Scripting Stuff

My personal 'collection' of short scripts (mostly Linux/Unix shell) that I may need to refer in the future...

Fix Missing Leading Zero

Issue when creating CSV from spreadsheet program (e.g. Libreoffice Calc). The number string in column 3 of each line needs to be exactly 12 and the first line is a header line.

```
file=file.txt ; \
( head -n 1 $file ; \
tail -n +2 $file | \
while read line ; do \
line=${line// ,/,} ; \
test=$(echo $line|cut -d',' -f3) ; \
temp=$test ; \
while [ ${#test} -lt 12 ] ; do \
test="0$test" ; \
done ; \
line=${line//$temp/$test} ; \
echo "$line" ; \
done )
```

A script:

check_nrid.sh

```
#!/bin/bash

# check_nrid.sh
# - filters invalid nric in student list csv file for unidata
# - assume input is list.csv
# - run 'bash check_nric.sh list.csv >/dev/null' to check
# - if nothing comes out, then the file is good
# - run 'bash check_nrid.sh list.csv > list1.csv 2>/dev/null'
# - will get fixed data in list1.csv

check_nrid()
{
    local file loop line stid name nrid prog look prev
    file=$1
        [ ! -f $file ] && exit 0
    loop=0
    while read line ; do
        if [ $loop -gt 0 ] ; then
            stid=$(echo $line|cut -d',' -f1)
```

```
            name=$(echo $line|cut -d',' -f2)
            nrid=$(echo $line|cut -d',' -f3)
            prog=$(echo $line|cut -d',' -f4)
            look=0 prev=$nrid
            while [ ${#nrid} -ne 12 ] ; do
                nrid="0$nrid"
                look=1
            done
            # trim prog
            prog=${prog% }
            line="$stid,$name,$nrid,$prog,,,"
            if [ $look -ne 0 ] ; then
                echo "** [$prev] $line" >&2
            else
                echo "$line"
            fi
        fi
        ((loop++))
    done < $file
}
check_nrid $@
```

Read line-by-line

```
while IFS="" read -r line || [ -n "$line" ] ; do
echo "> $line"
done <<< $(cat $file)
```

Ensure integer value

get_integer.sh

```
#!/bin/bash
get_integer()
{
    # use: get_integer <def_int> <var_int>
    local idef test temp
    idef=$1
    test=$2
    if [ ! -z $test ] ; then
        temp=$(echo "$test" | grep -E -v "^[0-9]*$")
        [ -z "$temp" ] && idef=$test
    fi
    echo -n "$idef"
}
```

```
[ "$(basename $0 sh)" = "get_integer" ] && get_integer $@
```

Some bash 1-liners

Find broken symbolic links in current path:

```
find $(pwd) -type l -exec test ! -e {} \; -print
```

From:
https://azman.my1matrix.cc/ - **Azman @MY1**

Permanent link:
https://azman.my1matrix.cc/doku.php?id=notes:scripting

Last update: **2026/02/08 03:46**