

4.Ders : C break ve continue

Break ifadesi döngüyü sonlandırırken, **continue** ifadesi döngünün bir sonraki yinelemesini zorlar. Bu derste, **ara vermeyi** ve örnekler yardımıyla **devam etmeyi** öğreneceksiniz.

Bazen, döngü içindeki bazı ifadeleri atlamak veya test ifadesini kontrol etmeden hemen döngü sonlandırılmak istenebilir.

Bu gibi durumlarda, **Break** ve **continue** ifadeleri kullanılır.

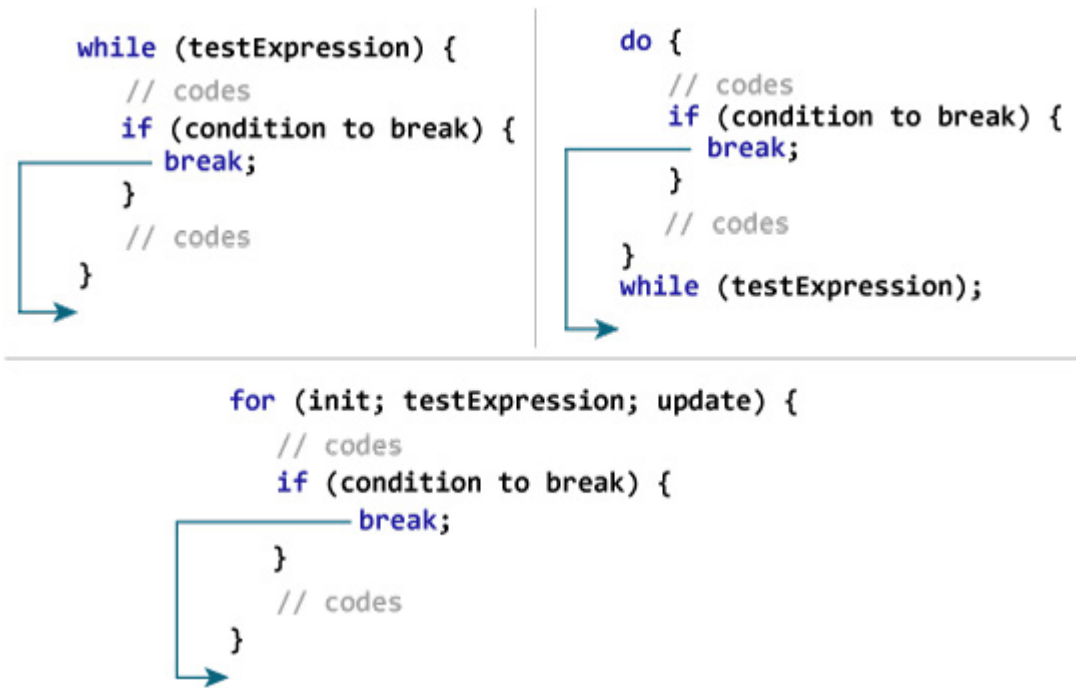
break ifadesi

Break ifadesi, rastlandığında derhal döngüyü sonlandırır.

```
break;
```

Break deyimi neredeyse her zaman döngü içindeki [if ... else](#) deyimi ile birlikte kullanılır.

Break ifadesi nasıl çalışır?



Örnek 1: break ifadesi

```
# include <stdio.h>
int main()
{
    int i;
    double number, sum = 0.0;

    for(i=1; i <= 10; ++i)
    {
        printf("bir sayı giriniz n%d: ",i);
        scanf("%lf",&number);

        // Kullanıcı negatif sayı girerse, döngü sonlandırılır
        if(number < 0.0)
        {
            break;
        }

        sum += number;
    }

    printf("Toplam= %.2lf",sum);
}
```

```
    return 0;  
}
```

Çıktısı :

```
bir sayı giriniz n1: 2.4  
bir sayı giriniz n2: 4.5  
bir sayı giriniz n3: 3.4  
bir sayı giriniz n4: -3  
Toplam = 10.30
```

continue İfadesi

continue İfadesi, Döngüden sonraki ifadeleri atlar.

```
continue;
```

While deyimi neredeyse her zaman [if ... else](#) deyimi ile birlikte kullanılır.

continue İfadesi nasıl çalışır?

<pre> while (testExpression) { // codes if (testExpression) { continue; } // codes } </pre>	<pre> do { // codes if (testExpression) { continue; } // codes } while (testExpression); </pre>
---	---

```

for (init; testExpression; update) {
    // codes
    if (testExpression) {
        continue;
    }
    // codes
}

```

Örnek 2 : continue İfadesi

```

#include <stdio.h>
int main()
{
    int i;
    double number, sum = 0.0;

    for(i=1; i <= 10; ++i)
    {
        printf("bir sayı giriniz n%d: ",i);
        scanf("%lf",&number);

        if(number < 0.0)
        {
            continue;
        }

        sum += number;
    }

    printf("Toplam = %.2lf",sum);
    return 0;
}

```

Çıktısı :

```
bir sayı giriniz n1: 1.1
bir sayı giriniz n2: 2.2
bir sayı giriniz n3: 5.5
bir sayı giriniz n4: 4.4
bir sayı giriniz n5: -3.4
bir sayı giriniz n6: -45.5
bir sayı giriniz n7: 34.5
bir sayı giriniz n8: -4.2
bir sayı giriniz n9: -1000
bir sayı giriniz n10: 12
Toplam = 59.70
```

Programda, kullanıcı pozitif sayı girdiğinde toplam, **toplam += sayı** kullanılarak hesaplanır.

Kullanıcı negatif sayı girdiğinde, devam ifadesi çalıştırılır ve negatif sayıyı hesaplamadan **atlar**.