

# DERS 1

Php ye giriş...

Öğr Gör Naciye MACİT SEZİKLİ

# Php'e Giriş

<?php

Php kodları bu taglar arasına yazılır.

?>

<?php

echo " ekrana yazdırılacak şeyler";

print " ekrana yazdırılacak şeyler";

?>

Echo ve print ekrana yazdırma işlemini sağlar.

# Php'e Giriş

```
<?php  
echo " ekrana yazdırılacak şeyler";  
print " ekrana yazdırılacak şeyler";  
?>
```

✓ Çift tırnak("") içerisinde yazılan metinlerde tekrar(" ") kullanılmaz.

✓ Kullanılması gerekirse ('') tek tırnak kullanılabilir.

" Merhaba 'php' dünyası "



" Merhaba "php" dünyası "



# Php'e Giriş

Php kodları html kodları arasında gömülür.

Kodlar sırası ile çalıştırılır ve php kodlarına sıra geldiğinde php kodları html e çevrilip işlenir.

Kullanıcı bunu farketmez.

```
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>Untitled Document</title>
6 </head>
7
8 <body>
9 Merhabaa
10 <?php
11 echo "Php dünyası ";
12 ?>
13
14 </body>
15 </html>
```

# Php'e Giriş

Php kodları html kodları arasında gömülür.

Kodlar sırası ile çalıştırılır ve php kodlarına sıra geldiğinde php kodları html e çevrilip işlenir.

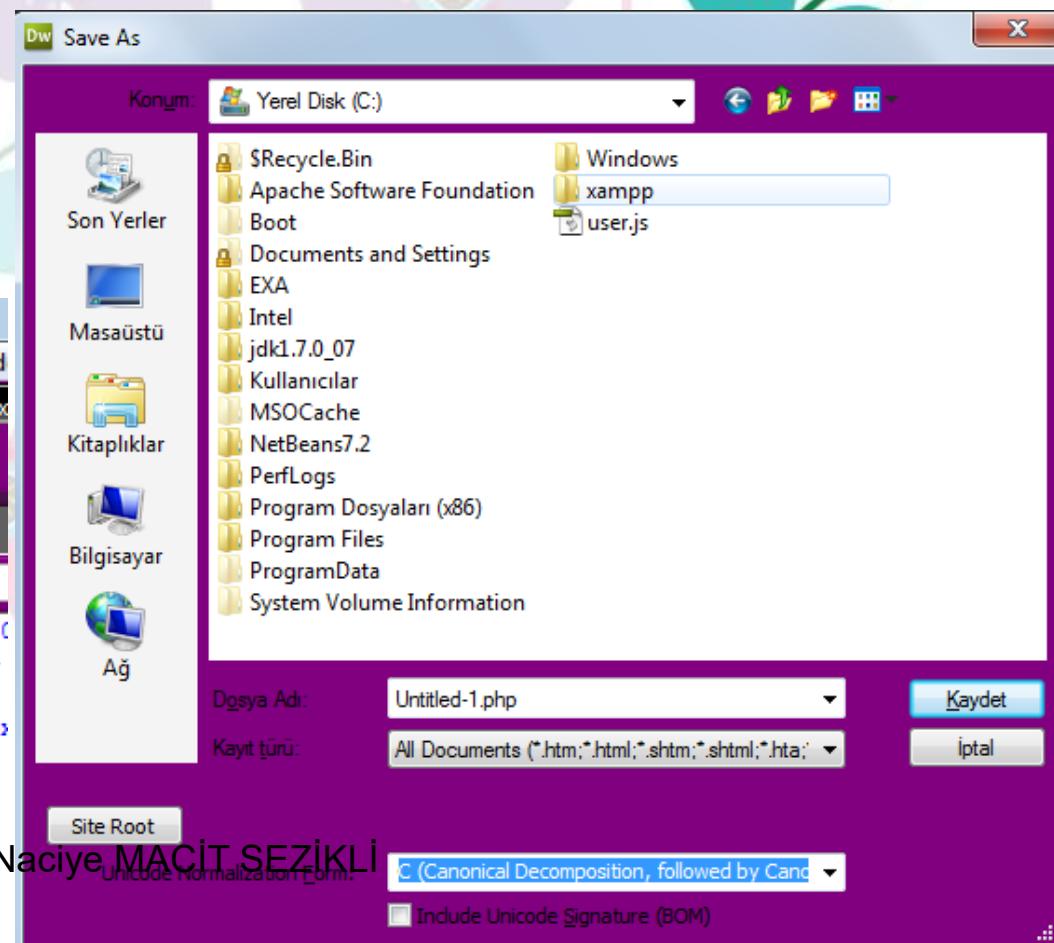
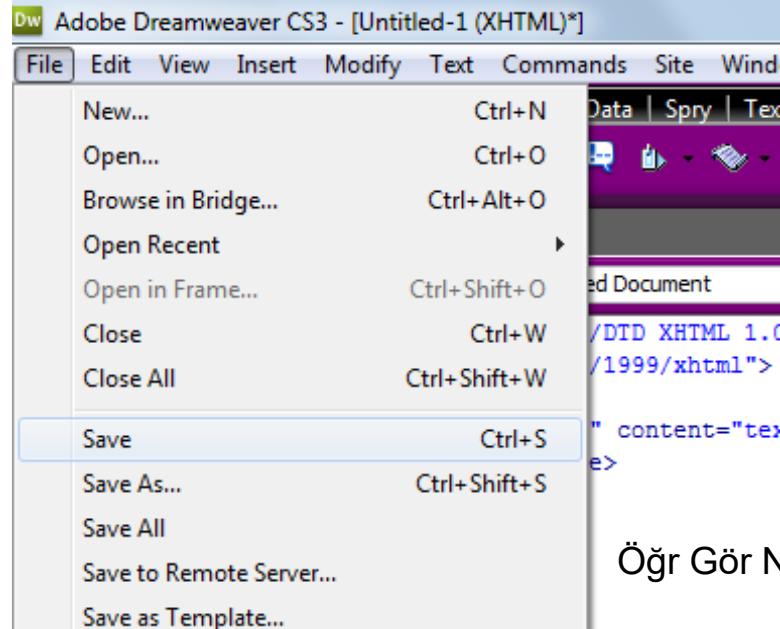
Kullanıcı bunu farketmez.

```
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>Untitled Document</title>
6 </head>
7
8 <body>
9 Merhabaa
10 <?php
11 echo "Php dünyası ";
12 ?>
13
14 </body>
15 </html>
```

# Php'e Giriş

Oluşturulan dosyayı kaydedelim .

