

Değişken Tipleri

Tamsayı Değişkenler (Integer)

Bilinen negatif ve pozitif tamsayılardır. $Z=\{\dots, -2, -1, 0, 1, 2, \dots\}$

```
<?php
$a=5;
$b=-24;
$c=160;
?>
```

Ondalıklı Sayı Değişkenleri (float)

Ondalık kısmı bulunan negatif ve pozitif sayılardır. Ondalıklı kısmı ayırmak için nokta (.) kullanılır.

```
<?php
$a=5.3;
$b=-2.4;
$c=1.60;
?>
```

Metin Değişkenleri (String)

Karakterlerden oluşan değerleri saklayan değişkenlerdir. Karakter olarak a-z, A-Z, 0-9 ve diğer simgeler kullanılabilir. String değişkenlere atanacak değerler çift tırnak (") arasında yazılır.

```
<?php
$adi="Ali";
$soyadi="KAVAK";
$sifre="Kxt87?_[ / ";
?>
```

String ifadeleri kullanırken çift tırnak içerisinde PHP için özel anlama sahip ifadeler varsa bunlar yorumlanır ve bunların karşılıkları kullanılır. Örneğin \$ işareti bir değişkenin varlığını söyler. Eğer çift tırnak içerisinde \$ ile başlayan bir ifade varsa bu değişken olarak algılanır ve yerine bunun değeri getirilir.

```
<?php
$gun=15;
$ay="Mart";
$yil=2010;
$tarih="Bugün $gun $ay $yil dur. ";
```

```
echo $tarih; // Ekrana Bugün 15 Mart 2010 dur. yazacaktır
$c=160;
?>
```

Eğer string ifadelerin içindeki özel anlamı olan ifadeleri devre dışı bırakmak isterseniz bu ifadelerin önüne \ koymalısınız. (Ekrana \ yazmak isterseniz \\ kullanabilirsiniz)

```
<?php
$gun=15;
$ay="Mart";
$yil=2010;
echo "\$gun=$gun \$ay=$ay \$yil=$yil";
// Ekrana $gun=15 $ay=Mart $yil=2010 yazacaktır
$c=160;
?>
```

Bir başka yöntem ise string değişken içerisindeki tüm özel ifadeleri devre dışı bırakmaktır. Bunun için çift tırnak yerine tek tırnak kullanılır. Eğer string ifade tek tırnaklar arasına yazılırsa içerisindeki hiçbir özel ifadeye bakılmaz olduğu gibi ekrana yazılır.

```
<?php
$gun=15;
$ay="Mart";
$yil=2010;
$tarih='Bugün $gun $ay $yil dur.';
echo $tarih; // Ekrana Bugün $gun $ay $yil dur. yazacaktır
$c=160;
?>
```

Örnek: Aşağıdaki örneği inceleyebilirsiniz.

```
<?php
$ali="Ben gelmiyorum!";
echo "Ali o kadar kızdı ki \"Ben gelmiyorum!\" diye bağırdı.";
echo "<br>";

$yol="c:\\windows";
echo $yol;
echo "<br>";

$tag='<font color="red" size="7">Nasıl Ama!</font>';
echo $tag;
echo "<br>";

$dere="PHP'de değişkenlerin önüne \$ işareti konulur. Örnek: \$sayi=15;";
echo $dere;
echo "<br>";
?>
```

Boolean Değişkenler

Yalnızca true yada false değerini alan değişkenlerdir. Aslında sıfırdan farklı her değer **true**, geriye kalan 0 yada null ise **false** olarak bilinir.

```
<?php
$a=true;
$b=false;
$c=1; // $c=true
$d=0; //$d=false
?>
```

Sabitler Değişkenler

Değeri değiştirilemeyen değişkenlerdir. Bunlar önceden tanımlanır ve program akışında sadece kullanılırlar. Değerleri değiştirilemez. Bir sabit **define(değişkenadı, değeri);** şeklinde tanımlanır. Aşağıdaki örneği inceleyiniz.

```
<?php
define("pi",3.14); // sabitler kullanılırken önüne $
// işareti konulmaz ve adı çift tırnak içinde tanımlanır
echo pi; // ekrana 3.14 yazar.
?>
```

