Bash

#!/bin/bash
#!/bin/sh

chmod +x *name.bash* == compiling, making executable ;
./*name.bash* == thus we get out put ;
bash *name.sh* == to compile and get the outputs;

if [-d ~/mydirectory] ; then echo “dir exist at home”
else echo “dir doesn’t exist “
fi

if [-f ~/mydirectory] ; then echo “file exist at home”
else echo “file doesn’t exist “
fi

```
foo=N

case $foo in
    Y|YES) date;
           ;;
    N|NO) cal
          ;;
esac
```

```
#! /bin/bash

set -x

#echo $#
#echo $0 $1

echo $myvar
date
```

```
# let foo=$1
# if [ $foo -ge 3 ];
# then
#     echo large
# else
#     echo small
# fi
```

```
# if [[ -w secret1.txt ]]
# then
#     echo can write
# else
#     echo cannot write
# fi

[ -w secret1.txt ] || echo "cannot write"
```

```
# let a=5+4
# echo $a # 9
# let "a = 5 + 4"
# echo $a # 9

# let a++
# echo $a # 10

# let "a = 4 * 5"
# echo $a # 20

let "a = $1 + 30"
echo $a # 30 + first command line argument
```

```
fname="$1"

cleanup() {
    echo "Removing $1 $fname"
    rm "$fname"
}

trap "cleanup howzat" EXIT

exec 5>"$fname"
echo "Row row row your boat" >&5
sleep 3 &
sleep 4
echo "Gently down the stream" >&5
exec 5>&-

cat >>"$fname" <<ANYNAME
Merrily etc.
Life is but etc.
ANYNAME

exec 5>>"$fname"
echo "Jingle all the way." >&5
exec 5>&-

while IFS= read -r var
do
  echo $var
done < "$fname"
```

```
globalv="today"

test_one_file() {
    if [[ -f $1 && -r $1 ]]; then
        echo $globalv $1 is a readable file
    else
        echo $globalv $1 is not a readable file
    fi
}

for fname in "$@"  # argv[1:]
do
    echo "$fname"
    test_one_file "$fname"
done
```

```
echo hello
date
/bin/date
```

```
#! /bin/bash

echo hello
date
```

```
tobuy=("apple" "orange" "kiwi")

#echo $tobuy
#echo ${tobuy}
#echo ${tobuy[*]}  # print whole array
#echo $tobuy[*]

# echo ${#tobuy[@]}

tobuy[4]="carrot"

# echo ${tobuy[*]}
# echo ${#tobuy[@]}

for (( i=0; i<${#tobuy[@]}; i++ ))
do
    echo $i ${tobuy[$i]}
done
echo at 4 _${tobuy[4]}_
```

```
read a b
echo a $a b $b

read -p "Enter c, d: " c d
echo $c $d
```

```
read -p "Password: " -s secret
echo $secret > secret1.txt

read -p "Password: " -e -s secret
echo $secret > secret2.txt
```

```
read a b
echo a $a b $b

read -p "Enter c, d: " c d
echo $c $d

read -p "Password: " -s secret
echo $secret > secret1.txt

read -p "Password: " -e -s secret
echo $secret > secret2.txt
```