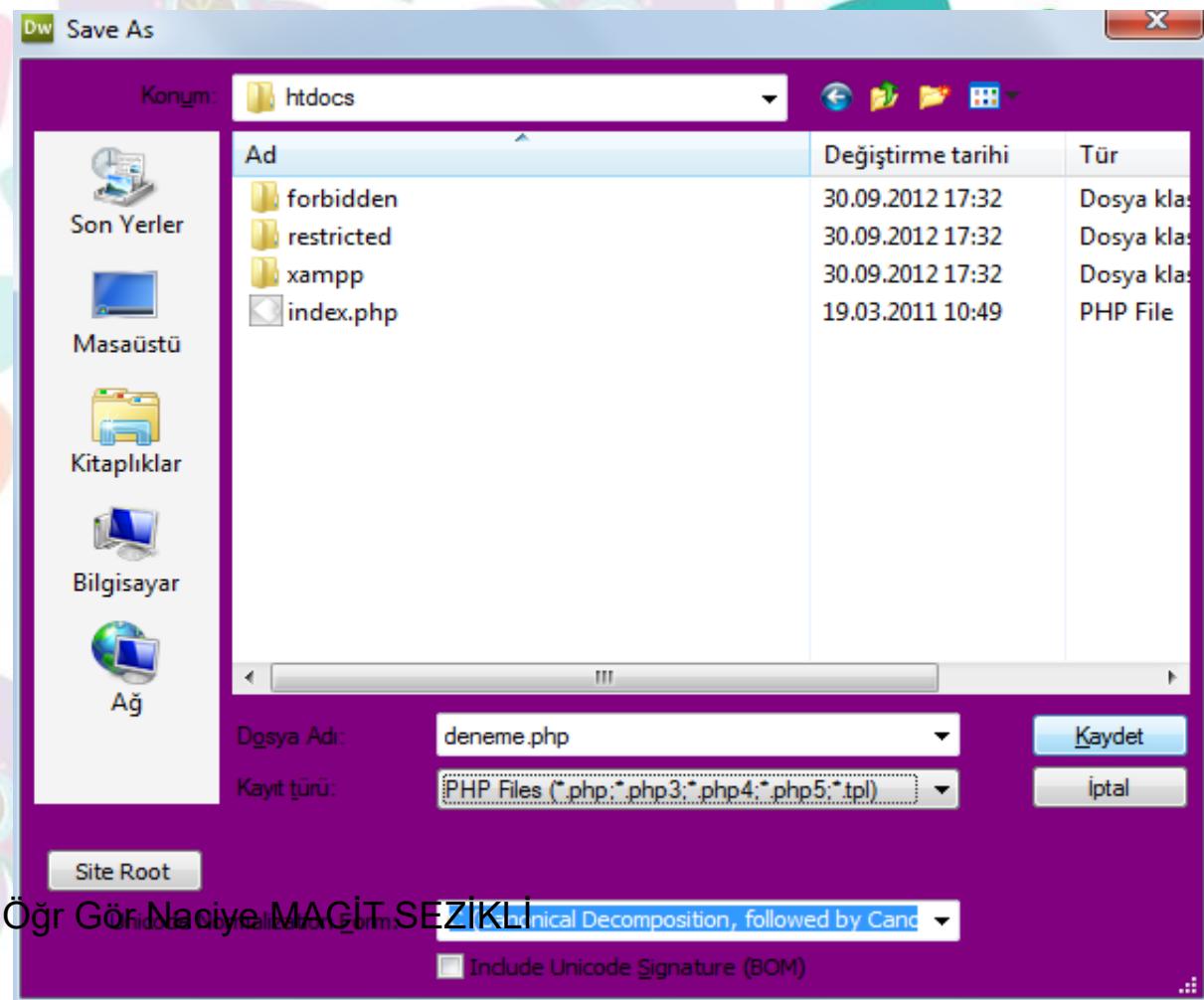
C ye kurduğumuz Xammp klasörünü, o klasörün içinde de htdoc isimli klasörü açalım.



# Php'e Giriş

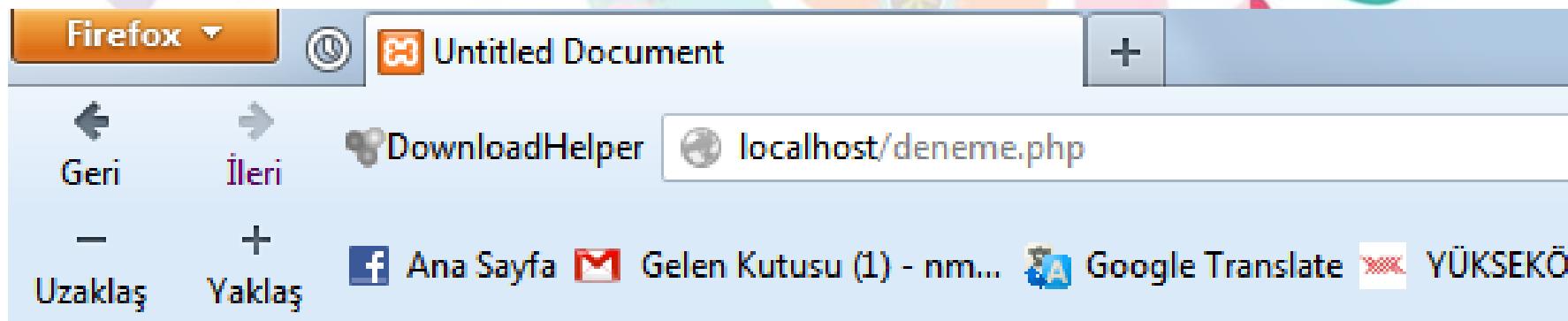
Htdoc'un içerisine oluşturduğumuz sayfayı kaydedelim.

- \*Uzantısı Php olmalı
- \*Dosya adında türkçe karakter olmamalı



# Php'e Giriş

Oluşturduğunuz sayfanın ön izlemesini görmek için internet tarayıcınızın adres çubuğuna gelip localhost/deneme.php yazın.



Merhabaa Php dünyası

VERİ

VERİ TİPI

değişken

# Değişken Tanımlama

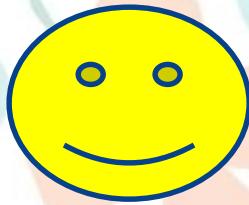
```
<?php  
$degiskenadi=değişkenin alacağı değer;  
?>
```

- ✓ Değişken adları (!),(?), Rakam ile başlamaz.
- Özel karakter, türkçe karakter, boşluk içermez.

Degiskenadi

Degiskenadi2

Degisken2adi



- Değişkenadi
- !degiskenadi
- ?degiskenadi
- 2değiskenadi
- Degisken\_adi

# Değişken Tanımlama

```
<?php  
$degisken=10;  
$degisken2="NACİYE";  
echo $degisken;  
?>
```

# Aritmetik İşlemler

(+) → Toplama İşlemi

(-) → Çıkarma İşlemi

(\*) → Çarpma İşlemi

(/) → Bölme İşlemi

(%) → Kalan İşlemi

# Örnek

```
?php
```

```
$s1=5;
```

```
$s2=10;
```

```
$s3=52;
```

```
$toplam=$s1+$s2;
```

```
$fark=$s3-$s2;
```

```
$carpim=$s1*$s2;
```

```
$bolum=$s3/$s2;
```

```
$kalan=$s3%$s1;
```

```
echo "$kalan";
```

```
?>
```

# Örnek

```
?php  
$sayi1=5;  
$sayi2=7;  
$sayi3=4;  
$toplam1=$sayi1*$sayi2+$sayi3;  
$toplam2=$sayi1*($sayi2+$sayi3);  
echo "$toplam1\n $toplam2";  
?>
```

# Nokta Operatörü

Nokta operatörü birleştirme görevi yapar.

```
<body>
```

```
Merhabaa
```

```
<?php
```

```
$ad="Naciye";
```

```
$soyad="Macit";
```

```
$yas=20;
```

```
echo "Ad Soyad:". $ad. " ". $soyad;
```

```
?>
```

```
</body>
```

# Nokta Operatörü

Html kodlarını php kodlarının içerisinde kullanabiliriz.

<br> komutu satır atlama kodudur.

```
<body>
```

```
Merhabaa
```

```
<?php
```

```
$ad="Naciye";
```

```
$soyad="Macit";
```

```
$yas=20;
```

```
echo "Ad Soyad:". $ad. " ". $soyad. "<br>";
```

```
echo "Yaşım:". $yas. "<br>";
```

```
?>
```

```
</body>
```

# gettype

Değişkenin veri tipini öğrenmemizi sağlar.

```
<?php  
$a=5;  
$b=-100;  
$c="Naciye";  
$d=0;  
$e=true;  
print gettype($a);  
print gettype($b);  
print gettype($c);  
print gettype($d);  
print gettype($e);  
?>
```

# Örnek 1

```
<?php  
$ad="Bilgisayar";  
$sonuc="$ad teknolojileri";  
echo $sonuc;  
?>
```

## Örnek 2-3

Dikdörtgene ait kenarlardan biri 4 cm biri 9 cm dir. Bu dikdörtgenin alanını ve çevresini ekrana yazdırın.

Vize notu 80, final notu 70 olan bir öğrencinin not ortalaması için vize notunun %40 i, final notunun %60 i alınmaktadır. Ortalamayı hesaplayıp ekrana yazdırın.

