292
293   rm "$temp_text_file"
294 }
295
296 process_option_files() {
297   local temp_text_file
298
299   temp_text_file=$(mktemp -t "$PROGNAME-XXXXXXXXX.txt")
300
301   # shellcheck disable=SC2012
302   printf '%s\n' "De volgende installatiebestanden zijn beschikbaar:
303 NUMMER BESTAND
304 $(
305   if ! ls --format=single-column                            \
306         "$PROGDIR/$PROGNAME-$DISTRO"*.sh                  \
307         2> /dev/null                                       |
308         nl --number-width=6                                \
309           --number-format=rn                               \
310           --number-separator=' '                          \
311           --body-numbering=a; then                          \
312     printf '%s\n' "      0 Geen bestanden gevonden.
313
314 $DFLTFILE_NOTFOUND"
315   else
316     printf '%s\n' "
317 De gevonden bestanden zijn te gebruiken als invoer voor dit script:
318   ${BLUE}$PROGNAME BESTAND...${NORMAL}
319 Voor meer informatie voer uit:
320   ${BLUE}$PROGNAME --sim BESTAND...${NORMAL}
321   ${BLUE}$PROGNAME --cat BESTAND...${NORMAL}"
322   fi
323 )" > "$temp_text_file"
324
325   if $OPTION_GUI; then
326     remove_non_printables "$temp_text_file"
327     if ! zenity --text-info                                \
328               --width      750                            \
329               --height     600                            \
330               --title      'Overzicht bestanden'          \

```

```

331         --filename      "$temp_text_file"      \
332         --font          'Ubuntu Mono 12'      \
333         --ok-label      'Oké'                 \
334         --cancel-label  'Annuleren'           \
335         2> >($LOGCMD); then
336             true
337         fi
338     else
339         less "$LESS_OPTIONS" "$temp_text_file"
340     fi
341
342     rm "$temp_text_file"
343 }
344
345 process_argument_app() {
346     local app=''
347     local file=''
348
349     for app in "${APP_ARGUMENT[@]"; do
350         # Zoek exact (line-regex) naar app-tag in installatiebestanden.
351         # Alleen eerstgevonden bestand verwerken (lines=1), want tag kan in
352         # meerdere bestanden voorkomen.
353         # shellcheck disable=SC2062
354         file=$(
355             if ! grep --files-with-matches \
356                 --line-regexp \
357                 --regexp="#@$app" \
358                 "$PROGDIR/$PROGNAME-$DISTRO"*.sh \
359                 2> /dev/null \
360                 head --lines=1; then
361                 printf '%s' ''
362             fi
363         )
364         if [[ $file ]]; then
365             process_argument_app_file
366         else
367             info "App '$app' niet gevonden."
368             file_notfound=true
369         fi
370     done
371     if $file_notfound; then
372         info "Typ 'kzsetup --apps' voor meer informatie."
373     fi
374 }
375
376 process_argument_app_file() {
377     local app_found=false
378     local description_found=false
379     local record=''
380     local temp_sh=''
381
382     temp_sh=$(mktemp -t "$PROGNAME-$app-XXXXXXXXXX.sh")
383     while read -r record; do
384         case $record in
385             '#@'*)
386                 if [[ $record = '#@$app' ]]; then
387                     if app_found; then
388                         info "Dubbele app-tag $app."
389                     else
390                         # Gezochte app-tag.
391                         printf '%b\n' \
392                             "# $LINEH\n# Bestand: $temp_sh" \
393                             >> "$temp_sh"
394                         # Regels doorschrijven naar tijdelijk bestand tot
395                         # eerstvolgende app-tag of beschrijving (verplicht), of
396                         # tot EOF.

```

