

Fonctions shell utiles pour tout script



Sommaire

- [Objectif](#)
- [Fonctions](#)
- [Alias utiles](#)

Objectif

Mettre ici les fonctions couramment utilisées dans les scripts.

Fonctions

fonctions_communes.sh

```
# fonctions

# fonction: die ()
# au moins 1 paramètres : le code erreur sortie
#
# usage:
#
# $ die <numéro d'erreur> [<messages d'erreur>+]
#
# cette fonction retourne le code erreur donné en
# premier paramètre et affiche le message sur la
# sortie erreur standard en deuxième et plus paramètre.
die ()
{
    [ $# -ge 1 ] || return 2
    err_code="$1"
    shift
    printf '%s\n' "$*" 1>&2
    exit $err_code
}
```

```

# fonction: espsqousligneminuscule ()
# au moins un paramètre : la chaîne à transformer.
#
# usage:
#
# $ espsouligneminuscule "Une chaîne avec des caractères bizarres et des accents"
#
# fonction utilitaire qui remplace par exemple un titre avec des espaces
# peu pratique pour un nom de répertoire en un titre avec des soulignés
# et met le tout en minuscules. Enfin, transforme les & en «and».
# exemple :
# espsouligneminuscule "Toto est bte & méchant"
# donne toto_est_bete_and_mechant
#
espsouligneminuscule ()
{
    [ $# -eq 0 ] && return 2
    chaine="$*"

    resultat=$(echo "$chaine" | tr " A-Z"("\:/{}," "_a-z-----")
    print "$resultat" | sed -e 's@&@and@g' -e 's@(\(_\)\|\(-_\)\|(\(_-\_)\)@-@g' -e 's@-$$$@'
}

# fonction err ()
#
# usage:
# $ err [<messages d'erreur>+]
#
# cette fonction affiche l'ensemble des arguments
# passés en paramètres sur l'erreur standard.
err ()
{
    printf '%s\n' "$*" 1>&2
}

# fonction mkcd ()
#
# usage:
# $ mkcd [<répertoire à créer pour s'y rendre>]
#
# cette fonction crée un répertoire s'il n'existe pas
# et y va.
mkcd ()
{
    [ $# -eq 1 ] && mkdir -vp "$1"
    cd "$1"
}

# fonction urldecode ()
#
# usage:
# $ urldecode 'chaîne url codée à décoder'
#
# Cette fonction décode une chaîne écrite en codage url
# Trouvé ici : https://unix.stackexchange.com/questions/159253/decoding-url-encoding-percent-encoding#159373
urldecode ()
{
    sed -re 's@%([0-9A-F][0-9A-F])@\x\1@g' | xargs -0 printf '%b'
}

# fonction urlencode ()
#
# usage:
# $ urlencode 'chaîne à coder en urlcode'
#
# Cette fonction code une chaîne en codage url
urlencode ()
{
    local LANG=C
    local c
    while IFS= read -r c

```

```
do
    case $c in [a-zA-Z0-9._~]*) printf "%c"; continue ;; esac
    printf "%c" | od -An -tx1 | tr ' ' % | tr -d '\n'
done <<EOF
    $(fold -w1)
EOF
    echo
}

# fonction webdate ()
#
# usage:
# $ webdate <url dont on voudrait la date de dernière modif>
#
# fonction qui recherche la date de dernière modif de l'URL donnée
webdate ()
{
    [ $# -ne 1 ] && return 2
    url="$1"
    wget -S --spider "$url" 2>&1 | grep -Ei 'last-modified|date'
}
```

Alias utiles

alias.sh

```
alias webheaders='wget -S --spider'
```