# Açıklama Satırları

<?php

//tek satırlı açıklama satırı

#tek satırlı açıklama satırı

/\*Bu çok satırlı

Açıklamalarda kullanılır

\*/

?>

# Standart Değişken

Program boyunca hiç değiştirmesini istemediğimiz değişkenleri tanımlarken kullanabiliriz. Mesela Pi sayısı değişmez ve program boyunca bu sayıyı kullanabiliriz. Değiştirmeden.

```
<?php  
define("PI",3.14);  
echo PI;  
?>
```

## Örnek 7

Yarı çapı 4 olarak tanımlanmış dairenin alanını ve çevresini  
sabit tanımlı pi sayısı ile hesaplayıp ekrana yazalım.

$$\text{Alan} = \pi * r * r;$$

$$\text{Çevre} = 2 * \pi * r;$$

## Örnek 7

Yarı çapı 4 olarak tanımlanmış dairenin alanını ve çevresini sabit tanımlı pi sayısı ile hesaplayıp ekrana yazalım.

Alan=pi\*r\*r;

Çevre=2\*pi\*r;

```
<?php
define ("pi",3);
$r=4;
$alan=pi*$r*$r;
$cevre=2*pi*r;
echo "<strong> Dairenin Alanı=</strong> $alan <br>";
echo "<strong> Dairenin Çevresi=</strong> $cevre <br>";
?>
```

## Örnek 8

Tanımlı olan bir sayı ile yine tanımlı olan başlığı ekrana yazdıralım.

```
<h5> Naciye Macit </h5>
```

```
$sayi=5;
```

```
$metin="Naciye Macit";
```

## Örnek 8

Tanımlı olan bir sayı ile yine tanımlı olan başlığı ekrana yazdıralım.

<br> komutları yerine \n (new line) komutunu kullanınız.

```
<?php  
$sayi=4;  
$metin="Naciye Macit";  
echo "<h$sayi> $metin </h$sayi> <br>";  
$sayi=8;  
echo "<h$sayi> $metin </h$sayi> <br>";  
$sayi=12;  
echo "<h$sayi> $metin </h$sayi> <br>";  
?>
```

## Örnek 9

Tanımlı olan bir sayı ile yine tanımlı olan başlığı ekrana yazdıralım. <br> komutları yerine \n (new line) komutunu kullanınız.

```
<font size=7 color=green>
<marquee><i> Naciye Macit </i></marquee>
</font>
```

```
<?php
$acilis=<marquee><i>";
$kapanis="</i></marquee>";
$renk="green";
$boyut=7;
$metin ="Naciye Macit"; ?>
```

## Örnek 9

Tanımlı olan bir sayı ile yine tanımlı olan başlığı ekrana yazdıralım.

<br> komutları yerine \n (new line) komutunu kullanınız.

```
<?php  
$acilis=<marquee><i>";  
$kapanis=</i></marquee>";  
$renk="green";  
$boyut=7;  
$metin ="Naciye Macit";  
$tammetin=<font size='$boyut' color='$renk'>"  
    . $acilis.$metin.$kapanis.  
    "</font>";  
echo $tammetin;  
?>
```

# String Fonksiyonları

**strtoupper( değişken)** → büyük harfe çevirir

**strtolower(değişken)** → küçük harfe çevirir

**ucfirst(değişken)** → cümledeki ilk harfi büyük yapar.

**ucwords(değişken)** → cümledeki kelimelerin  
hepsinin ilk harfini büyük yapar.

## Örnek 10

```
<?php  
$metin="Naciye Macit";  
echo strtoupper($metin);  
echo strtolower($metin);  
echo ucwords($metin);  
echo ucfirst($metin);  
?>
```

# String Fonksiyonları

**strlen( değişken);**

→string değişkenin uzunluğunu bulur

```
<?php
```

```
$metin="Naciye Macit";
```

```
echo "Değişkenin uzunluğu". strlen($metin);
```

```
?>
```

# String Fonksiyonları

**trim ( değişken);**

→değişkenin başındaki ve sonundaki boşlukları kırpar

```
<?php $metin=" Naciye Macit ";
echo "Kırılan değişkenin". trim($metin); ?>
```

**rtrim ( değişken);**

→değişkenin sonundaki boşlukları kırpar

```
<?php $metin=" Naciye Macit ";
echo "Kırılan değişkenin". trim($metin); ?>
```

# String Fonksiyonları

→değişkenin içerisinde bir kelime aranacak bu kelimededen sonra gelen karakterler yeni değişkene aktarılacak.

Yeni değişken=**Strstr** ( değişkenadı, "aranacak değer");

```
<?php $metin=" selem bugün hava çok güzel";  
$yeni=strstr($metin, "çok");  
echo "Kesilen kısım".$yeni."<br> Başındaki  
kısım".$metin;  
?>
```

# String Fonksiyonları

→ değişkenin içerisinde bir kelime bul, bu kelimeyi yeni kelime ile değiştir.

```
str_replace("aranacak değer","değiştirilecek değer",  
değişken ismi);
```

```
<?php $metin=" selem bugün hava çok güzel";  
echo "bul değiştir".  
str_replace("güzel","züpper",$metin);  
?>
```

# String Fonksiyonları

→ değişkeni tekrarlanmasılığını sağlar

**str\_repeat(değişken ismi, tekrar sayısı);**

```
<?php  
$metin=" Naciye";  
echo str_repeat($metin,5);  
?>
```

# String Fonksiyonları

→ string in içinden belli bir kısmı almak.

**substr**(değişken ismi, başlangıç karakter sayısı, alınacak karakter sayısı);

```
<?php  
$metin="Öğr Gör Naciye Macit";  
echo "Alınan kısım".substr($metin,9,6);  
?>
```

# String Fonksiyonları

→ string in içinde aranan karakterin ilkin yerini  
geri döndürür.

**strpos(değişken ismi, "aranacak karakter");**

```
<?php  
$metin="Öğr Gör Naciye Macit";  
echo "Karakterin yeri ".strpos($metin,"N");  
?>
```

# Float

```
<?php  
$sayi1=5.4;  
$sayi2=7/2;  
echo $sayi1."<br>".$sayi2;  
?>
```

# Float

→float olan değerde virgülden sonra gelen değeri 5 e eşit ve büyükse bir üst tamsayıya yuvarlar.

**round(değişken ismi);**

→virgülden sonra gösterilmesini istediğiniz basamak sayısını belirlemek içinde kullanılır

**round(değişken ismi, gösterilmesi istenen basamak miktarı);**

# Float

```
<?php  
$sayi1=5.402;  
$sayi2=7/2;  
echo round($sayi2);  
echo round($sayi1);  
?>
```

# Float

→float olan değeri bir üstündeki tamsayıya tamamlar. Virgülüden sonraki basamak önemli değil

**ceil(değişken ismi);**

→float olan değeri bir altındaki tamsayıya tamamlar. Virgülüden sonraki basamak önemli değil

**floor(değişken ismi);**

# Abs Fonksiyonu

→sayının mutlak değerini alır

**abs(sayı);**

# SQRT Fonksiyonu

→sayının karekökünü alır

`sqrt(sayı);`

# FMOD Fonksiyonu

→ Bölümden kalanı verir. % işaretini ile aynı işlevi görür.

**fmod(bölünen, bölen);**

# RAND Fonksiyonu

→ Rastgele sayı üretir.

Boş yazılırsa 0-32768 arası sayı üretir.

**rand();**

Boş yazılırsa 0-32768 arası sayı üretir.

**rand(başlangıç değeri, bitiş değeri);**

# Operatörlerin Önceliği ( ) + - / \* ÖCELİK SIRASI

1. Parantezlerin önceliği vardır

2. \* ve / ve %

+ ve -

$10/5*4;$

# Birleşik Atama Operatörü

`+ =    - =    * =    / =    . =`

`$a = $a . $b;`

`$a += $b;`

## Örnek 13

```
?php  
$a= 5;  
$b=7;  
echo $a;  
echo "<br>";  
  
$a.=$b;  
echo $a;  
?>
```

## Örnek 14

```
?php  
$ad= "Naciye";  
$soyad="Macit";  
echo $ad;  
echo "<br>";  
  
$ad.=$soyad;  
echo $ad;  
?>
```

# Artırma operatörü ++

Değişkenin önüne yazıldığında, değerinin 1 artırılmasını sağlar.

```
?php  
$x=5;  
echo $x;  
echo "<br>"  
echo ++$x;  
?>
```

# Azaltma operatörü --

Değişkenin önüne yazıldığında, değerinin 1 azaltılmasını sağlar.

```
?php  
$x=5;  
echo $x;  
echo "<br>"  
echo --$x;  
?>
```

# Karşılaştırma Operatörleri?

`==` Eşit mi?