Açıklamada da yazıldığı gibi sabit değişkenler tanımlanırken ve kullanılırken önüne \$ işareti konulmaz. Aşağıda sabitlerin kullanımı ile ilgili olarak yapılan birkaç hata ve açıklaması vardır.

```
<?php
define("pi",3.14); // Doğru bir sabit tanımlaması yapılmıştır
echo $pi; // Hata: Sabitin önüne $ konulmuş
pi=3; // Hata: Sabitin değeri değiştirilmeye çalışılıyor
$alan=pi*2*2; // Burada herhangi bir hata yoktur.
// Alan hesaplaması için sabit kullanılmıştır.
?>
```

Birkaç farklı sabit tanımlaması da aşağıda verilmiştir.

```
<?php
define("kullaniciadi","mustix"); // string bir sabit tanımlanmıştır
define("edit",true); // boolean bir sabit tanımlanmıştır
define("yil",2010); // integer bir sabit tanımlanmıştır
define("katsayi",0.013); // float bir sabit tanımlanmıştır.

echo kullaniciadi." ".edit." ".yil." ".katsayi." ";
// ekrana mustix 1 2010 0.013 yazar
?>
```

Dizi Değişkenler

Diziler, içerisinde bir çok değer saklayabilen değişkenler kümesi olarak tanımlanabilir. Örneğin 30 kişilik bir sınıftaki öğrencilerin adlarını saklamak üzere 30 tane değişken tanımlamaktansa 30 elemanlı bir dizi tanımlamak daha kolaydır.

Dizideki bir değer ulaşmak için indis'ler kullanılır. Aşağıdaki örneği inceleyiniz.

```
<?php
$meyveler[0]="Elma";
$meyveler[1]="Armut";
$meyveler[2]="İncir";
echo $meyveler[1]; // ekrana Armut yazar.
?>
```

Buradaki 0, 1 ve 2 indislerdir. Elma, Armut ve İncir ise değerlerdir.

Örnek: Günleri saklayan bir dizi tanımlayalım.

```
<?php
$gunler[0]="Pazartesi";
$gunler[1]="Salı";
$gunler[2]="Çarşamba";
$gunler[3]="Perşembe";
$gunler[4]="Cuma";
$gunler[5]="Cumartesi";
$gunler[6]="Pazar";
?>
```

Burada indis tanımlamaları çok esnektir. Biz burada Pazartesi için 0 kullandık. Sizler Pazartesi için 1 kullanabilirsiniz. Burada yapılan iş aslında diziye yeni değerler eklemektir. Bazen bir dizide kaçtane değer olduğunu, kullanılan enson indisin kaç olduğunu bilmediğimiz durumlar olur. Ya da indisi yazmak istemediğiniz durumlar da olabilir. Bu tür durumlarda diziye ekleme yapmak için şu yöntemi kullanabilirsiniz.

```
<?php
$gunler[]="Pazartesi";
$gunler[]="Salı";
$gunler[]="Çarşamba";
$gunler[]="Perşembe";
$gunler[]="Cuma";
$gunler[]="Cumartesi";
$gunler[]="Pazar";
echo $gunler[3]; // ekrana Perşembe yazar.
?>
```

Dizilerde indis olarak sayılar kullanmak zorunda değilsiniz. İndis olarak string ifadeler de kullanılabilir. Biz buna değerleri etiketlemek diyoruz. Bir örnekle açıklamak gerekirse. Gerçek hayatta okuldaki odalar numaralandırılmaz. Odalara isimler verilir. Örneğin müdür odasının kapısına Müdür Odası etiketini, öğretmenler odasına Öğretmenler Odası etiketini asarlar. Yani odalar etiketlenir. Benzer mantıkla bir dizideki değerler de etiketlenebilir. Aşağıdaki örneği inceleyiniz.

```
<?php
$personel["güvenlik"]="Ali Kavak";
$personel["sekreter"]="Ayşe Eren";
$personel["müdür"]="Ahmet Ergün";
$personel["hizmetli"]="Ercan Kır";
echo $personel["müdür"]; // ekrana Ahmet Ergün yazar.
?>
```

Bir dizideki değerleri yukarıdaki örneklerde olduğu gibi tek tek tanımlamak zorunda değilsiniz. Bu değerleri toplu bir şekilde tanımlamak için şu yapıyı kullanmalısınız.

```
$gunler=array("Pazartesi","Salı","Çarşamba","Perşembe","Cuma");
```

Bu tür bir tanımlamada dizinin ilk değerinin indisi 0, diğerleri sırasıyla 1, 2, ... şeklinde devam eder. Ancak siz böyle bir tanımlama yaparken indisin 0 dan başlamasını istemeyebilirsiniz. Bu tür durumlarda her değerın indisi belirtebilirsiniz.

```
$gunler=array(1=>"Pazartesi",2=>"Salı",3=>"Çarşamba",4=>"Perşembe",5=>"Cuma");
```