```

397         app_found=true
398     fi
399     elif $app_found; then
400         # Volgende app-tag.
401         break
402     fi
403     ;;
404     '#1'*)
405     # Beschrijving.
406     if $description_found; then
407         # Volgende beschrijving.
408         break
409     elif $app_found; then
410         printf '%b\n' \
411             "#   Bron: $file\n# $LINEH\n\n$record" \
412             >> "$temp_sh"
413         description_found=true
414     fi
415     ;;
416 *)
417     # Overige regels.
418     if $app_found; then
419         printf '%s\n' "$record" >> "$temp_sh"
420     fi
421     ;;
422 esac
423 done < "$file"
424 if ! grep --quiet \
425     --line-regexp \
426     --regexp='# EOF' "$temp_sh"
427 then
428     printf '%s\n' '# EOF' >> "$temp_sh"
429 fi
430 INPUTFILE[$file_num]=$temp_sh
431 (( ++file_num ))
432 }
433
434 process_argument_file() {
435     local file=' '
436
437     for file in "${FILE_ARGUMENT[@]}; do
438         if [[ -e $file ]]; then
439             INPUTFILE[$file_num]=$file
440             (( ++file_num ))
441         else
442             info "Bestand '$file' niet gevonden."
443             file_notfound=true
444         fi
445     done
446     if $file_notfound; then
447         info "Typ 'kzsetup --files' voor meer informatie."
448     fi
449 }
450
451 process_dflfiles() {
452     if [[ -e $DFLTFILE_1 ]]; then
453         INPUTFILE[$file_num]=$DFLTFILE_1
454         (( ++file_num ))
455     else
456         file_notfound=true
457     fi
458     # Niet verplicht aanwezig.
459     if [[ -e $DFLTFILE_2 ]]; then
460         INPUTFILE[$file_num]=$DFLTFILE_2
461         (( ++file_num ))
462     fi

```

```
463     if $file_notfound; then
464         info "$DFLTFILE_NOTFOUND"
465     fi
466 }
467
468 process_option_cat() {
469     local temp_text_file
470
471     temp_text_file=$(mktemp -t "$PROGNAME-XXXXXXXXXX.txt")
472
473     for file in "${INPUTFILE[@]}; do
474     {
475         printf '%s\n' "${BLUE}# $file${NORMAL}"
476         cat "$file"
477     } >> "$temp_text_file"
478     done
479
480     if $OPTION_GUI; then
481         remove_non_printables "$temp_text_file"
482         if ! zenity --text-info \
483             --width 1100 \
484             --height 600 \
485             --title 'Inhoud bestanden' \
486             --filename "$temp_text_file" \
487             --font 'Ubuntu Mono 12' \
488             --ok-label 'Oké' \
489             --cancel-label 'Annuleren' \
490             2> >($LOGCMD); then
491             true
492         fi
493     else
494         less "$LESS_OPTIONS" "$temp_text_file"
495     fi
496
497     rm "$temp_text_file"
498 }
499
500 process_input() {
501     local file=''
502     local -i app_seq_num=0
503     local -i app_tot_num=0
504     local text=''
505
506     TEMP_TEXT_FILE=$(mktemp -t "$PROGNAME-XXXXXXXXXX.txt")
507
508     for file in "${INPUTFILE[@]}; do
509         if $OPTION_SIMULATE || $EXECUTE_COMMANDS; then
510             app_seq_num=0
511             # 'if ! grep --regex='^#1' ...; then ...' werkt niet;
512             # grep geeft 0 terug als niets gevonden, OK, maar dan ook rc=1, -OK.
513             app_tot_num=$(
514                 grep --word-regexp \
515                     --regex='^#1' \
516                     --count "$file" ||
517                 true
518             )
519         fi
520
521         if $EXECUTE_COMMANDS && $OPTION_GUI; then
522             title='Apps installeren'
523             text="Verwerk bestand $file"
524             # Met '|& zenity --progress' worden globale variabelen uit
525             # aangeroepen functies niet doorgegeven, vandaar de
526             # 'process substitution' met "> >(zenity ...)".
527             # Zenity progress-informatie per opdracht (bij setup per bestand).
528             process_file "$file" > >(
```



