

# PHP Döngüler ve Kullanımı

## Döngü Nedir ?

Programımızı yazarken tekrarlanmasını istediğimiz işlemleri elle yazmak yerine otomatik yazması için kullandığımız kod sistemine döngü adı verilir. Toplu güncelleme, bir tablodan toplu veri çekme gibi işlemlerde döngüleri kullanırız.

Döngüler ikiye ayrılır;

- **Sınırlı Döngü** : Belirli bir sayıya bağlı olarak dönecek olan döngülere denir
- **Sınırsız Döngü** : Bir koşulun sonucuna (TRUE,FALSE) bağlı olarak dönen döngülere sınırsız döngü denir. Bu durum döngü içerisindeki kodlara bağlıdır.

## While Döngüsü

While döngüsü koşulun sonucu TRUE olduğu sürece çalışır. Kullanımına dikkat edilmelidir çünkü çok kolay sonsuz döngüye girebiliriz. While döngüsü koşulun sonucu FALSE olduğu zaman sonlanır.

```
[crayon-5c694e22814cd393322614/]
```

## Do-While Döngüsü

do...while döngüsünde koşulun sonucu FALSE olsa bile döngümüz bir kere çalışır daha sonra koşulumuzun sonucu kontrol edilir ve koşulumuz sonucu TRUE ise döngümüz koşulun sonucu FALSE olana kadar çalışır.

```
[crayon-5c694e22814dc939628317/]
```

## For Döngüsü

Belirtmiş olduğumuz sayı kadar istediğimiz komutları çalıştıran döngüdür. 3 Parametreden oluşur. Bunlar;

1. **Başlangıç** : Döngünün başlayacağı değeri belirtir.

2. **Koşul** : Döngünün çalışması için gerekli koşuldur. Koşul doğru olduğu sürece döngü çalışır.
3. **Döngü İşlemi** : Koşul doğru olduğu sürece döngünün nasıl ilerleyeceğini belirtir.

[crayon-5c694e22814e4962830097/]

## **Foreach Döngüsü**

Diziler içerisindeki elemanlar üzerinde tekrarlayan işlemler yapmak istediğimizde foreach döngüsü kullanırız. Foreach döngüsü dizinin eleman sayısı kadar çalışır ve her elemanı parametre olarak belirttiğimiz değişkene atar.

[crayon-5c694e22814ec796535097/]

## **Break ve Continue**

**Break** : Yazdığımız program içerisindeki döngülerin bazı koşullarda durdurulması gerekir. Böyle koşullarda break komutu ile döngüyü istediğimiz zaman durdurabiliriz.

**Continue** : Döngümüzün bazı koşullarda çalışmasının durdurulması ve diğer değerden çalışmaya devam etmesi istenilebilir. Continue komutu ile böyle durumlarda döngümüzü o koşulda durdurup diğer koşuldan devam etmesini sağlıyoruz. Bir nevi atlatıyoruz döngüyü o değer için.

*Break ve Continue komutları ile sayısal değerler kullanılarak kaç üst seviyedeki döngüye müdahale edebilecekleri belirtilebilir.*

Artık döngülerle ilgili örneklere başlayalım.

[crayon-5c694e22814fa742260157/]

Yukarıdaki kodda, 1 sayısından 100 sayısına kadar olan tek sayıları aralarına – koyarak yazdırdık. Döngü içerisinde sayımızın 2 sayısına bölünüp bölünmediğini kontrol ettirdik. Eğer 2 sayısına bölünüyorsa sayıyı atlamasını söyledik.

[crayon-5c694e2281503200273054/]

Bu kodumuzda normalde while ile yaptığımız işlem sonsuz döngüdür. Çünkü deger sürekli 1 artacak ve bu 0 sayısından

büyük olduğu sürece devam edecek ki her zaman sıfırdan büyük olacağı için sonsuza kadar gidecektir. Biz kodumuzda deger değişkenini kontrol ettik ve eğer 100 sayısına eşitse 100 sayısını ekrana yazdır ve döngüden çık komutu verdik. Break ile döngüyü sonlandırdık.

[crayon-5c694e228150b944508622/]

Bu örnekte de switch yapısı ve while döngüsü kullandık. i değişkenini 0 dan başlattık ve sürekli arttırdık. switch ile i değişkenimizi kontrol ettik. eğer i değişkenimizin değeri 5 sayısına eşit ise break 1 ile 1 üst yapıdan çıkmasını söyledik yani switch yapısından. eğer sayımız 10 sayısına eşitse de break 2 komutu ile iki üstteki while döngüsünden çıkmasını söyledik doğal olarak while döngüsü sonlanınca switch yapısı da sonlandı.