

DATE : 08.01.2025
DT/NT : DT
LESSON : LINUX
SUBJECT: SHELL SCRIPTING LOOPS
SESSION : 10
BATCH : B 303

AWS-DEVOPS



TECHPRO
EDUCATION



techproeducation.com

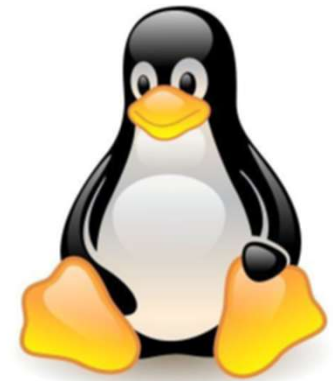


+1 (585) 304 29 59





Loops



While loops

while döngüsü, belirli bir koşul doğru olduğu sürece komutları tekrarlar



```
while [[ <some test> ]]  
do  
  <commands>  
done
```

```
#!/bin/bash  
  
number=1  
  
while [[ $number -le 10 ]]  
do  
  echo $number  
  ((number++))  
done  
echo "Now, number is $number"
```

Output:

```
$_while-loops.sh  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
Now, number is 11
```

Until loops

until döngüsü, belirli bir koşul doğru olana kadar bir dizi komutu tekrarlar.



```
until [[ <some test> ]]  
do  
  <commands>  
done
```

```
#!/bin/bash  
  
number=1  
  
until [[ $number -ge 10 ]]  
do  
  echo $number  
  ((number++))  
done  
echo "Now, number is $number"
```

Output:

```
$/until.sh  
1  
2  
3  
4  
5  
6  
7  
8  
9  
Now, number is 10
```

For loops



for döngüsü, belirli bir aralıkta veya liste üzerinde dolaşarak komutları tekrarlar.

```
for item in [list]
do
commands
done
```

```
#!/bin/sh

echo "Numbers:"

for number in 0 1 2 3 4 5 6 7 8 9
do
    echo $number
done
```

Output:

```
$/for-loop.sh
Numbers:
0
1
2
3
4
5
6
7
8
9
```

Continue and Break Statement

continue ve break komutları, döngülerin ve şartlı ifadelerin kontrol akışını yönlendirmek için kullanılır.



Infinite loop

```
#!/bin/bash

number=1

until [[ $number -lt 1 ]]
do
    echo $number
    ((number++))
done
echo "Now, number is $number"
```

Continue and Break Statement



break komutu, döngüyü veya şartlı ifadeyi sonlandırır. Yani, belirli bir koşul gerçekleştiğinde döngüyü tamamen sonlandırır ve döngü dışındaki bir sonraki komut veya ifadeye geçer.

Break Statement

```
#!/bin/bash

number=1

until [[ $number -lt 1 ]]
do
    echo $number
    ((number++))
    if [[ $number -eq 10 ]]
    then
        break
    fi
done
```

Output:

```
./infinite-loop.sh
1
2
3
4
5
6
7
8
9
```

Continue and Break Statement



continue komutu, bir döngü içinde bulunduğunuzda, döngünün geri kalan kısmını atlayarak bir sonraki iterasyona geçmenizi sağlar.

Continue Statement

```
#!/bin/bash
number=1
until [[ $number -lt 1 ]]
do
    ((number++))
    tens=$(( $number % 10 ))
    if [[ $tens -eq 0 ]]
    then
        continue
    fi
    echo $number
    if [[ $number -gt 14 ]]
    then
        break
    fi
done
```

Output:

```
$/continue.sh
2
3
4
5
6
7
8
9
11
12
13
14
15
```


Exercise 1

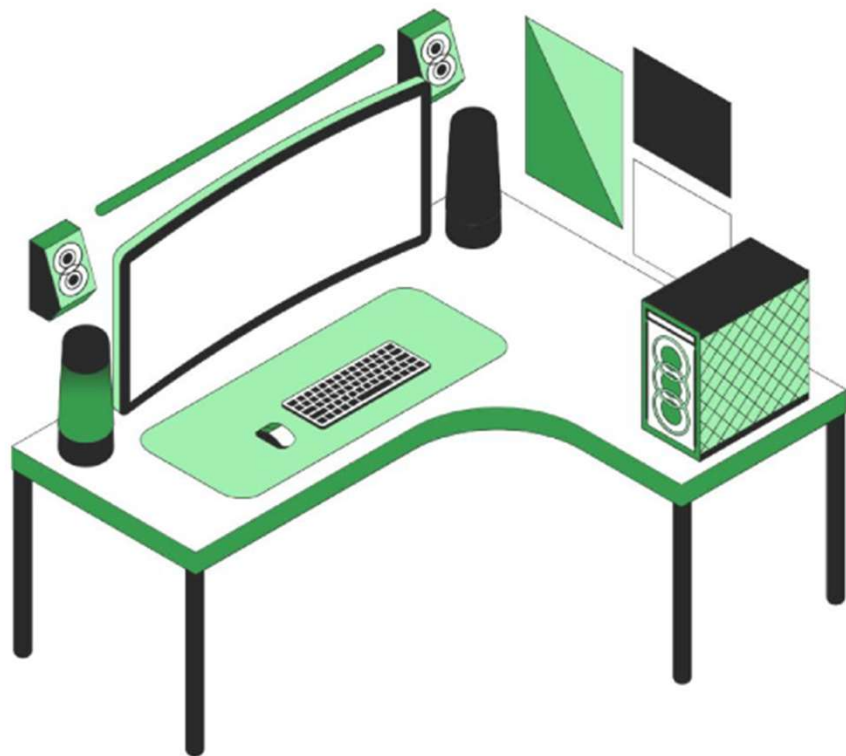


1. 1'den 100'e kadar olan sayıların toplamını hesaplayın.
2. Sonucu yazdırın.

Exercise 2



1. Kullanıcıdan tek bir satıra birden fazla ad girmesini isteyin
2. Her isim için ayrı satırlara “Merhaba” mesajını yazdırın.



Do you have any questions?

Send it to us! We hope you learned something new.