```
529         zenity --progress          \  
530                --pulsate           \  
531                --auto-close        \  
532                --no-cancel         \  
533                --width 700         \  
534                --height 50         \  
535                --title "$title"    \  
536                --text "$text"      \  
537                2> >($LOGCMD)      \  
538     ) \  
539     else \  
540         process_file "$file" \  
541     fi \  
542 done \  
543 } \  
544 \  
545 process_file() { \  
546     local file=${1:-file?} \  
547     local description='### GEEN BESCHRIJVING ###' \  
548     local file_text='' \  
549     local -i cmd_seq_num=0 \  
550     local prev_recordtype='' \  
551     local record='' \  
552     local recordtype='' \  
553     local write_description_line=true \  
554 \  
555     file_text="${BOLD}[BEGIN] Bestand $file${NORMAL}" \  
556     if $EXECUTE_COMMANDS; then \  
557         log "$file_text" \  
558         printf '%b' "${CURSOR_INVISIBLE}" \  
559         printf '%s\n' "$file_text" \  
560     elif $OPTION_SIMULATE; then \  
561         printf '%s\n' "$file_text" >> "$TEMP_TEXT_FILE" \  
562     fi \  
563 \  
564     while read -r record; do \  
565         recordtype=${record:0:2} \  
566         case $recordtype in \  
567             '#1') \  
568                 process_description "$record" \  
569                 ;; \  
570             '#2') \  
571                 process_install_instruction "$record" \  
572                 ;; \  
573             '#3') \  
574                 process_remove_instruction "$record" \  
575                 ;; \  
576             '|#'|#*) \  
577                 continue \  
578                 ;; \  
579             *) \  
580                 process_command "$record" \  
581                 ;; \  
582             esac \  
583         prev_recordtype=$recordtype \  
584     done < "$file" \  
585 \  
586     file_text="${BOLD}[EINDE] Bestand $file${NORMAL}" \  
587     if $EXECUTE_COMMANDS; then \  
588         log "$file_text" \  
589         printf '%s\n' "$file_text" \  
590         printf '%b' "${CURSOR_VISIBLE}" \  
591     elif $OPTION_SIMULATE; then \  
592         printf '%s\n' "$file_text" >> "$TEMP_TEXT_FILE" \  
593     fi \  
594 }
```

```

595
596 process_description() {
597     local record=${1:-record?}
598
599     description="${record:3}"
600     if ! $OPTION_SIMULATE && ! $EXECUTE_COMMANDS; then
601         return 0
602     fi
603
604     write_description_line=true
605     (( ++app_seq_num ))
606     cmd_seq_num=0
607 }
608
609 process_install_instruction() {
610     local record=${1:-record?}
611
612     INFO_FOUND=true
613     if ! $OPTION_INFO; then
614         return 0
615     fi
616     if [[ $app_seq_num -eq 0 ]]; then
617         printf '%s\n' "$BEGIN_INFO" > "$TEMP_TEXT_FILE"
618     fi
619     if [[ $prev_recordtype = '#2' ]]; then
620         printf '%s\n' "    ${record:3}" >> "$TEMP_TEXT_FILE"
621     else
622         (( ++app_seq_num ))
623         printf "\n%2s. %s\n          %s\n"                \
624                "$app_seq_num"                            \
625                "${UNDERLINE}$description${NORMAL}"     \
626                "${record:3}"                             >> "$TEMP_TEXT_FILE"
627     fi
628 }
629
630 process_remove_instruction() {
631     local record=${1:-record?}
632
633     REMOVE_INFO_FOUND=true
634     if ! $OPTION_REMOVE_INFO; then
635         return 0
636     fi
637     if [[ $app_seq_num -eq 0 ]]; then
638         printf '%s\n' "$BEGIN_REMOVE_INFO" > "$TEMP_TEXT_FILE"
639     fi
640     if [[ $prev_recordtype = '#3' ]]; then
641         printf '%s\n' "    ${record:3}" >> "$TEMP_TEXT_FILE"
642     else
643         (( ++app_seq_num ))
644         printf "\n%2s. %s\n          %s\n"                \
645                "$app_seq_num" "$${UNDERLINE}$description${NORMAL}" \
646                "${record:3}"  >> "$TEMP_TEXT_FILE"
647     fi
648 }
649
650 process_command() {
651     local cmd=${1:-cmd?}
652     local -i cmd_rc=0
653     local description_line=''
654     local -i description_maxlen=1000
655     local description_printf=''
656     local status=''
657
658     COMMANDS_FOUND=true
659     (( ++cmd_seq_num ))
660

```

