Linux és shell programozás (scriptek)

```
which bash

#!/bin/bash

chmod +x script.sh
chmod 755 script.sh

bash script.sh
sh script.sh
./script.sh

# comment
: '
comment
more
lines
'

nano hello.sh
#!/bin/bash
echo "Hello World!"
echo -n "Hello World!"
echo -e "Hello\nWorld!"

chmod +x hello.sh
./hello.sh

echo $RANDOM
echo $((RANDOM % 100 + 1))

nano sum.sh
#!/bin/bash
echo -n "Number one: "
read A
echo -n "Number two: "
read B
echo "Sum: "`expr $A + $B`

chmod +x sum.sh
./sum.sh
```

```
nano num.sh
#!/bin/bash
read -p "Number one: " A
echo "Number one: "$A

chmod +x num.sh
./num.sh

nano commands.sh
#!/bin/bash
echo -n "Commands: "
read -a COMMANDS
echo ${COMMANDS[0]}
echo ${COMMANDS[1]}
echo ${COMMANDS[2]}

chmod +x commands.sh
./commands.sh
Commands: ls free df

nano password.sh
#!/bin/bash
read -p "Password: " -s PASSWORD
echo -e "\nPassword: "$PASSWORD

chmod +x password.sh
./password.sh

nano password_user.sh
#!/bin/bash
read -e -i "User" -p "Username: " USER
echo "Username: "$USER

chmod +x password_user.sh
./password_user.sh
```

```
nano equal.sh
#!/bin/bash
read -p "Number one: " A
read -p "Number two: " B
if [ $A -gt $B ]
then
  echo $A
elif [ $B -gt $A ]
then
  echo $B
else
  echo "Equal"
fi

chmod +x equal.sh
./equal.sh

nano operation.sh
#!/bin/bash
select i in "Addition" "Subtraction" "Multiplication" "Division" "Exit"
do
  echo "Menu: " $i
  if [ "$i" = "Exit" ]
  then
    break
  fi
done

chmod +x operation.sh
./operation.sh
```

```
nano calculator.sh
#!/bin/bash
echo "[A]ddition"
echo "[S]ubtraction"
echo "[M]ultiplication"
echo "[D]ivision"
echo -n "Choice: "
read MENU
case "$MENU" in
 "A" | "a" )
    echo "Addition"
    ;;
 "S" | "s" )
    echo "Subtraction"
    ;;
 "M" | "m" )
    echo "Multiplication"
    ;;
 "D" | "d" )
    echo "Division"
    ;;
esac

chmod +x calculator.sh
./calculator.sh

nano for.sh
#!/bin/bash
for (( i=0; i<11; i++ ))
do
  echo `expr $i "*" $i`
done

chmod +x for.sh
./for.sh
```

```
nano while.sh
#!/bin/bash
i=1
while [ $i -lt 11 ]
do
  echo `expr $i "*" $i`
  ((i+=1))
done

chmod +x while.sh
./while.sh

nano until.sh
#!/bin/bash
i=1
until [ $i -eq 11 ]
do
  echo `expr $i "*" $i`
  ((i+=1))
done

chmod +x until.sh
./until.sh

nano numbers.sh
#!/bin/bash
NUM=1
SUM=0
CNT=0
while [ $NUM -ne 0 ]
do
  echo -n `expr $CNT + 1`"Number: "
  read NUM
  ((CNT++))
  ((SUM+=NUM))
done
echo "Sum: "$SUM

chmod +x numbers.sh
./numbers.sh
```

```
cat > data.txt
CPU
RAM
GPU
HDD
SSD
^D

nano file_read.sh
#!/bin/bash
while read ROW
do
  echo $ROW
done < data.txt

chmod +x file_read.sh
./file_read.sh

cat > language.txt
C
C++
Java
^D

nano text_list.sh
#!/bin/bash
DIR=Directory/Subdirectory
cd $DIR
find $DIR -name '*.txt' | while read ROW
do
  echo $ROW
done

chmod +x text_list.sh
./text_list.sh

nano array.sh
#!/bin/bash
ARRAY=(8 "one" 6 "two" 5 "three")
echo ${ARRAY[3]}

chmod +x array.sh
./array.sh
```

```
nano array_2.sh
#!/bin/bash
ARRAY=(8 6 4 2 0)
# All items:
echo ${ARRAY[*]}
echo ${ARRAY[@]}
# All indexes:
echo ${!ARRAY[*]}
echo ${!ARRAY[@]}
# Number of items:
echo ${#ARRAY[*]}
echo ${#ARRAY[@]}
# Length of zeroth item
echo ${#ARRAY[0]}

chmod +x array_2.sh
./array_2.sh

nano array_3.sh
#!/bin/bash
ARRAY=("one" "two" "three" "four" "five")
for ITEM in ${ARRAY[*]}
do
  echo $ITEM
done

chmod +x array_3.sh
./array_3.sh

nano array_4.sh
#!/bin/bash
RATING=("first ZH" "second ZH" "big homework" "extra" "grade")
n=${#RATING[@]}
for (( i=0; i<n; i++))
do
   echo ${RATING[$i]}
done

chmod +x array_4.sh
./array_4.sh
```

```
nano array_5.sh
#!/bin/bash
ARRAY=(8 6 4)
ARRAY+=(2 0)
echo ${ARRAY[*]}
echo ${ARRAY[*]:1:3}
unset ARRAY[2]
echo ${!ARRAY[*]}
ARRAY=()
unset ARRAY

chmod +x array_5.sh
./array_5.sh

nano string.sh
#!/bin/bash
TEXT="Debian GNU/Linux"
SIZE=${#TEXT}
echo $SIZE
LENGHT=$(printf "%s" "$TEXT" | wc -c)
echo $LENGHT

chmod +x string.sh
./string.sh

nano string_2.sh
#!/bin/bash
TEXT="Debian GNU/Linux"
if [[ $TEXT == *"a"* ]]
then
  echo "There is"
else
  echo "There isn't"
fi

chmod +x string_2.sh
./string_2.sh
```

```
nano string_3.sh
#!/bin/bash
TEXT="Debian GNU/Linux"
for (( i=0; i<${#TEXT}; i++ ))
do
  echo "${TEXT:$i:1}"
done

chmod +x string_3.sh
./string_3.sh

nano hello_f.sh
#!/bin/bash
function list() {
  echo "Hello World!"
}
list

chmod +x hello_f.sh
./hello_f.sh

nano list_f.sh
#!/bin/bash
function list() {
  echo $1
}
list "This is a parameter!"

chmod +x list_f.sh
./list_f.sh

nano sqr_f.sh
#! /bin/bash
function sqr() {
  local RESULT=$(echo "$(($1 * $1))")
  echo "$RESULT"
}
echo -n "Number: "
read N
RESULT=$(sqr N)
echo $RESULT

chmod +x sqr_f.sh
./sqr_f.sh
```

```
nano array_f.sh
#! /bin/bash
function get_array() {
  ARRAY=(8 4 2 0)
  echo ${ARRAY[@]}
}
ARRAY=($(get_array))
echo "All items: "${ARRAY[@]}
echo "Size: "${#ARRAY[@]}

chmod +x array_f.sh
./array_f.sh
```