

Shell script

קבצי shell script הם למעשה קבצי אצווה.
ישנם מספר סוגי shell :

sh
csh, bash
tcsh
ksh (IBM)

Unix נותן מספר סוגי shell-ים בעוד pc נותן רק אחד.
אם המשתמש לא מבקש להשתמש ב-shell מסויים המערכת בוחרת את ברירת המחדל שלה, במקרה שלנו ברירת המחדל היא tcsh.

בגלל שאפשר להשתמש בכמה סוגי shell-ים כדי מאוד (צריך) להגדיר באיזה shell משתמשים.
כדי להגדיר באיזה shell משתמשים קודם כותבים #! ולאחר מכן את סוג ה-shell שעובדים אתו.

שפות ה-shell מצפות לקבל מחרוזות (string) ולכן אין צורך במירכאות (" "), ברירת המחדל היא מחרוזות. אם רוצים לכתוב מחרוזת עם רווחים אז חייבים לשים מירכאות: "abc def".
משתנים ב-shell יכולים להיות עם סימן \$ או בלי סימן \$. היכן שברור למערכת שמדובר במשתנה כותבים ללא \$ למשל במצב של השמה a=I (a חייב להיות משתנה).

פרמטרים של השוואה :

less then = -lt
greater then = -gt
equal = -eq
not equal = -ne
less then or equal = -le
greater or equal = -ge

כעת נעבור לקוד.

הפקודה echo פשוט מדפיסה למסך.
הפקודה for מבצעת לולאה עד "שנגמרים" לה האיברים ברשימה.
לולאת for נסגרת ע"י הפקודה done.

תכניות לדוגמא :

```
myghaz 1 >cat shel
# ! /bin/ sh
```

```
for i in a b c; do
    echo $i
done
```

```
myghaz 2 >shc1
a
b
c
```

```
myghaz 3 >cat shc2
# ! /bin/ sh
```

```
for i in `ls klum`; do
    echo $i
done
```

```
myghaz 4 >shc2
klum1
klum2
klum3
```

```
myghaz 5 >cat shc3
# ! /bin/ sh
```

```
for i in `ls klum`; do
    echo nothing$i
done
```

```
myghaz 6 >shc3
nothingklum1
nothingklum2
nothingklum3
```

```
myghaz 7 >cat shc4
# ! /bin/ sh
```

```
a=nothing
for i in `ls klum`; do
    echo $a$i
done
```

```
myghaz 8 >shc4
nothingklum1
nothingklum2
nothingklum3
```

התנאי if נסגרת ע"י הפקודה fi
המשך תכניות לדוגמא :

```
myghaz 9 >cat shc5
# ! /bin/ sh
```

```
for i in `ls klum`; do
    if [ $i != klum1 ]; then
        echo $i
    fi
done
```

```
myghaz 10 >shc5
klum2
klum3
```

```
myghaz 9 >cat shc6
```

```
# ! /bin/ sh
```

```
for i in `ls klum`; do
    if [ -r klum/$i ]; then
        echo $i
    fi
done
```

```
myghaz 12 >shc6
```

```
klum2
```

```
myghaz 13 >cat shc7
```

```
# ! /bin/ sh
```

```
for i in `ls klum`; do
    if [ ! -r klum/$i ]; then
        echo $i
    fi
done
```

```
myghaz 14 >shc7
```

```
klum1
```

```
klum3
```

```
myghaz 15 >cat shc8
```

```
# ! /bin/ sh
```

```
for i in `ls`; do
    if [ -d $i ]; then
        echo $i
    fi
done
```

```
myghaz 16 >shc8
```

```
klum
```

```
klum6
```

התנאי if נסגרת ע"י הפקודה fi אבל מה נעשה אם אנו רוצים לבצע משהו מיוחד אם התנאי לא מתקיים ורק אח"כ להמשיך?
משתמשים בפקודה המשלימה .else.
המשך תכניות לדוגמא:

```
myghaz 17 >cat shc9
```

```
# ! /bin/ sh
```

```
for i in `ls klum`; do
    if [ $i = klum1 -o $i = klum2 ]; then
        echo $i
    else
        echo abc$i
    fi
done
```

```
myghaz 18 >shc9
klum1
klum2
abcklum3
```

הפקודה case היא סוג של ריבוי תנאים. כאשר רוצים לבצע הרבה דברים שונים על כל קלט לא יעיל כל כך להשתמש במספר רב של if-ים, לצורך זה קיימת הפקודה case. הפקודה case נסגרת ע"י הפקודה esac.

המשך תכניות לדוגמא :

```
myghaz 19 >cat shc10
# ! /bin/ sh
```

```
for i in `ls klum`; do
    case $i in
        klum1)
            echo aa$i
            ;;
        klum2)
            echo bb$i
            ;;
        *)
            echo cc$i
            ;;
    esac
done
```

```
myghaz 20> shc10
aaklum1
bbklun2
ccklum3
```

הפקודה while מבצעת לולאה כל עוד התנאי שבסוגריים מתקיים (תנאי שהתשובה עליו היא כן). הפקודה while נסגרת ע"י הפקודה done.

הואיל ו-shell משתמשת במחרוזות אז אין ערך לוגי בביטוי (ציור) $1+1$ כי הם רק רצף של תווים ללא משמעות מספרית. כאשר רוצים לעשות פעולה מתמטית אז צריך להשתמש בפונקציה עזר שהיא expr.

המשך תכניות לדוגמא :

```
myghaz 21 >cat shc11
#! /bin/ sh
```

```
i=0
while [ $i -lt 5 ]; do
    echo $i
    i=`expr $i + 1`
done
```

```
myghaz 22 >shc11
0
1
2
3
4
```

פונקציות ב-shell :

פונקציות ב-shell אינן מחזירות ערך, הן רק מבצעות את מה שכתוב להן לבצע.

המשך תכניות לדוגמא :

```
myghaz 23 >cat shc12
#! /bin/ sh
```

```
func ( )
(
    echo abc
)
```

```
func
myghaz 24 >shc12
abc
```

```
myghaz 25 >cat shc13
#! /bin/ sh
```

```
i=0
while [ $i -lt 5 ]; do
    i=`expr $i + 1`
    if [ $i -eq 3 ]; then
        continue
    else
        echo $i
    fi
done
```

myghaz 26 >shc13

1
2
4
5

myghaz 27 >cat shc14

! /bin/ sh

i=0

while [\$i -lt 5]; do

i=`expr \$i + 1`

if [\$i -eq 3]; then

break

else

echo \$i

fi

done

echo abc

myghaz 28 >shc14

1
2
abc

myghaz 29 >cat shc15

! /bin/ sh

i=0

while [\$i -lt 5]; do

i=`expr \$i + 1`

if [\$i -eq 3]; then

exit

else

echo \$i

fi

done

echo abc

myghaz 30 >shc15

1
2

משתנים מיוחדים ב- :shell script

מספר הארגומנטים בשורת פקודה.	\$#
שם התכנית.	\$0
הארגומנטים בשורת הפקודה.	\$1-\$9
כל הארגומנטים (מופרד במרכאות).	\$@
כל הארגומנטים.	\$*
מספר ה-ID של ה-script שרץ.	\$\$