```

661     if $OPTION_SIMULATE; then
662         if $write_description_line; then
663             printf "[%2d/%-2d] %s\n" \
664 "$app_seq_num" "$app_tot_num" "$description" >> "$TEMP_TEXT_FILE"
665         fi
666         write_description_line=false
667         printf '%7s [%2d] %s\n'          \
668             " " "$cmd_seq_num" "${BLUE}$record${NORMAL}"          \
669             >> "$TEMP_TEXT_FILE"
670     elif $EXECUTE_COMMANDS; then
671         description_printf=$description
672         if $OPTION_GUI; then
673             printf '%s\n' \
674                 "#Bestand:\t$file\n\nVerwerk:\t[$app_seq_num/$app_tot_num] \
675 $description\n\nOpdracht:\t$record"
676         else
677             # Afbreken in Terminalvenster moet i.v.m. updaten status van BEZIG
678             # naar GOED of FOUT.
679             # <----- tput cols ----->
680             # '[BEZIG] [ 1/1 ] [ 1] Te lange descr...' <=== moet 1 regel blijven
681             # <----- 21 ----->
682             description_maxlen=$(( (tput cols) - 21 ))
683             if [[ $#description_printf -gt $description_maxlen ]; then
684                 description_printf=\
685                 ${description_printf:0:((($description_maxlen - 3))}'...}'
686             fi
687         fi
688         printf -v description_line          \
689             "[%2d/%-2d] [%2d] %-${description_maxlen}.\ \
690 ${description_maxlen}s"          \
691             "$app_seq_num" "$app_tot_num" "$cmd_seq_num" \
692 "$description_printf"
693         execute_command "$record"
694     fi
695 }
696
697 execute_command() {
698     local cmd=${1:-cmd?}
699     local -i cmd_rc=0
700     local status=''
701
702     check_dpkg_available
703     if ! $OPTION_GUI; then
704         if [[ $cmd_seq_num -eq 1 ]; then
705             printf " $STATUS_BUSY %s\r" "$description_line"
706         else
707             printf " $STATUS_BUSY          %s\r" "${description_line:8}"
708         fi
709     fi
710
711     log "$description_line"
712     log "${BLUE}$cmd${NORMAL}"
713     eval "$cmd" |& $LOGCMD || cmd_rc=$?
714     if [[ $cmd_rc -gt $MAXRC ]; then
715         MAXRC=$cmd_rc
716     fi
717     if [[ $cmd_rc -eq 0 ]; then
718         status="$STATUS_SUCCESS"
719     else
720         status="$STATUS_ERROR"
721     fi
722
723     printf ' %s\n' "$status"
724     log "$status rc=$cmd_rc, maxrc=$MAXRC"
725 }
726

```