`!=` Eşit değil mi?

`>` Büyüк mü?

`<` Küçük mü?

`>=` Büyüк veya Eşit mi?

`<=` Küçük veya Eşit mi?

# Karşılaştırma Operatörleri

**==** Eşitse

```
$s1=5;  
$s2=7;  
$s3=5;
```

**$\$s1 == \$s2$**



**$\$s1 == \$s3$**



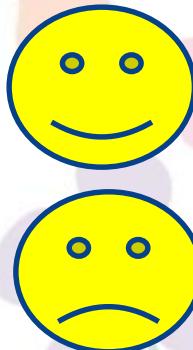
# Karşılaştırma Operatörleri

**!=** Eşit Değil mi

```
$s1=5;  
$s2=7;  
$s3=5;
```

**\$s1 != \$s2**

**\$s1 != \$s3**



# Karşılaştırma Operatörleri

> Büyük mü?

`$s1=5;`

`$s2=7;`

`$s3=4;`

`$s1>$s2`

`$s1>$s3`



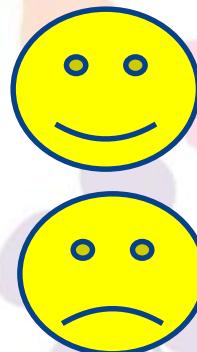
# Karşılaştırma Operatörleri

< Küçük mü?

```
$s1=5;  
$s2=7;  
$s3=4;
```

$\$s1 < \$s2$

$\$s1 < \$s3$



# Karşılaştırma Operatörleri

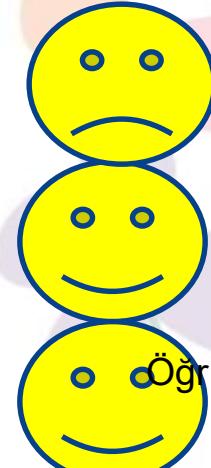
$\geq$     Büyük veya Eşit mi?

```
$s1=5;  
$s2=7;  
$s3=5;
```

$\$s1 \geq \$s2$

$\$s1 \geq \$s3$

$\$s2 \geq \$s3$



Öğr Gör Naciye MACİT SEZİKLİ

# Karşılaştırma Operatörleri

$\leq$  Küçük veya eşit mi mü?

```
$s1=5;  
$s2=7;  
$s3=5;
```

$\$s1 \leq \$s2$

$\$s1 \leq \$s3$



İF

İf (Koşul)

{

Koşul doğru ise yapılacaklar

}

# İF

```
<?php  
$a=5;  
$b=7;  
if($a<$b)  
{  
    echo "A sayısı B sayısından küçüktür!";  
}  
?>
```

# İF ELSE

if (Koşul)

{  
Koşul doğru ise yapılacaklar;

}

else

{  
Yanlışsa yapılacaklar;

}

# İF ELSE

```
<?php  
$a=51;  
$b=7;  
if($a<$b)  
{  
    echo "A sayısı B sayısından küçüktür!";  
}  
else  
{  
    echo "A sayısı B sayısından büyüktür!";  
}  
?>
```

# iç içe if

if (1.Koşul)

{

Koşul doğru ise yapılacaklar;

}

else if (2. Koşul)

{

2.Koşul doğru ise yapılacaklar;

}

else

{

Yukarıdaki koşulların hiç biri doğru değilse yapılacaklar;

}

# iÇ İÇE İF

```
<?php  
$a=5;  
$b=7;  
if($a<$b)  
{  
    echo "A sayısı B sayısından küçüktür!";  
}  
else if($a>$b)  
{  
    echo "A sayısı B sayısından büyüktür!";  
}  
else  
{  
    echo "A sayısı B sayısına eşittir!";  
}  
?>
```

# Switch Case

```
switch (durum)
```

```
{
```

```
case durum1:
```

```
{
```

```
Durumun 1 olduğu zaman yapılacaklar;
```

```
break;
```

```
}
```

```
case durum2:
```

```
{
```

```
Durumun 1 olduğu zaman yapılacaklar;
```

```
break;
```

```
}
```

```
default:
```

```
{
```

```
Durumun yukarıdaki iki durumdan farklı olduğu zaman yapılacaklar;
```

```
}
```

```
}
```

# Switch Case

```
<?php  
$sayi=5;  
Switch($sayi)  
{Case 1:  
    {  
        echo "Ocak ayı"; break;  
    }  
Case 2:  
    {  
        echo "Şubat ayı"; break;  
    }  
Default:  
    {  
        echo "böyle bir ay yok"; break;  
    }  
}  
?>
```

## Örnek 7

Sayı diye bir değişken oluşturun ve içerisinde 5 değerini verin. Sayı değişkenin içerisindeki veriye göre ekrana ayın ismini yazan sayfayı if else yapısı ile tasarlayın.

## Örnek 8

Sayı diye bir değişken oluşturun ve içerisinde 5 değerini verin. Sayı değişkenin içerisindeki veriye göre ekrana ayın ismini yazan sayfayı switch case ile tasarlayın.

## Örnek 9

Sayı diye bir değişken oluşturun ve içerisinde 5 değerini verin. Sayı değişkenin içerisindeki verinin çift olup olmadığını ekrana yazdırın.

## Örnek 10

Sayı diye bir değişken oluşturun ve içerisinde 5 değerini verin. Sayı değişkenin içerisindeki verinin 3 ün katı olup olmadığını ekrana yazdırın.

## Örnek 11

İki basamaklı bir sayının okunuşunu ekran'a yazdıran programı tasarlayın.

# $\vee \rightarrow \& \rightarrow *$

`if(ort>20 & ort<50)`

`{`

`Her iki koşulda doğru  
 ise yapılacaklar;`

`}`

1.Durum	1.Durum	Sonuç
0	0	0
0	1	0
1	0	0
1	1	1

VEYA  $\rightarrow$  ||  $\rightarrow$  +

`if(ort>20 || ort<50)`

{

Koşullardan herhangi biri doğru  
ise yapılacaklar

}

1.Durum	1.Durum	Sonuç
0	0	0
0	1	1
1	0	1
1	1	1

## Örnek 12

Vize notu=70, final notu 80 olan kişinin ortalamasını harf sistemine göre ekrana yazdırın.

# Diziler

```
<?php  
$gunler=array("Pazartesi","Salı","Çarşamba","Perşembe",  
"cuma","Cumartesi","Pazar");  
echo $gunler[0]."<br>";  
echo $gunler[1]."<br>";  
echo $gunler[2]."<br>";  
echo $gunler[3]."<br>";  
echo $gunler[4]."<br>";  
echo $gunler[5]."<br>";  
echo $gunler[6]."<br>";  
?>
```

Pazartesi
Salı
Çarşamba
Perşembe
Cuma
Cumartesi
Pazar

## Örnek 4

Yandaki dizide bazı sayı ve metinsel değerler verilmiştir.  
15 ve 10 sayısı toplanıp 0 yazan hücreye atayın.

"Internet" ve "Programcılığı" yazılarını birleştirip boş olan  
bir hücreye atayın.

15
10
"Internet"
"Programcılığı"
""
0

# Diziler(2 boyutlu)

```
<?php  
$araba=array(  
    array("BMW","Siyah",2007),  
    array("FİAT","Beyaz",2005)  
);  
echo $araba[1][0];  
?>
```

"BMW"	"Siyah"	2007
"FİAT"	"Beyaz"	2005

# Diziler(2 boyutlu)

```
<?php  
$otel=array(  
    "kat1"=>array("oda1"=>"a","oda2"=>"b","oda3"=>"c"),  
    "kat2"=>array("oda1"=>"d","oda2"=>"e","oda3"=>"f"),  
);  
echo $otel["kat1"]["oda2"];  
?>
```

	oda1	oda2	oda3
kat1	"a"	"b"	"c"
kat2	"d"	"e"	"f"

## Örnek 5

Aşağıdaki toplada ali ve ayşeye ait vize final notları bulunmaktadır. Dersin ortalaması vizenin %40'ı, finalin %60ının toplamı ile hesaplanır. Dizinin ortalama alanına alive ayşenin ortalamalarını hesaplayarak ekrana yazdırın.

	vize	final	ort
Ali	70	80	?
Ayşe	80	70	?

