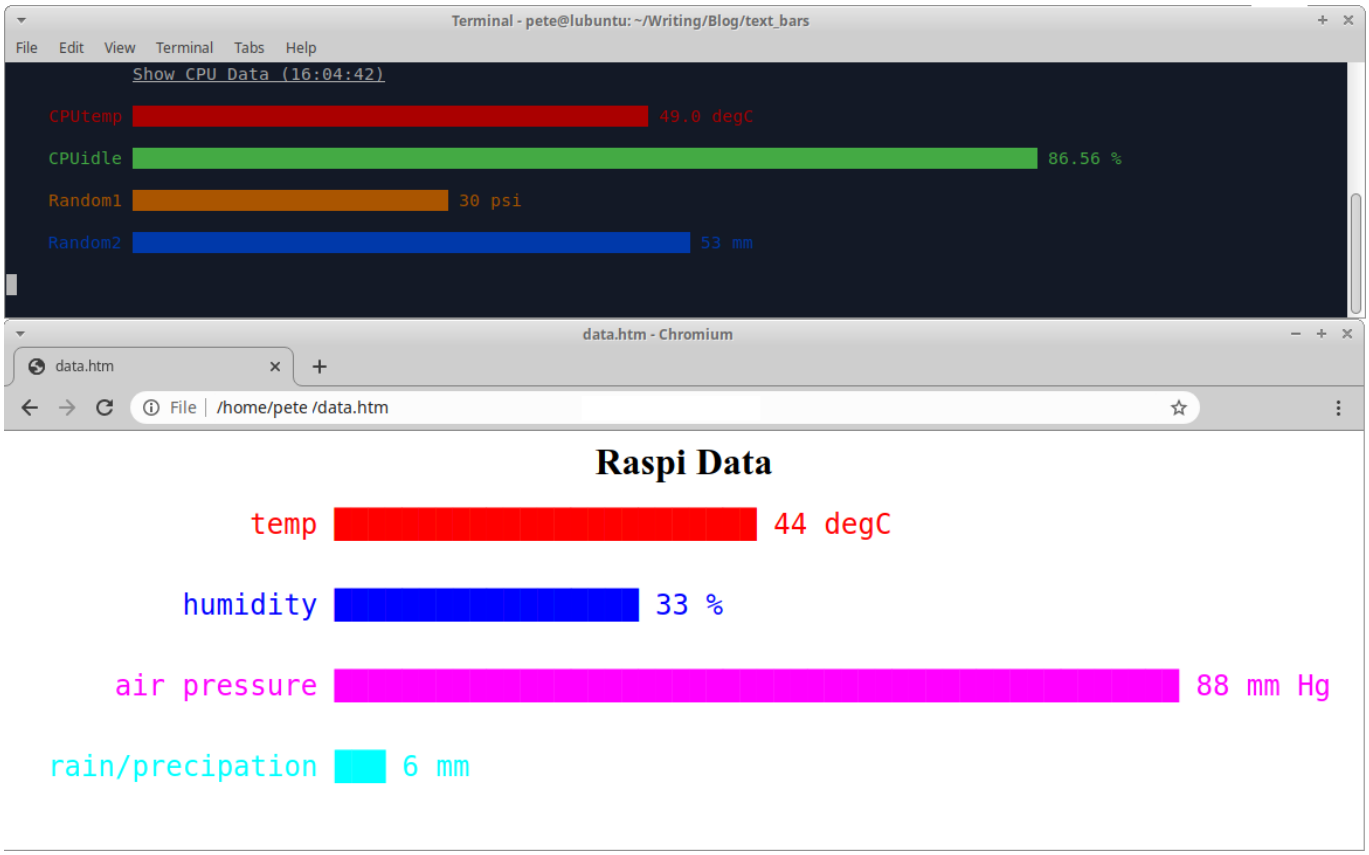
Table of Contents

HBars Bash Grafik Script

Hbars adalah script yang dapat menhasilkan grafik




```
#!/bin/bash
#
# hbars.sh - show some text bars
#   pass data as:  label1,value1,unit1,color1; label2,value2,unit2,colour2;
....
#
width=50
title=""
color=7
pretty=False
web=False
font=24

usage() {
  echo "usage: hbars [data] [option] "
  echo "  -h --help     print this usage and exit"
  echo "  -c --color    set color to all bars (default 7=white)"
  echo "                 (0-grey,1-
red,2=green,3=yellow,4=blue,5=magenta,6=cyan,7=white)"
  echo "  -p --pretty   add different colors to bars (-c overrides)"
  echo "  -t --title    top title"
  echo "  -w --width    max width of bar (default 50)"
```

```
  echo "  -W --Web      make output HTML formatted"
  echo "  -f --fontsize fontsize for web output (default 24)"
  echo ""
  echo " examples:"
  echo "   echo 'temp,33,C;pressure,14,psi' | ./hbars -t Weather -p -w 40 "
  echo "   ./hbars -t Weather -p -w 40  'temp,33;pressure,14' "
  echo "   cat data.txt | ./hbars -W -f 24 -t 'Raspi Data' > data.htm"
  echo ""

  exit 0
}
# Show help usage if no pipe data and no cmd line data
if [ -t 0 ]  && [ $# -eq 0 ] ; then
  usage
fi
# Check for command line options
while getopts "hpc:t:w:Wf:" arg; do
  case "$arg" in
    h) usage ;;
    c)  color=$OPTARG ;;
    p)  pretty=True; icolor=0 ;;
    t)  title=$OPTARG ;;
    w)  width=$OPTARG ;;
    W)  web=True;;
    f)  font=$OPTARG ;;
  esac
done
#-------------------------------------------------
# Setup formatting for text, Web and color
# ------------------------------------------------
if [[ ${color} != 7 && ${pretty} = True ]]; then
  pretty=False
fi
colidx=0

setcolors() {
if [ $web = True ]; then
  colors=(gray red green yellow blue magenta cyan white)
  titlebold="echo '<h1>'"
  titlereset="echo '</h1>'"
  #color_set='echo "<span style=font-size:$(font)px  >" '
  #color_set="printf '<span  style=\"font-
size:$(font)px;color:${colors[colidx]}\" >'"
  color_set="printf '<pre><span style=\"color:${colors[colidx]} ; font-
size:${font}px \" >'"
  color_rs="echo '</span></pre>'"
else
  colors=(0 1 2 3 4 5 6 7 )
  titlebold="tput bold; tput smul"
  titlereset="tput rmul; tput rmso"
  color_set="tput setaf ${colors[colidx]}"
```

```
  color_rs="tput rmso"
fi
}
setcolors
#-----------------------------------------------
# Get data, check if stdlin, file, if not assume string
#-----------------------------------------------
