

# Tartalomjegyzék

<b>Szkriptek</b> .....	<b>1</b>
------------------------	----------



# Szkriptek

Néhány, általunk használt Bash és egyéb szkript.

```
#!/bin/bash

# Created by LEX0, http://www.lexo.ch
# Version 1.0
#
# Ez a script konvertál minden alább megadott fájlt UTF-8 kódolásúra.
#

#checking Parameters
if [ ! -n "$1" ] ; then
    echo "You did not supply any directory at the command line."
    echo "Meg kell adni a konvertálandó fájlok elérési útját."
    echo ""
    echo "Példa: $0 /path/to/directory"
    echo ""
    echo "Fontos: ne futtasd ezt a szkriptet ugyanaból a könyvtárból, ahol
a konvertálandó fájlok vannak."
    exit
fi

# Ez a tömb minden fájlt tartalmaz (a binárisakat is!)
# Nem minden fájlt szabad konvertálni, ezért
# add meg, hogy mely fájltípusokat szeretnéd.

filestoconvert=(htm html php txt tpl asp css js)

# define colors
# default color
reset="\033[0;00m"

# Sikeres konvertálás (zöld)
success="\033[1;32m"

# Nem kell konvertálni, mert már UTF-8 (kék)
noconversion="\033[1;34m"

# nem szerepel a konvertálandó fájltípusok között (fehér)
fileskipped="\033[1;37m"

# nem konvertálható (piros)
fileconvertererror="\033[1;31m"

## function to convert all files in a directory recusrively
function convert {
#clear screen first
```

```

clear

dir=$1

# Get a recursive file list
files=(`find $dir -type f`);
fileerrors=""

#loop counter
i=0

find "$dir" -type f |while read inputfile
do
    if [ -f "$inputfile" ] ; then
        charset="$(file -bi "$inputfile"|awk -F "=" '{print $2}')"
        if [ "$charset" != "utf-8" ]; then
            #if file extension is in filestoconvert variable the
file will always be converted
            filename=$(basename "$inputfile")
            extension="${filename##*."}"
            # If the current file has not an extension that is
listed in the array $filestoconvert the current file is being skipped (no
conversion occurs)
            if in_array $extension "${filestoconvert[@]}" ; then
                # create a tempfile and remember all of the
current file permissions to be able to reapply those to the new converted
file after conversion
                tmp=$(mktemp)
                owner=`ls -l "$inputfile" | awk '{ print $3
}'`
                group=`ls -l "$inputfile" | awk '{ print $4
}'`
                octalpermission=$( stat --format=%a
"$inputfile" )
                echo -e "$success
$inputfile\t$charset\t->\tUTF-8 $reset"
                iconv -f "$charset" -t utf8 "$inputfile" -o
$tmp &>2
                RETVAL=$?
                if [ $RETVAL > 0 ] ; then
                    # There was an error converting the
file. Remember this and inform the user about the file not being converted
at the end of the conversion process.
                    fileerrors="$fileerrors\n$inputfile"
                fi
                mv "$tmp" "$inputfile"
                #re-apply previous file permissions as well
as user and group settings
                chown $owner:$group "$inputfile"
                chmod $octalpermission "$inputfile"
            else

```

```
                echo -e "$fileskipped
$inputfile\t$charset\t->\tSkipped because its extension (.$extension) is not
listed in the 'filestoconvert' array. $reset"
            fi
        else
            echo -e "$noconversion $inputfile\t$charset\t->\tNo
conversion needed (file is already UTF-8) $reset"
        fi
    fi
    (( ++i ))
done
echo -e "$success Done! $reset"
echo -e ""
echo -e ""
if [ ! $fileerrors == "" ]; then
    echo -e "The following files had errors (origin charset not recognized)
and need to be converted manually (e.g. by opening the file in an editor
(IDE) like Komodo or Netbeans:"
    echo -e $fileconvertererror$fileerrors$reset
fi
exit 0
} #end function convert()

# Check if a value exists in an array
# @param $1 mixed Needle
# @param $2 array Haystack
# @return Success (0) if value exists, Failure (1) otherwise} #end function
in_array()
# Usage: in_array "$needle" "${haystack[@]}"
in_array() {
    local needle=$1
    local hay=$2
    shift
    for hay; do
#     echo "Hay: $hay , Needle: $needle"
        [[ $hay == $needle ]] && return 0
    done
    return 1
} #end function in_array

#start conversion
convert $1
```

Használata: ./convert.sh /útvonal/a/konvertálandó/könyvtárhoz

From:

<https://wiki.mav-it.hu/> - **Mav-IT Wiki**

Permanent link:

<https://wiki.mav-it.hu/scripts/scripts>

Last update: **2019. March 12. 22:54**