# DİZİ FONKSİYONLARI

Öğr Gör Naciye MACİT SEZİKLİ

# Count Komutu

Oluşturulan dizinin kaç indisli olduğunu bulmamızı sağlar.

count(\$dizininismi);

```
<?php  
$gunler=array("Pazartesi","Salı","Çarşamba","Perşembe","c  
uma","Cumartesi","Pazar");  
echo count($gunler);  
?>
```

## Örnek 6

Bir dizide bir haftanın sıcaklık değerleri tutulmaktadır.

Count komutunu kullanarak, haftanın sıcaklık ortalamasını bulup ekrana yazdırın.

# Print\_r komutu

Dizinin içerisindeki elemanları yazdırmanızı sağlar.

```
<?php  
$gunler=array("Pazartesi","Salı","Çarşamba","Perşembe  
","cuma","Cumartesi","Pazar");  
print_r($gunler);  
  
?>
```

# Max komutu

Dizinin içerisindeki max değeri geri gönderir.

```
<?php  
$gunler=array(40,55,66,47,65,23,15);  
echo max($gunler);  
  
?>
```

# Min komutu

Dizinin içerisindeki minimum değeri geri gönderir.

```
<?php  
$gunler=array(40,55,66,47,65,23,15);  
echo min($gunler);  
?  
?
```

# Sort komutu

Dizinin içerisindeki değerleri küçükten büyüğe sıralamayı sağlar.

```
<?php  
$gunler=array(40,55,66,47,65,23,15);  
sort($gunler);  
?>
```

# Rsort komutu

Dizinin içerisindeki değerleri büyükten küçüğe sıralamayı sağlar.

```
<?php  
$gunler=array(40,55,66,47,65,23,15);  
rsort($gunler);  
?  
>
```

# implode komutu

Dizinin içerisindeki değerleri tek bir string halinde birleştirmek için kullanılır.

**implode("araya konulacak değer", dizi ismi);**

```
<?php  
$gunler=array(40,55,66,47,65,23,15);  
echo implode("+", $gunler);  
  
?>
```

# explode komutu

String bir ifadeyi dizilere bölmek için kullanılır.

**explode("nerden bölüneceği", bölünecek string  
parametrenin adı);**

```
<?php  
$metin="Öğretim-Görevlisi-Naciye-Macit";  
$dizi=explode("-",$metin);  
print_r($dizi);  
  
?>
```

# in\_array komutu

Belirlenen bir ifadenin dizi içerisinde olup olmadığını bulmamızı sağlar. Eğer bulursa 1 değeri bulmazsa da hiçbir şey göndermez.

**in\_array(ArancakDeger, Dizilsmi);**

```
<?php  
$gunler=array(40,55,66,47,65,23,15);  
echo in_array(40, $gunler);  
  
?>
```

# Boolean

True(1) – False(0)

```
<?php  
$a=true;  
$b=false;  
echo $a;  
echo $b;  
?>
```

# Isset() Komutu

Değişkenin içerisinde veri olup olmadığını gösterir.

Eğer veri varsa true değeri döner, yoksa false değeri döner.

```
isset( değişkeninadı );
```

```
<?php  
$cd="Bu bir rock cd'si";  
if(isset($cd))  
{  
echo $cd;  
}  
?>
```

Değişkenin icerisinde veri	durum
Varsa	True
Yoksa	False

# Unset() Komutu

Değişkeni hafızadan silmek için kullanılır.

```
unset( değişkeninadı );
```

```
<?php  
$cd="Bu bir rock cd'si";  
if(isset($cd))  
{  
echo $cd;  
}  
else  
{  
unset($cd);  
}  
?>
```

# Empty() Komutu

Değişkenin içerisinde veri varsa false, yoksa true değeri döndürür.

`empty( değişkeninadı );`

```
<?php  
$cd="";  
if(empty($cd))  
{  
$cd="arabesk";  
}  
else  
{  
echo $cd;  
}  
?>
```

Değişkenin icerisinde veri	durum
Varsa	False
Yoksa	True

# Settype komutu

Değişkenin veri tiipini değiştirmek için kullanılır.

**settype( değişkeninadı, "değişkentipi" );**

```
<?php  
$a=5;  
echo gettype($a);  
echo $a*5;  
  
settype($a,"string");  
echo $a*5;  
echo gettype($a);  
  
?>
```

# Komutlar

**is\_array( değişkeninadı );** //Değişkenin dizi olup olmadığını kontrol eder.

**is\_bool( değişkeninadı );** //Değişkenin boolean olup olmadığını kontrol eder.

**is\_float( değişkeninadı );** //Değişkenin float olup olmadığını kontrol eder.

**is\_int( değişkeninadı );** //Değişkenin integer olup olmadığını kontrol eder.

# Komutlar

**is\_null( değişkeninadı);** //Değişkenin null olup olmadığını kontrol eder.

**is\_numeric( değişkeninadı);**  
//Değişkenin sayısal değer olup olmadığını kontrol eder.

**is\_string( değişkeninadı);** //Değişkenin string olup olmadığını kontrol eder.

# KONTROL YAPILARI- DÖNGÜLER

While Döngüsü

For Döngüsü

Foreach Döngüsü

Continue

Break

Dizi İşaretcileri

# While Döngüsü

```
While(Koşul)
{
    Koşul doğru
    ise tekrar
    edicek durum
}
```

1. Koşul Doğru Mu?

Yanlış

Doğruysa  
yapılacaklar.

# Örnek

0 dan 50 ye kadar olan sayıları  
aralarına – koyarak yazdırın.

0-1-2-3-4-.....

# Örnek

İçerisinde günlerin yazılı olduğu bir dizi oluşturun. Bu dizinin tüm elemanlarını ekrana while döngüsü ile yazdırın.

# For Döngüsü

```
for(başlangıçdeğeri ; koşul; artış/azalış miktarı)
{
    Koşul doğru olduğu müddetçe yapılacak işlem;
}
```

# Örnek

İçerisinde günlerin yazılı olduğu bir dizi oluşturun. Bu dizinin tüm elemanlarını ekrana FOR döngüsü ile yazdırın.

# Örnek

Çift boyutlu bir dizinin tüm elemanlarını ekrana sırasıyla yazdırın. For döngüsünü kullanın.

	vize	final	ort
Ali	70	80	?
Ayşe	80	70	?

# Örnek

İçerisinde günlerin yazılı olduğu çift boyutlu bir dizi oluşturun. Bu dizinin tüm elemanlarını ekrana FOR döngüsü ile yazdırın.

# Örnek

Bir aya ait her günün sıcaklık değeri dizi içeresine kaydedilmektedir. For döngüsünü kullanarak her haftaya ait ortalama sıcaklığı ekrana yazdırın programı tasarlayın.

- 1.Hafta ortalaması:50;
- 2.hafta ortalaması:30;.....

	pzt	sl	çrş	pr	cu	cmt	pz
1.Hafta							
2.Hafta							
3.Hafta							
4.Hafta							
			Öğr Gör Naciye MACİT SEZİKLİ				

# Foreach Döngüsü

Bir dizinin her elmanı için aynı işlemin yapılmasını sağlar. Sırasıyla dizi elemanı ulaşır.

```
foreach($dizi ismi as $degiskenadi)
{
echo $degiskenadi;

}
```

# Foreach Döngüsü

```
<?php  
$sayilar=array(10,11,12,15,17,16,35);,  
foreach($sayilar as $x)  
{  
echo $x;  
}  
?>
```

# Foreach Döngüsü

```
foreach($dizi ismi as $indisno => $degiskenadi)
{
echo $indisno." ". $degiskenadi;
}
```

# Foreach Döngüsü

```
<?php  
$sayilar=array(10,11,12,15,17,16,35);,  
foreach($sayilar as $y=>$x)  
{  
echo $y."indisindeki değer: ". $x;  
}  
?>
```

