

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-5c6951690fa68478926198/]
```

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-5c6951690fa77397598781/]
```

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-5c6951690fa7f815665699/]

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-5c6951690fa87486746012/]

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-5c6951690fa90856727890/]

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-5c6951690fa98001401641/]

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-5c6951690faa0428351961/]
```

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.

PHP Tanımı ve Kullanım Alanları

PHP nedir?

PHP adını “Personal Home Page tool” kelimelerinin baş harflerinden almıştır. PHP, özellikle dinamik web sayfaları yaratmak amacıyla geliştirilmiş, sunucu tarafında çalışan (ASP tarzı) bir programlama dilidir. Bir ziyaretçi sayfanıza girdiğinde, sunucu PHP komutlarını çalıştırır ve o anda bir HTML çıktısı üretir. Ziyaretçinin bütün görebileceği bu çıktı olur. PHP kodunu göremez. Dinamik sayfa kavramı da buradan gelmektedir. Ziyaretçinin kullandığı tarayıcıya, günün saatine ve akla gelebilecek her türlü değişkene bağlı olarak farklı bir sayfa yaratılıp gönderilebilir. Bu şekilde MyYahoo tarzı,

tamamıyla kullanıcıya özel sayfalar yaratmak PHP ile hiç de zor değildir.

Örnek 1- Giriş örneği

```
<html>

<head>

<title>Örnek</title>

</head>

<body>

<?php echo "Merhaba Dünya!"; ?>

</body>

</html>
```

Eğer daha önceden Perl ya da C gibi dillerle CGI scripti yazdıysanız, PHP'nin HTML kodunun içine bu kadar basit bir şekilde gömülebildiğine şaşırmissinizdir. PHP kodu özel başlangıç ve bitiş tagleriyle sınırlandırılır ve bu tagler sayesinde PHP moduna girme ve PHP modundan çıkma imkanınız olur.

PHP ne yapabilir?

En basit haliyle, PHP, formdan bilgi almak, dinamik sayfa içeriği yaratmak, cookie göndermek ve almak gibi diğer bütün CGI programlarının yaptıklarını yapabilir. PHP'nin belki de en güçlü olduğu ve en çok kendini gösterdiği özelliği, geniş veritabanı desteğidir. Veritabanı kullanan bir site hazırlamak çok kolaylaştırılmıştır. Desteklenen veritabanlarını şöyle sıralayabiliriz:

Adabas
Interbase
Solid
dBase
mSQL
Sybase
Empress
MySQL
Velocis
FilePro
Oracle
Unix dbm
Informix
PostgreSQL
Microsoft SQL

PHP ile adı en çok geçen veritabanı MySQL'dir. Bizim için de öncelik MySQL olacak, örneklerde sıkça MySQL kullanımına rastlayacaksınız. Bütün bu veritabanlarının dışında, ODBC kullanarak Access gibi diğer veritabanlarına da ulaşmanız mümkün. ODBC konusuna ilerki yazılarda değineceğiz.

PHP aynı zamanda diğer hizmetlerle iletişimi sağlayan IMAP, SNMP, NNTP, POP3 ve hatta HTTP protokollerini destekler. Bütün bu protokollerin dışında soket programlayabilmenizi ve adı geçmeyen diğer protokollerle de iletişim kurabilmenizi sağlar.

Neden PHP?

Bir çoğunuzun kafasında olduğundan emin olduğumuz soru bu. Neden ASP, ColdFusion, Java değil de PHP? Cevabımız şu, basitlik, veritabanlarıyla mükemmel iletişim ve platformdan bağımsızlık. Tabii bir de, kaynak kodunun açık olması.

Tabii ki Perl ve benzeri programlar da platformdan bağımsız, ve kaynak kodu açık. Ama Perl size ilk verdiğimiz örnekteki rahatlığı sağlayamaz. HTML kodunun arasına bir tag atıp araya Perl kodu sıkıştırıramazsınız. PHP Web'te çalışmak üzere

tasarlanmıştır, bir veritabanına bağlanmak ve sorgulama yapmak iki veya üç satır kodla mümkün hale getirilmiştir. PHP özellikle 4.0 versiyonu ile yeni Zend motorunu kullanmaya başlamış, performans üst seviyeye çıkarılmıştır. Daha fazla performans için, PHP'yi Web sunucunuzun bir parçası haline getirebilmeniz dahi mümkün.

Eğer sorun sadece scriptlerin hızını arttırmak olsaydı, PHP tek başına bir çok problemin cevabı olurdu. Ama PHP'nin özellikleri bundan çok daha fazla. Programlama dili basit, veritabanlarına ulaşım kolay ve hızlı, nesne-yönelimli programlama ile tekrar kullanılabilir kodlar yazmak çok kullanışlı. PHP'de ilerledikçe, class'lar ile karşılaşacaksınız ve tek bir kodu birden fazla scriptte nasıl etkili bir şekilde kullanacağınızı öğreneceksiniz.

PHP Neden Bu Kadar Popüler?

PHP internet üzerinde oldukça hızlı yayılan dillerden biri. Yayılma hızı konusunda tek rakibi Java olabilir ki o da artık olgunluk devresini yaşıyor ve PHP ile oldukça farklı platformlarda yarışıyorlar. PHP 1995 yılında ilk kez kullanılmaya başlandığından bu yana epey yol aldı. Öncelikle linux için yazılmış diğer pek çok uygulama gibi C ile yazılmış ve kodunun herkese açık olması oldukça hızlı bir şekilde gelişmesini sağladı. Öyle ki şu anda istediğiniz bir fonksiyonu PHP için yazıp kullanabilirsiniz. 1998 ortalarında yüz bin sunucuya PHP modül olarak kurulu iken bugün bu sayı bir milyonu aşmış durumda ki bu sayının içinde CGI olarak kurulu olan sunucular yok. Ayrıca Microsoft un ASP si gibi işletim sistemine bağımlı olmaması ve 95/98/NT, Unix ve Linux türevleri üzerinde çalışabilmesi en büyük avantajı. PHP ile nesneye dayalı programlama (object oriented) yapabilir, yani sınıf ve aile yapılarını kullanabilirsiniz..