Eğer indisler sıralı bir şekilde ard arda gidecekse işlemi biraz daha kısaltabiliriz. Bunun için ilk değere bir indis verilir. Diğerlerine verilmediği zaman sıradaki indis değerini alırlar.

```
$gunler=array(1=>"Pazartesi","Salı","Çarşamba","Perşembe","Cuma");
```

Bu tanımlama görüldüğü gibi Pazartesi 1 indisini alırken sırasıyla Salı 2, Çarşamba 3 indislerini alacaktır. Bu işlem bu şekilde devam edecektir.

Aşağıda farklı şekilde tanımlanmış diziler bulunmaktadır. İnceleyiniz.

```
<?php
$dersler[1]="Matematik";
$dersler[2]="Fizik";
$dersler[3]="Geometri";
echo $dersler[2]; // ekrana Fizik yazar
echo "<br>";
//-----
$takim[]="Ali";
$takim[]="Ömer";
$takim[]="Erkan";
$takim[]="Serkan";
$takim[]="Emre";
echo $takim[3]; // ekrana Serkan yazar
echo "<br>";
//-----
$gorevler["pazartesi"]="Faturalar ödenecek";
$gorevler["çarşamba"]="Alışveriş yapılacak";
$gorevler["cumartesi"]="Temizlik yapılacak";
echo $gorevler["çarşamba"]; // ekrana Alışveriş yapılacak yazar.
echo "<br>";
//-----
$yaz=array("Haziran","Temmuz","Ağustos");
echo $yaz[1]; // ekrana Temmuz yazar
echo "<br>";
//-----
$bahar=array(1=>"Mart","Nisan","Mayıs");
echo $bahar[3]; // ekrana Mayıs yazar
```

```

echo "<br>" ;
//-----
$sorular=array(15=>"Ram Nedir?",65=>"CPU ne anlama gelir?",18=>"1MB=?KB");
echo $sorular[18]; // ekrana 1MB=?KB yazar.
echo "<br>" ;
//-----
$futbol=array("kaleci"=>"Can","defans"=>"Ercan","orta"=>"Ali","forvet"=>"şükrü");
echo $futbol["defans"]; // ekrana Ercan yazar
?>

```

Örnek: Aşağıdaki tabloyu dizi olarak tanımlayalım.

Linux	
KDE	Pardus
GNOME	Fedora
XFCE	Xubuntu

```

<?php
$linux=array( "KDE"=>"Pardus" , "GNOME"=>"Fedora" , "XFCE"=>"Xubuntu" ) ;
?>

```

Örnek: Aşağıdaki tabloyu dizi olarak tanımlayalım.

Diller	
1	PHP
2	PYTHON
3	C

```

<?php
$diller=array(1=>"PHP" , "PYTHON" , "C" ) ;
?>

```

Örnek: Aşağıdaki tabloyu dizi olarak tanımlayalım.

Kazananlar
Ayşe
Oya
Meral
Canan

```

<?php
$kazananlar=array("Ayşe","Oya","Meral","Canan");
?>

```

Çok Boyutlu Diziler

Dizi içindeki diziler olarak düşünülebilir. Yine bir örnekle açıklamak gerekirse bir otelin katları bir dizi olarak düşünülürse katlardaki odalar da bir dizi olarak düşünülebilir. Dolayısıyla burada dizi içindeki diziler söz konusudur. Başka bir örnek olarak şu verilebilir: Sınıftaki öğrenciler bir dizi, onları notları ise yine bir dizi olarak düşünülürse yine karşımıza dizi içindeki diziler çıkmaktadır.

Çok boyutlu dizilerde her boyutun indisi için [] kullanılır. Aşağıdaki örneği inceleyebilirsiniz.