# Continue

Döngü yapısının içerisindeki işlemleri atlatarak işleme geçici olarak ara vererek bir sonraki adımdan devam etmesini sağlar.

```
<?php  
for ($i=0;$i<10;$i++)  
{  
if($i==5)  
{  
    continue;  
}  
echo $i."<br>";  
}  
?>
```

# Break

For-Foreach-While-Do While-Switch yapılarında kullanılır. Döngü yapısının içerisindeki diğer işlemlerin tümü atlatarak döngünün dışındaki ilk adımdan devam eder.

```
<?php  
for ($i=0;$i<10;$i++)  
{  
if($i==5)  
{  
    break;  
}  
echo $i."<br>";  
}  
?>
```

# Dizi İşaretcileri

Bir dizi tanımladığınızda, bilgisayar dizinin yerini göstermek için bir işaretçiye sahiptir. Bu değer her zaman dizinin ilk değerini gösterir.

```
<?php  
$sayi=array(1,5,9,46,7,2);  
echo "dizinin geçerli elemanı ". current($sayi)."  
next($sayi); //işaretciyi bir artırır.  
echo "dizinin geçerli elemanı ". current($sayi)."  
//dizinin geçerli elemanı artık ikinci indisdeki değerdir.  
reset($sayi); //işaretciyi sıfırlar  
echo "dizinin geçerli elemanı ". current($sayi)."  
//işaretci sıfırlandığı için ilk eleman gösterilecektir.  
?>
```

# Dizi İşaretcileri

Bir dizi tanımladığınızda, bilgisayar dizinin yerini göstermek için bir işaretçiye sahiptir. Bu değer her zaman dizinin ilk değerini gösterir.

```
<?php  
$sayi=array(1,5,9,46,7,2);  
while($a=current($sayi))  
{  
echo $a . "-" ;  
next($sayi);  
}  
?>
```

# Soru

/\*rakamlar dizinindeki 1.jpg, 2.jpg ,3.jpg ,4.jpg  
,5.jpg ,6.jpg ,7.jpg

,8.jpg ,9.jpg ve 0.jpg

resimlerin adlarını bir dizi olarak tanımlayınız.

0 ile 9 arasında ürettiğiniz rastgele

sayıya karşılık gelen resmi ekranда gösteriniz.

(Üretilen sayı 6 olursa ekran 6.jpg resmi  
görünecek)

Html kodu ile rakamlar klasörünün altındaki bir  
resmi ekranда göstermek için aşağıdaki kod  
kullanılır.

```

```

\*/

```
<?php
/*SORU 6
rakamlar dizinindeki 1.jpg, 2.jpg ,3.jpg ,4.jpg ,5.jpg ,6.jpg
,7.jpg
,8.jpg ,9.jpg ve 0.jpg
resimlerin adlarını bir dizi olarak tanımlayınız.
0 ile 9 arasında ürettiğiniz rastgele
sayıya karşılık gelen resmi ekranda gösteriniz.
(Uretilen sayı 6 olursa ekran 6.jpg resmi görünecek)
Html kodu ile rakamlar klasörünün altındaki bir
resmi ekranda göstermek için aşağıdaki kod kullanılır.

/*
$rakamlar=array("0.png","1.png","2.png","3.png","4.png","5.png",
"6.png","7.png","8.png","9.png");
///          0           1           2           3           4           5
   6           7           8           9
$i=rand(0,9);
echo "<img src='rakamlar/$rakamlar[$i]'>";
?>
```

```
//SORU7
/*
    Tanımlı olan metni font etiketinin size parametresini
    kullanarak
        bir döngü yardımıyla aşağıdaki gibi gösteriniz.
    (Bilindiği üzere size parametresi 1 ile 7 arasında bir
    değer
    alır)
Font komutu kullanımı: <font size=7>naciye
macit</font><br>
naciye macit 7
naciye macit 6
naciye macit 5
naciye macit 4
. . .
*/
$metin="Naciye MACİT SEZİKLİ";
for($i=7;$i>0;$i--){
    echo "<font size=$i>$metin</font><br>";
}
?>
```

```
//SORU 8
/* Bu sefer bir diziyle renk isimlerini tutuyoruz,
rastgele birini seçip arka planı o renkle değiştiriyoruz*/
$renkler=array("red","blue","green","yellow","orange","purple","pink"
,"cyan","gray","lime");
$i=rand(0,count($renkler)-1);
$secilenRenk=$renkler[$i];
echo "<h2>Rastgele Seçilen Renk: $secilenRenk </h2>";
echo "
<div style='width:200px; height:200px; background-color:$secilenRenk;
border:2px solid black;'></div>";

//////bütün renkler gözüksün
echo "<h2>Tüm Renkler </h2>";
foreach($renkler as $r){
echo "
<div style='display:inline-block; width:40px; height:40px;
background-color:$r; margin:3px'></div>";
}
?>
```

```
<?php
//SORU 9
/*Dizideki sorulardan rastgele birini seçip gösteren
uygulamayı yazın (örneğin küçük test sayfası).

*/
$sorular = [
    "Türkiye'nin başkenti neresidir?", 
    "En büyük gezegen hangisidir?", 
    "HTML'in açılımı nedir?", 
    "Python hangi tür bir programlama dilidir?", 
    "Atatürk'ün doğum yılı nedir?"
];
$index = rand(0, count($sorular) - 1);

echo "<h2>سؤال Bugünün Sorusu:</h2>";
echo "<p style='font-size:20px;
color:darkgreen;'>{$sorular[$index]}</p>";

?>
```

```
<?php
//SORU 10
/*Rastgele bir ülke seçecek bir ülkeler dizisinden
yine o seçtiği ülkenin bayrağını resim olarak gösterecek */

// Bayrak resimleri ve ülke adları aynı sırada dizide tutuluyor
$bayraklar = ["turkiye.png", "afganistan.png", "azerbeycan.png",
"filistin.png",
"gkore.png",
"karadag.png", "meksiка.png", "ürdün.png", "yemen.png"];
$ulkeler = ["Türkiye", "Afganistan", "Azerbeycan", "Filistin",
"GüneyKore",
"Karadağ", "Meksika", "Ürdün", "Yemen"];

// Rastgele bir ülke seç
$index = rand(0, count($bayraklar) - 1);

// Ekranda göster
echo "<h2>Rastgele seçilen ülke: {$ulkeler[$index]}</h2>";
echo "<img src='bayraklar/{$bayraklar[$index]}' alt='Bayrak'
width='200' style='border:2px solid black'>";

?>
```

# FONKSİYONLAR

\_\_\_\_\_

A  
B  
C

\_\_\_\_\_

A  
B  
C

\_\_\_\_\_

A  
B  
C

\_\_\_\_\_

\*

\_\_\_\_\_

\*

\_\_\_\_\_

# FONKSİYONLAR

function ad( \$parametreler )

{

Fonksiyon içinde yapılacaklar;

}

# Parametre Almayan ve Değer Döndürmeyen Fonksiyon

function ad()

{

Fonksiyon içinde yapılacaklar;

}

# Parametre Almayan ve Değer Döndürmeyen Fonksiyon

```
<?php  
function topla()  
{  
    $a=5;  
    $b=7;  
    echo $a+$b;  
}  
  
topla();  
?>
```

# Parametre Alan ve Değer Döndürmeyen Fonksiyon

```
<?php  
function topla($x, $y)  
{  
echo $x+$y;  
}  
  
$a=5;  
$b=7;  
topla($a,$b);  
?>
```

# Parametre Almayan ve Değer Döndüren Fonksiyon

```
<?php  
function topla()  
{$a=5;  
$b=7;  
return $a+$b;  
}  
  
$deger= topla();  
echo $deger;  
?>
```

# Parametre Alan ve Değer Döndüren Fonksiyon

```
<?php  
function topla($x, $y)  
{return $x+$y;  
}  
  
$a=5;  
$b=7;  
$deger= topla($a,$b);  
echo $deger;  
?>
```

# PHP'de FONKSİYONLAR

	PARAMETRE ALAN	PARAMETRE ALMAYAN
DEĞER DÖNDÜREN  return	<pre>function topla(\$x, \$y) {     return \$x+\$y; } \$a=5;  \$b=7; \$deger= topla(\$a,\$b); echo \$deger;</pre>	<pre>function topla() {     \$a=5;  \$b=7;     return \$a+\$b; } \$deger= topla(); echo \$deger;</pre>
DEĞER DÖNDÜRMEYEN	<pre>function topla(\$x, \$y) {     echo \$x+\$y; } \$a=5;  \$b=7; topla(\$a,\$b);</pre>	<pre>function topla() {     \$a=5;  \$b=7;     echo \$a+\$b; } topla();</pre>

# ÖRNEK

Üç değişken içerisinde yer alan metinleri bir cümle haline getiren kodları tasarlayın. fonksiyon ile bu işlemi yapın.

