

# Kabuk Programlama (Bash) - 5

Erkan Esmer

Temmuz, 2013

Bu uygulama her yazı dizimiz sonucunda yeni özellik bir kazanıyor veya yaptığı işlemi daha kısayoldan yapıyor. Şimdiye kadarki ilerleme şeklimiz ve durumumuz için önceki sayılarımıza bakabilirsiniz.

Giriş mahiyetinde bakacak olursak bir klasör ve dosya oluşturup yaptığımız matematiksel işlemi bu dosya ve klasöre kaydeden bir uygulama yaptık şimdiye kadar. İf kontrolleri kullandık. GoTo deyimini kullanıp ardından bunu döngüye çevirdik. Fonksiyon kullanıp kaldırdık.

Bu sayımızda da uygulamamıza, temel matematik işlemlerini seçimimize bağlı olarak yaptıracağız. Uygulama sadece çıkartma işlemi yapıyordu. Şimdi uygulama bizden yine sayıları isteyecek ve ardından yapacağımız işlemi bize seçtirip sonucunu verecek. Bunu Case-Esac kalıbını kullanarak yapacağız.

Ayrıca dergi okurlarımızdan forum isimleri jayw ve monthy\_python arkadaşlarımızın ricası üzerine kodları text biçiminde yazımızda kullanacağız. Umarız sizin için kolaylık olur.

Dergimize göstermiş olduğunuz ilgiden dolayı çok teşekkür ederiz.

### **Başlıyoruz;**

Case-Esac Kalıbı: Bir işlem için birden fazla seçeneğin olduğu durumlarda kullanılır. Bir menü tasarımı yapıp kullanıcıya bu menüden işlem seçimi yaptırılabilir.

Şimdi uygulamamızın son hâlini hatırlayalım.

```
#!/bin/bash

echo "Lufen birinci klasorun ismini giriniz."
read klasor1
echo "Lutfen ikinci klasorun ismini giriniz."
read klasor2

mkdir $klasor1
mkdir $klasor2
echo $klasor1 " ve " $klasor2 " isminde klasorler olusturuldu. "

echo "Lutfen olusturulacak dosyanin ismini giriniz."
read dosya1
touch $klasor1/$dosya1
echo $dosya1 " isimli dosya olusturuldu."

echo "Lutfen birinci sayiyi giriniz."
read sayi1
echo "Lutfen ikinci sayiyi giriniz."
read sayi2

while [ $sayi1 -lt $sayi2 ];
do
echo "Birinci sayi, ikinci sayidan kucuk. Lutfen birinci sayiyi tekrar giriniz."
read sayi1
echo "Lutfen ikinci sayiyi giriniz"
read sayi2
done

typeset -i sonuc
sonuc=$((sayi1-$sayi2))
echo $sonuc > $klasor1/$dosya1
echo "Islemin sonucu:" $sonuc " Dosya yolu:" $klasor1/$dosya1
```

Şekil 1:

Burada while döngüsünü girilen sayıların büyük/küçük karşılaştırmasını yapıyoruz. Bu kontrolü temel matematik işlemlerinde sadece çıkartmada kullanacağız. Bu kontrolü yapıp ardından typeset -i sonuc ile başlayan işlemimizle çıkartma

işlemini yapıyoruz.

Şimdi bu araya Case-Esac kalıbıyla bir ekleme yapıp istediğimiz işlemi yapıp sonucunu alacağız.

While döngüsünden önceki birinci ve ikinci sayıyı aldığımız kodun arkasına işlemler için seçim menüsünü oluşturalım.

```
echo "Yapılacak işlemi seçiniz."
echo "1-Topla"
echo "2-Çıkart"
echo "3-Böl"
echo "4-Çarp"
read secenek
case $secenek in
```

Şekil 2:

Burada ekrana "Yapılacak işlemi seçiniz" mesajıyla beraber 1, 2, 3, 4 şeklinde maddelendirerek "secenek" değişkenimize atayacağımız işlemin seçimini yansıtıyoruz.

Daha sonra ekrana yansıyan 4 maddeden herhangi birinin seçilmesi ile yapılacak işlemleri belirtiyoruz.

```
1)typeset -i sonuc
sonuc=$((sayi1+$sayi2)
echo $sonuc > $klasori/$dosya1
echo "İşlemin sonucu:" $sonuc " Dosya yolu:" $klasori/$dosya1;;
2)while [ $sayi1 -lt $sayi2 ];
do
echo "Birinci sayı ikinci sayıdan küçük. Lütfen birinci sayıyı tekrar giriniz."
read sayi1
echo "Lütfen ikinci sayıyı giriniz."
read sayi2
done
typeset -i sonuc
sonuc=$((sayi1-$sayi2)
echo $sonuc > $klasori/$dosya1
echo "İşlemin sonucu:" $sonuc " Dosya yolu:" $klasori/$dosya1;;
3)typeset -i sonuc
sonuc=$((sayi1/$sayi2)
echo $sonuc > $klasori/$dosya1
echo "İşlemin sonucu:" $sonuc " Dosya yolu:" $klasori/$dosya1;;
4)typeset -i sonuc
sonuc=$((sayi1*$sayi2)
echo $sonuc > $klasori/$dosya1
echo "İşlemin sonucu:" $sonuc " Dosya yolu:" $klasori/$dosya1;;
esac
```

Şekil 3:

Gördüğümüz gibi 1), 2), 3), 4) şeklinde sonları ;; ile biten kodlar girdik.