Otel		
Kat1	Oda1	Ali
	Oda2	Ahmet
	Oda3	Ömer
Kat2	Oda1	Hasan
	Oda2	Kenan
	Oda3	Mert
Kat3	Oda1	Ayşe
	Oda2	Fatma
	Oda3	Hatice

```
<?php
$otel["kat1"]["oda1"]="Ali";
$otel["kat1"]["oda2"]="Ahmet";
$otel["kat1"]["oda3"]="Ömer";

$otel["kat2"]["oda1"]="Hasan";
$otel["kat2"]["oda2"]="Kenan";
$otel["kat2"]["oda3"]="Mert";

$otel["kat3"]["oda1"]="Ayşe";
$otel["kat3"]["oda2"]="Fatma";
$otel["kat3"]["oda3"]="Hatice";

echo $otel["kat2"]["oda3"]; // ekrana Mert yazar
?>
```

Görüldüğü gibi katlar dizinin birinci boyutu, odalar ise ikinci boyutudur. Aynı diziyi şimdi farklı bir şekilde tanımlayalım.

```
<?php
$otel=array(
    "kat1"=>array("oda1"=>"Ali","oda2"=>"Ahmet","oda3"=>"Ömer"),
    "kat2"=>array("oda1"=>"Hasan","oda2"=>"Kenan","oda3"=>"Mert"),
    "kat3"=>array("oda1"=>"Ayşe","oda2"=>"Fatma","oda3"=>"Hatice")
);

echo $otel["kat2"]["oda3"]; // ekrana Mert yazar
?>
```

Not: En son tanımlanan dizinin sonunda virgül (,) yoktur. Dikkat ediniz. Çünkü virgül dizi elemanlarını ayırır. En son elamandan sonra elaman olmadığından virgül konmaz.

Bir sınıftaki öğrencilerin matematik dersinden almış oldukları 3 notu öğrenci numaralarına göre saklayan bir dizi tanımlayalım.

Matematik		
155	1	52
	2	64
	3	35
225	1	98
	2	90
	3	70
302	1	15
	2	45
	3	63

```
<?php
$matematik[155][1]=52;
$matematik[155][2]=64;
$matematik[155][3]=35;

$matematik[225][1]=98;
$matematik[225][2]=90;
$matematik[225][3]=70;

$matematik[302][1]=15;
$matematik[302][2]=45;
$matematik[302][3]=63;
?>
```

Burada birinci boyut öğrenci numaraları (155, 225 ve 302), ikinci boyut ise sınavlar (1, 2 ve 3) olarak düşünülmüştür. Bunlara atanan sayılar ise değerler yani öğrencilerin almış oldukları notlar (52, 64, 35, ...) olarak düşünülebilir. Aynı diziyi toplu bir şekilde tanımlamak istersek aşağıdaki gibi yapabiliriz.

```
<?php
$matematik=array(
    155=>array(1=>52,2=>64,3=>35),
    225=>array(1=>98,2=>90,3=>70),
    302=>array(1=>15,2=>45,3=>63)
);
?>
```

Örnek: Aşağıdaki tabloda 3 farklı sınıfın 4'er öğrencisinin gösterildiğini kabul ederek bunu bir dizi olarak tanımlayalım.

Okul	
Birinci sınıf	Ebru
	Fuat
	Gülüzar
	Emre
İkinci sınıf	Zeynep
	Haşim

	Merve
	Nihat
Üçüncü sınıf	Raşit
	Gizem
	Onur
	Ayşe

```
<?php
$okul[0][]="Ebru";
$okul[0][]="Fuat";
$okul[0][]="Gülüzar";
$okul[0][]="Emre";

$okul[1][]="Zeynep";
$okul[1][]="Haşim";
$okul[1][]="Merve";
$okul[1][]="Nihat";

$okul[2][]="Raşit";
$okul[2][]="Gizem";
$okul[2][]="Onur";
$okul[2][]="Ayşe";

echo $okul[1][1]; // ekrana Haşim yazar
?>
```

Yukarıdaki tanımlamada dikkat ederseniz öğrenciler için indis belirtilmemiştir. Dolayısıyla öğrenciler dizinin sonuna eklenecektir. Böylece ikinci boyutun indisleri ise 0,1,2 ve 3 olacaktır. İkinci boyuttaki 0; birinci öğrenci olarak düşünülürse, 1,2 ve 3 ise sırasıyla diğer öğrenciler olarak düşünülebilir. Bu tamamen sizin tasarrufunuzda olan bir anlamlandırmadır. Aynı diziyi farklı bir şekilde tanımlayalım.

```
<?php
$okul[]=array("Ebru","Fuat","Gülüzar","Emre");
$okul[]=array("Zeynep","Haşim","Merve","Nihat");
$okul[]=array("Raşit","Gizem","Onur","Ayşe");

echo $okul[1][2]; // ekrana Merve yazar
?>
```