# ÖRNEK

Kendisine gönderilen yapılacak işleme  
 $(+, -, *, /)$  göre yine kendisine gönderilen  
iki sayı ve  $+, -, *, /$  işlemlerine tabi tutan  
bir fonksiyon oluşturun.

# FONKSİYONLarda DİZİ

```
<?php  
function islem($x,$y)  
{ $snc[0]=$x+$y;  
  $snc[1]=$x-$y;  
  return $snc; }  
  
$d=islem(10,20);  
echo $d[0];  
echo $d[1];  
?>
```

# FONKSİYONLarda DİZİ

```
<?php
    //FONKSİYONLARA DİZİ GÖNDERMEK
    /* Fonksiyona hem dizi gönderilebilir hem de geriye
    döndürülen değer bir dizi olabilir. */
    function diziIslem($gelenDizi) {
        $toplam=0;
        $carpim=0;
        foreach($gelenDizi as $deger){
            $toplam+=$deger;
            $carpim*=$deger;
        }
        return array($toplam,$carpim);
    }
    $sonuc=diziIslem(array(5,6,3,2,5,9,2));
    $sonuc=diziIslem(array(5,6,3,2,5,9,2));
    echo "Toplam=". $sonuc[0];
    echo "Çarpım=". $sonuc[1];
?>
```

# FONKSİYONLAR-GLOBAL DEĞİŞKEN

```
<?php  
$x=5;  
function islem($a)  
{  
    $sonuc=$x*$a;  
    return $sonuc;  
}  
  
echo islem(10);  
?>
```

# FONKSİYONLAR-GLOBAL DEĞİŞKEN

```
<?php  
$x=5;  
function islem($a)  
{  
    global $x;  
    $sonuc=$x*$a;  
    return $sonuc;  
}  
  
echo islem(10);  
?>
```

# FONKSİYONLAR-GLOBAL

```
$degisen=56;
```

```
{
```

```
global $degisen;
```

```
Değişkenin kullanım alanı ;
```

```
}
```

# FONKSİYONA VARSAYILAN DEĞER ATAMA

Fonksiyonun içerisinde parametre olarak bir değer gönderilmediği zaman onun yerine geçebilecek bir değer almasını şu şekilde sağlarız.

```
<?php  
function islem($a=5)  
{  
$sonuc=3*$a;  
return $sonuc;  
}  
echo islem(10);  
echo islem();  
?>
```

# ÖRNEK

Gönderilen iki sayıdan büyük olanı ekrana yazdırın yöntemi yazınız.

# ÖRNEK

Fonksiyona gönderilen sayının faktoriyelini hesaplayan metodu tanımlayın ve programınızda kullanın.

# ÖRNEK

Fonksiyona gönderilen iki sayıyı paremetre olarak alan, bu iki sayıdan birincisinin, ikinci sayıya kuvvetini hesaplayan metotu tanımlayın ve programınızda kullanın.

Örnek;

Üssü alınacak sayıyı girin :2

Üst değeri girin:5

2 üzeri 5 değeri( $2^5$ )=32

# ÖRNEK

Kaç km yol yaptığı, gaz fiyatı ve kaç litre gaz harcandığı bilgileri alınan bir aracın gidilen km için harcanan tl miktarını hesaplayan programı metot kullanarak yazınız.

Örnek;

Kaç km yol yaptı:100

Kaç litre gaz harcadı:50

Gazın litre fiyatı(tl):2.8

Km başına harcanan tl miktarı :  $50 * 2.8 / 100$   
=1.4 tl para hacamıştır.

(gazfiyatı\*litre)/yol=km başına tl miktarı

# ÖRNEK

Tanımlı olan \$boy ve \$cinsiyet değişkenlerine göre kişinin ideal kilosunu aşağıdaki

açıklamaya göre hesaplayıp ekrana **İdeal Kilonuz=...** şeklinde yazdırınız.

**Kadınlar için:** Boyunuzun ilk 150 santimetresi için 45 kilo, sonraki her 2.5 cm için 2.2

kilo ekler. Örnek olarak; boyunuz 160 cm ise, ideal kilonuz  $45+8.8=53.8$  kilogram

olacaktır.**Erkekler için:** Boyunuzun ilk 150 santimetresi için 48 kilo, sonraki her 2.5 cm için 2.7

kilo ekler. Örnek olarak boyunuz 180 cm ise, ideal kilonuz  $48+32.4=80.4$  kilogram olacaktır.

**Not:** Bu fonksiyon boyu 150 cm ve üzerinde olan kişiler içindir. Dolayısıyla gönderilen

boyun en az 150 cm olduğu varsayıılır.

# ÖRNEK

Soru1: Şifre Geçerlilik Kontrolü için bir fonksiyona şifreyi gönderin şifre ile ilgili bilgilendirme yapsın.

Aşağıda verilen PHP string fonksiyonlarını **kullanarak** bir şifre doğrulama programı yazınız.

## İstenilenler:

Kullanıcıdan alınan şifrenin aşağıdaki kurallara uyup uymadığını kontrol eden bir PHP programı yazınız:

- 1.Şifre en az **8 karakter** uzunluğunda olmalıdır.
- 2.Şifre en az **bir büyük harf** içermelidir.
- 3.Şifre en az **bir küçük harf** içermelidir.
- 4.Şifre en az **bir özel karakter (sembol)** içermelidir.
- 5.(Örnek semboller: ! @ # \$ % & \* ( ) - \_ + =)

## Örnek Çalışma:

Şifre: TesT@123 Çıktı: Şifre kurallara uygun ✓ Şifre: test123 Çıktı: Şifre en az bir büyük harf içermeli. Şifre en az bir sembol içermeli.

# ÖRNEK

Soru2: E-posta (Email) Geçerlilik Kontrolü için bir fonksiyona şifreyi gönderin şifre ile ilgili bilgilendirme yapsın.

Aşağıda verilen **PHP string fonksiyonlarını** kullanarak bir e-posta adresinin geçerli olup olmadığını kontrol eden bir program yazınız.

## İstenilenler:

Kullanıcıdan alınan e-posta adresinin aşağıdaki kurallara uyup uymadığını kontrol eden bir PHP programı yazınız:

- 1.E-posta en az **6 karakter** uzunluğunda olmalıdır.
- 2.E-posta içinde mutlaka '@' karakteri bulunmalıdır.
- 3.'@' işaretinden sonra '.' (nokta) karakteri bulunmalıdır.
- 4.'@' işaretinden önce en az **bir karakter** olmalıdır.
- 5.'' karakterinden sonra en az **iki karakter** olmalıdır.

## Örnek Çalışma:

E-posta: nmacit@gelisim.edu.tr Çıktı: Geçerli e-posta ✓ E-posta: nmacitgelisim.edu.tr Çıktı: Geçersiz e-posta X '@' karakteri bulunamadı. E-posta: @gmail.com Çıktı: Geçersiz e-posta X '@' öncesinde karakter yok.

# ÖRNEK

## Soru 3:

Bir PHP fonksiyonu yazın. Fonksiyon, kendisine gönderilen bir sayı dizisindeki en büyük ve en küçük sayıyı bulacak ve bir dizi olarak geri döndürecek.

### Şartlar:

- 1.Fonksiyon adı minMaxBul olacak.
- 2.Fonksiyon, bir sayı dizisini parametre olarak alacak.
- 3.Fonksiyon, sonucu [enKucukSayi, enBuyukSayi] formatında geri döndürecek.
- 4.Fonksiyonu kullanarak [4, 7, 1, 9, 12, 0, 5] dizisinin en küçük ve en büyük sayısını bulun ve ekrana yazdırın.