- 1) alanında typeset -i ile başlayan kodumuzla direkt girilen 2 sayıyı toplayıp sonucu klasör ve dosyaya yazıyoruz. Burada toplama işlemi yaptığımız için sayılara büyük/küçük karşılaştırması yapmadık.
- 2) alanında ise çıkartma işlemi yapıyoruz. Eksi sonuç vermemesi için while [ \$sayi1 -lt \$sayi2 ] do döngümüz ile girilen sayılara büyük/küçük karşılaştırması yapıyoruz.

**Not:**-Siz de diğer işlemler için buna benzer kontroller yapabilirsiniz.- 52. sayımızdaki yazıda -lt, gt ve diğer aritmetik karşılaştırmaları belirtmiştik.

3) ve 4) alanlarında da herhangi bir kontrol yapmadan işlemi yapıp kaydı gerçekleştiriyoruz.

Şimdi uygulamayı çalıştırıp kullanalım. Sarı renkli satırlarda göreceğiniz gibi Case-Esac kalıbı sayıları aldıktan sonra bize seçim menüsünü çıkartıyor. Biz de bu ekranda 1 numaralı seçimi yapıp toplama işlemini yaptırıyoruz.

```
Lutfen birinci klasorun ismini giriniz.
klasor1
Lutfen ikinci klasorun ismini giriniz.
klasor2
klasor1 ve klasor2 isiminde klasorler olusturuldu.
Lutfen olusturulacak dosyanin ismini giriniz.
dosya1
dosya1 isimli dosya olusturuldu.
Lutfen birinci sayiyi giriniz.
10
Lutfen ikinci sayiyi giriniz.
19
Yapilacak islemi seciniz.
1-Topla
2-Cikart
3-Bol
4-Carp
1
Islemin sonucu: 29 Dosya yolu: klasor1/dosya1
```

Şekil 4:

Bir de çıkartma işlemine koyduğumuz kontrolün çalışıp çalışmadığına bakalım.

```
Yapilacak islemi seciniz.
1-Topla
2-Cikart
3-Bol
4-Carp
2
Birinci sayi ikinci sayidan kucuk. Lutfen birinci sayiyi tekrar giriniz.
```

Şekil 5:

Uygulamayı tekrar çalıştırıp aynı sayılarla bu çıkartma işlemi seçtiğimizde karşılaştırma yapıp eksi sonuç vermemesi için uygulama, sayıyı bizden tekrar istiyor.

Böylece Case-Esac kalıbını da kullanmış olduk. Daha çeşitli ve karmaşık kullanımlar için sizler de denemeler yapabilirsiniz. Daha önce de söylediğimiz gibi bizimle de paylaşabilirsiniz. Uygulamamızın kodları aşağıdadır.

Görüşmek üzere,

```
1 #!/bin/bash
2 echo "Lutfen birinci klasorun ismini giriniz."
3 read klasor1
4 echo "Lutfen ikinci klasorun ismini giriniz."
5 read klasor2
6
7 mkdir $klasor1
8 mkdir $klasor2
9 echo "$klasor1 " ve " $klasor2 " isiminde klasorler oluşturuldu. "
10
11 echo "Lutfen oluşturulacak dosyanın ismini giriniz."
12 read dosya1
13 touch $klasor1/$dosya1
14 echo "$dosya1 " isimli dosya oluşturuldu."
15
16 echo "Lutfen birinci sayıyı giriniz."
17 read sayi1
18 echo "Lutfen ikinci sayıyı giriniz."
19 read sayi2
20
21 echo "Yapılacak işlemi seçiniz."
22 echo "1—Topla"
23 echo "2—Çıkart"
24 echo "3—Bol"
25 echo "4—Carp"
26 read secenek
27 case $secenek in
28
29 1)typeset -i sonuc
30 sonuc=$((sayi1+$sayi2)
31 echo "$sonuc > $klasor1/$dosya1
32 echo "İslemin sonucu:" "$sonuc " Dosya yolu:" "$klasor1/$dosya1;;
33 2)while [ $sayi1 -lt $sayi2 ];
34 do
35 echo "Birinci sayı ikinci sayıdan küçük. Lutfen birinci sayıyı tekrar giriniz."
36 read sayi1
37 echo "Lutfen ikinci sayıyı giriniz."
38 read sayi2
39 done
40 typeset -i sonuc
41 sonuc=$((sayi1-$sayi2)
42 echo "$sonuc > $klasor1/$dosya1
43 echo "İslemin sonucu:" "$sonuc " Dosya yolu:" "$klasor1/$dosya1;;
44
45 3)typeset -i sonuc
46 sonuc=$((sayi1/$sayi2)
47 echo "$sonuc > $klasor1/$dosya1
48 echo "İslemin sonucu:" "$sonuc " Dosya yolu:" "$klasor1/$dosya1;;
49 4)typeset -i sonuc
50 sonuc=$((sayi1*$sayi2)
51 echo "$sonuc > $klasor1/$dosya1
52 echo "İslemin sonucu:" "$sonuc " Dosya yolu:" "$klasor1/$dosya1;;
53 esac
```