```
727 check_dpkg_available() {
728     local -i wait_for_apt=10
729     local text='Wacht totdat andere softwarebeheerders klaar zijn ...'
730
731     if ls /snap/core/*/var/cache/debconf/config.dat && /dev/null; then
732         # shellcheck disable=SC2024
733         while sudo fuser
734             /var/{lib/{dpkg,apt/lists},cache/apt/archives}/lock \
735             /var/cache/debconf/config.dat \
736             /snap/core/*/var/cache/debconf/config.dat \
737             && /dev/null; do
738             info "$text"
739             sleep $wait_for_apt
740         done
741     else
742         # shellcheck disable=SC2024
743         while sudo fuser
744             /var/{lib/{dpkg,apt/lists},cache/apt/archives}/lock \
745             /var/cache/debconf/config.dat \
746             && /dev/null; do
747             info "$text"
748             sleep $wait_for_apt
749         done
750     fi
751 }
752
753 term_script() {
754     local title=''
755
756     if $OPTION_SIMULATE; then
757         if $OPTION_GUI; then
758             title='Simuleer installatie'
759             remove_non_printables "$TEMP_TEXT_FILE"
760             if ! zenity --text-info
761                 --width 1100 \
762                 --height 600 \
763                 --title "$title" \
764                 --filename "$TEMP_TEXT_FILE" \
765                 --font 'Ubuntu Mono 12' \
766                 --ok-label 'Oké' \
767                 --cancel-label 'Annuleren' \
768                 2> >($LOGCMD); then
769                 true
770             fi
771         else
772             less "$LESS_OPTIONS" "$TEMP_TEXT_FILE"
773         fi
774         QUIET=true
775     elif $OPTION_INFO && $INFO_FOUND; then
776         printf '%s\n' "$END_INFO" >> "$TEMP_TEXT_FILE"
777         if $OPTION_GUI; then
778             title='Installatie-instructies'
779             if ! zenity --text-info
780                 --width 850 \
781                 --height 400 \
782                 --title "$title" \
783                 --filename "$TEMP_TEXT_FILE" \
784                 --font 'Ubuntu Mono 12' \
785                 --ok-label 'Oké' \
786                 --cancel-label 'Annuleren' \
787                 2> >($LOGCMD); then
788                 true
789             fi
790         else
791             less "$LESS_OPTIONS" "$TEMP_TEXT_FILE"
792         fi
793     fi
794 }
```

```
793     QUIET=true
794     elif $OPTION_INFO && ! $INFO_FOUND; then
795         info 'Geen instructies aanwezig.'
796     elif $OPTION_REMOVE_INFO && $REMOVE_INFO_FOUND; then
797         printf '%s\n' "$END_REMOVE_INFO" >> "$TEMP_TEXT_FILE"
798         if $OPTION_GUI; then
799             title='Verwijder-instructies'
800             if ! zenity --text-info \
801                 --width 850 \
802                 --height 400 \
803                 --title "$title" \
804                 --filename "$TEMP_TEXT_FILE" \
805                 --font 'Ubuntu Mono 12' \
806                 --ok-label 'Oké' \
807                 --cancel-label 'Annuleren' \
808                 2> >($LOGCMD); then
809                 true
810             fi
811         else
812             less "$LESS_OPTIONS" "$TEMP_TEXT_FILE"
813         fi
814     QUIET=true
815     elif $OPTION_REMOVE_INFO && ! $REMOVE_INFO_FOUND; then
816         info 'Geen instructies aanwezig.'
817     elif ! $COMMANDS_FOUND; then
818         info 'Geen opdrachten aanwezig.'
819     fi
820     rm "$TEMP_TEXT_FILE"
821
822     if $EXECUTE_COMMANDS &&
823        [[ $MAXRC -eq 0 ]] &&
824        $INFO_FOUND &&
825        ! $OPTION_GUI; then
826         info "Er zijn aanvullende instructies, gebruik de opdracht:
827     ${BLUE}$PROGNAME --info${NORMAL}"
828     fi
829
830     if $EXECUTE_COMMANDS && [[ $MAXRC -ne 0 ]]; then
831         exit $WARNING
832     else
833         exit $SUCCESS
834     fi
835 }
836
837 #-----#
838 # Main line #
839 #-----#
840 main() {
841     init_script
842     check_input "$@"
843     process_input
844     term_script
845 }
846
847 main "$@"
848
849 # EOF
```