if [ -p /dev/stdin ]; then
        lastarg=$(< /dev/stdin)
else
    lastarg=$(echo "${@: -1}")
    if test -f "$lastarg"; then
      lastarg=$(<$lastarg)
    fi
fi
# Cleanup the input data
lastarg=$(echo $lastarg | sed 's/\n/;/g; s/  / /g; s/\t//g; s/;;/;/g')
IFS=';' read -r -a array <<< $lastarg

# ensure that there is some data
if [[ ${array} == 0 ]]; then
  echo "No data found"
  exit 0
fi
echo "input:$lastarg"
#exit 0
#-------------------------------------
# Get max value and max label length
#-------------------------------------
maxval=0
maxlbl=10
#echo "array:${array[@]}"
for element in "${array[@]}"
do
  IFS=',' read -r -a datapt <<< $element
  if (( $(echo "$maxval < ${datapt[1]}" |bc -l) )); then
    maxval=${datapt[1]}
  fi
  if (( $(echo "$maxlbl < ${#datapt[0]}" |bc -l) )); then
    maxlbl=${#datapt[0]}
  fi
done
#---------------------------------
# Print Title - use bold/underline
#---------------------------------
if [[ ! -z $title ]]; then
  printf "\n %${maxlbl}s " " "
  eval $titlebold
  printf "%s" "${title}" ; printf "\n\n"
  eval $titlereset
fi
```

```
#-----------------------------------
# Cycle thru data and build bar chart
#-----------------------------------
for element in "${array[@]}"
do
# check data values
  IFS=',' read -r -a datapt <<< $element
  # check for empty records
  if [ ${#datapt[0]} = 0 ]; then
    break
  fi
  label=${datapt[0]}
  if [[ ${label} != "-t*" ]]; then
      val=${datapt[1]}
      sval=$(bc <<< "$width * $val / $maxval")

      # add color, use 4th item if available
      if [[ ${#datapt[@]} > 3 && $pretty = False ]]; then
        icolor=${datapt[3]}
      fi
      if [[ $pretty = True ]] ; then
        let colidx++
        if [ $colidx -gt 7 ]; then
          let colidx=1
        fi
      elif [[ ${#datapt[@]} > 3 ]]; then
        colidx=${datapt[3]}
      else
        colidx=$color
      fi
      setcolors
          eval $color_set
          printf " %${maxlbl}s " "$label"
      bar="printf '%.0s' {1..$sval}"
      eval $bar;
      # add value and units if available
      units=""
      if [[ ${#datapt[@]} > 2 ]]; then
        units=${datapt[2]}
      fi
      printf " %d %s\n\n" $val "$units"
      eval $color_rs
  fi
done
```

Sumber : https://funprojects.blog/2021/05/14/bash-bar-charts-for-text-and-web-pages/

From:
https://www.pusathosting.com/kb/ - **PusatHosting Wiki**

Permanent link:
https://www.pusathosting.com/kb/linux/bash/hbars-bash-grafik-script

Last update: **2022/10/02 21:57**