Dikkat ederseniz burada hiçbir indis belirtilmemiştir. Dolayısıyla her dizi elemanı dizinin sonuna eklenecektir. Böylece her iki boyutun indisleri 0 dan başlayacaktır. Eğer indislerin 1 den başlamasını isterseniz şöyle bir tanımlama yapabilirsiniz.

```
<?php
$okul[1]=array(1=>"Ebru","Fuat","Gülüzar","Emre");
$okul[]=array(1=>"Zeynep","Haşim","Merve","Nihat");
$okul[]=array(1=>"Raşit","Gizem","Onur","Ayşe");

echo $okul[1][2]; // ekrana Merve yazar
?>
```

Şimdi de aynı diziyi iki farklı şekilde daha tanımlayalım.
Aşağıdaki tanımlamada her iki boyutun indisleri 0 dan başlar.

```

<?php
$okul=array(
    array("Ebru","Fuat","Gülüzar","Emre"),
    array("Zeynep","Haşim","Merve","Nihat"),
    array("Raşit","Gizem","Onur","Ayşe")
);

echo $okul[1][2]; // ekrana Merve yazar
?>

```

Aşağıdaki tanımlamada ise her iki boyutun indisleri 1 den başlar.

```

<?php
$okul=array(
    1=>array(1=>"Ebru","Fuat","Gülüzar","Emre"),
    array(1=>"Zeynep","Haşim","Merve","Nihat"),
    array(1=>"Raşit","Gizem","Onur","Ayşe")
);

echo $okul[1][2]; // ekrana Fuat yazar
?>

```

Dizilerdeki boyut sayısını daha da artırabilirsiniz. O zaman dizi içindeki dizilerin içindeki dizilerden bahsetmiş olursunuz ki bunlarında mantığı yukarıda anlatılan iki boyutlu diziler gibidir. İhtiyaç duyulduğunda bunlar da kullanılabilir. Örneğin 3 boyutlu bir dizi olarak, okuldaki sınıflarda bulunan öğrencilerin coğrafya dersinden aldıkları 2 farklı notu saklayan bir dizi düşünülebilir.

```

<?php
// birinci sınıf
$cogرافya[0][156][]=74;
// birinci sınıfın 156 nolu öğrencisinin birinci coğrafya notudur.
$cogرافya[0][156][]=68;
// birinci sınıfın 156 nolu öğrencisinin ikinci coğrafya notudur.
$cogرافya[0][174][]=18;
// birinci sınıfın 174 nolu öğrencisinin birinci coğrafya notudur.
$cogرافya[0][174][]=23;
// birinci sınıfın 174 nolu öğrencisinin ikinci coğrafya notudur.

// ikinci sınıf
$cogرافya[1][223][]=56;
// ikinci sınıfın 223 nolu öğrencisinin birinci coğrafya notudur.
$cogرافya[1][223][]=61;
// ikinci sınıfın 223 nolu öğrencisinin ikinci coğrafya notudur.
$cogرافya[1][254][]=41;
// ikinci sınıfın 254 nolu öğrencisinin birinci coğrafya notudur.
$cogرافya[1][254][]=63;
// ikinci sınıfın 254 nolu öğrencisinin ikinci coğrafya notudur.

// üçüncü sınıf
$cogرافya[2][415][]=33;
// üçüncü sınıfın 415 nolu öğrencisinin birinci coğrafya notudur.
$cogرافya[2][415][]=51;
// üçüncü sınıfın 415 nolu öğrencisinin ikinci coğrafya notudur.
$cogرافya[2][521][]=89;
// üçüncü sınıfın 521 nolu öğrencisinin birinci coğrafya notudur.
$cogرافya[2][521][]=96;
// üçüncü sınıfın 521 nolu öğrencisinin ikinci coğrafya notudur.

```

```
echo $cografya[1][254][1];  
// ekrana 63 yazar. ikinci sınıfın 254 nolu öğrencisinin ikinci coğrafya notudur.  
?>
```

Aynı diziyi toplu bir şekilde tanımlayalım.

```
<?php  
$cografya=array(  
    array( // birinci sınıf  
        156=>array(74,68), // birinci öğrenci  
        174=>array(18,23) // ikinci öğrenci  
    ),  
    array( // ikinci sınıf  
        223=>array(56,61), // birinci öğrenci  
        254=>array(41,63) // ikinci öğrenci  
    ),  
    array( //üçüncü sınıf  
        415=>array(33,51), // birinci öğrenci  
        521=>array(89,96) // ikinci öğrenci  
    )  
);  
  
echo $cografya[1][254][1];  
// ekrana 63 yazar. ikinci sınıfın 254 nolu öğrencisinin ikinci coğrafya notudur.  
?>
```