# ÖRNEK

```
<?php
// Fonksiyon tanımı
function minMaxBul($dizi) {
    $enKucuk = $dizi[0];
    $enBuyuk = $dizi[0];

    foreach ($dizi as $sayi) {
        if ($sayi < $enKucuk) {
            $enKucuk = $sayi;
        }
        if ($sayi > $enBuyuk) {
            $enBuyuk = $sayi;
        }
    }

    return [$enKucuk, $enBuyuk];
}

// Fonksiyonu kullanma
$sayilar = [4, 7, 1, 9, 12, 0, 5];
list($min, $max) = minMaxBul($sayilar);

echo "En küçük sayı: " . $min . "<br>";
echo "En büyük sayı: " . $max;
?>
```

# ÖRNEK

## Soru 4:

Bir PHP fonksiyonu yazın. Fonksiyon, kendisine gönderilen satır ve sütun sayısına göre bir HTML tablosu oluştursun.

### Şartlar:

- 1.Fonksiyon adı tabloOlustur olacak.
- 2.Fonksiyon iki parametre alacak: \$satir ve \$sutun.
- 3.Tablo hücreleri, Satır X - Sütun Y şeklinde numaralandırılacak.
- 4.Tabloya CSS ile kenarlık (border) ve biraz padding ekleyin.
- 5.Fonksiyonu kullanarak 4 satır ve 3 sütunluk tablo oluşturun.

### Örnek Çıktı (Görsel olarak tablo şeklinde):

Satır 1 - Sütun 1	Satır 1 - Sütun 2	Satır 1 - Sütun 3
Satır 2 - Sütun 1	Satır 2 - Sütun 2	Satır 2 - Sütun 3
Satır 3 - Sütun 1	Satır 3 - Sütun 2	Satır 3 - Sütun 3
Satır 4 - Sütun 1	Satır 4 - Sütun 2	Satır 4 - Sütun 3

# ÖRNEK

```
<?php
// Fonksiyon tanımı
function tabloOlustur($satir, $sutun) {
    echo '<table style="border-collapse: collapse;">';
    for ($i = 1; $i <= $satir; $i++) {
        echo '<tr>';
        for ($j = 1; $j <= $sutun; $j++) {
            echo '<td style="border: 1px solid black; padding: 8px;">';
            echo "Satır $i - Sütun $j";
            echo '</td>';
        }
        echo '</tr>';
    }
    echo '</table>';
}

// Fonksiyonu kullanma
tabloOlustur(4, 3);
?>
```

# ÖRNEK

Menü / Navigasyon Çubuğu Soru: Fonksiyona, parametre olarak menü elemanlarını alacak ve CSS ile yatay veya dikey menü oluşturacak.

```
<?php
function menuOlustur($menuElemanlari) {
    echo '<style>
        /* Menü stilini ayarlıyoruz */
        .menu {
            list-style: none;
            padding: 0;
            margin: 0;
            display: flex; /* Yatay yapmak için flex kullanıyoruz */
            background-color: #333;
        }
        .menu li {
            margin: 0;
        }
        .menu li a {
            display: block;
            padding: 12px 20px;
            text-decoration: none;
            color: white;
            transition: all 0.3s;
        }
        .menu li a:hover {
            background-color: #555;
            border-bottom: 3px solid yellow; /* Alt çizgi efekti */
        }
    </style>';

    echo '<ul class="menu">';
    foreach ($menuElemanlari as $eleman) {
        echo '<li><a href="#">' . $eleman . '</a></li>';
    }
    echo '</ul>';
}

?>
```

# HTML Karakter Referansları (Character Entities)

- HTML'de bazı karakterler doğrudan yazılamaz veya özel anlam taşır (örn: <, >).
- Bu karakterler için **entity kodları** kullanılır: &entity; veya &#kod;

Karakter	Kod	Açıklama
★	&#9733;	Dolu yıldız
☆	&#9734;	Boş yıldız
©	&copy;	Telif işaretü
<	&lt;	Küçükür işaretü
<	&gt;	Büyükür işaretü

Unicode tablosunda arama yapabilirsiniz:  
link: <https://symbl.cc/>

# ÖRNEK

Fonksiyon, parametre olarak bir sayı alacak (ör: 1–5) ve CSS ile stil verilmiş yıldız puan sistemi gösterecek.

Örnek: puanGoster(3) → 3 dolu yıldız, 2 boş yıldız

← → ⌂ i localhost/InternetProgram

FILE GELİŞİM ÜNİVERSİTESİ PERSİS



```
<?php  
function puanGoster($puan) {  
    // CSS stilini ekliyoruz  
    echo '<style>  
        .yildiz {  
            font-size: 30px;  
            color: gray; /* boş yıldız rengi */  
        }  
        .dolu {  
            color: gold; /* dolu yıldız rengi */  
        }  
    </style>';  
  
    // Yıldızları yazdırıyoruz  
    for ($i = 1; $i <= 5; $i++) {  
        if ($i <= $puan) {  
            echo '<span class="yildiz dolu">#9733;</span>'; // dolu yıldız  
        } else {  
            echo '<span class="yildiz">#9733;</span>'; // boş yıldız  
        }  
    }  
}  
?  
?
```

# ÖRNEK

Fonksiyon, parametre olarak bir sayı alacak (ör: 1–5) ve CSS ile stil verilmiş yıldız puan sistemi gösterecek.

Örnek: puanGoster(3) → 3 dolu yıldız, 2 boş yıldız

# FONKSİYONLARA DİZİ GÖNDERMEK

```
<?php
    //FONKSİYONLARA DİZİ GÖNDERMEK
    /* Fonksiyona hem dizi gönderilebilir hem de
geriye
    döndürülen değer bir dizi olabilir. */
    function diziIslem($gelenDizi) {
        $toplam=0;
        $carpim=0;
        foreach($gelenDizi as $deger){
            $toplam+=$deger;
            $carpim*=$deger;
        }
        return array($toplam,$carpim);
    }
    $sonuc=diziIslem(array(5,6,3,2,5,9,2));
    $sonuc=diziIslem(array(5,6,3,2,5,9,2));
    echo "Toplam=". $sonuc[0];
    echo "Çarpım=". $sonuc[1];
?>
```

# FONKSİYONLARA DEĞER OLARAK REFERANS GÖNDERME

```
<?php
```

```
//FONKSİYONLARA DEĞER OLARAK REFERANS GÖNDERME
//ADRES GÖNDERME
/**Normalda fonksiyona gönderilen parametreler değerleridir.
 * ama bazen değer yerine parametrenin adresini gönderip
 * içerisindeki değere müdahale edebilmek isteriz.
 */
function degistir($sayi){
    $dayi=55;
}
function referansDegistir(& $sayi){
    $sayi=4;
}
$sayi=1;
degistir($sayi); //değeri gönderiyoruz yani $sayi değişmeyecek
echo $sayi."<br>";
referansDegistir($sayi); // sayının adresini gönderiyoruz-
icerigine ulaşılacak
echo $sayi;
?>
```

# FONKSİYONLARA GÖNDERİLEN YAPAY PARAMETRELER

```
<?php
```

```
    //FONKSİYONLARA GÖNDERİLEN YAPAY PARAMETRELER
```

```
    /**Eğer fonksiyona gönderilecek paramtere sayısında belirsizlik  
varsa bu yöntemleri kullanırız
```

▣ Çalışma Mantığı:

`func_num_args()` → kaç tane argüman geldiğini verir.

`func_get_args()` → tüm argümanları dizi olarak verir.

`func_get_arg(n)` → n. argümanı döndürür (örnek: `func_get_arg(0)` ilkini verir). \*/ ?>

# FONKSİYONLARA GÖNDERİLEN YAPAY PARAMETRELER

```
<?php
function argumanlar(){
$adet=func_num_args();
if($adet==0) return;

echo $adet. 'parametre bulundu'."<br>";
if($adet>1){
    $dizi=func_get_args();
    foreach($dizi as $deger){
        echo $deger.'<br>';
    }
}
else
{
    echo func_get_arg(0)."<br>";
}
argumanlar(3);
// argumanlar('a','b','c');
// argumanlar(2,6,9,52,2);
?>
```