

Php 3. Sunum

Fonksiyonlar

Tarih ve Saat Fonksiyonları

Matematiksel Fonksiyonlar

Karakter Kümesi Fonksiyonları

Mail Gönderme

PHP'de Fonksiyonlar

- Fonksiyonlar bizi defalarca kod yazmaktan kurtaran, daha modüler iş yaparak zamandan ve emekten tasarruf sağlayan yapılardır. PHP'de fonksiyonların yazım şekli C,C++,Java, JavaScript ile hemen hemen aynıdır.

PHP'de Fonksiyon Tanımlama

- //Parametresiz
- `function fonksiyon_adi()`
- `{`
- `.....`
- `}`

- //Parametrelili
- `function fonksiyon_adi(parametre1,parametre2,...)`
- `{`
- `.....`
- `return (donus_degeri);`
- `}`

Fonksiyonlar

- ```
function yaz()
{
 echo "yaz fonksiyonu";
}
```
- ```
function yaz2($ad,$soyad)  
{  
    echo "merhaba $ad $soyad";  
}
```
- Yukarıda ilk yaz fonksiyonu parametre almamıştır, ikincisi ise ad ve soyad parametrelerini almıştır.

Fonksiyonun Çağırılması:

```
yaz();  
$isim="Ali";  
$soyisim="Ak";  
yaz2 ($isim, $soyisim);
```

Fonksiyonlar

- `//Parametresiz`
- `function yaz()`
- `{`
- `echo "merhaba fonksiyon";`
- `}`

- `//Parametrelili`
- `function mesaj($deger)`
- `{`
- `echo $deger;`
- `}`

Fonksiyonun Çağırılması:

```
yaz ();  
$deger="merhaba  
fonksiyon";  
mesaj ($deger) ;
```

Fonksiyonlar - Örnek

- //Aşağıdaki fonksiyon verilen sayı aralığını toplar.
<?php
- **function sayitopla(\$ilk,\$son)**
- **{**
- **\$toplam=0;**
- **for (\$i=\$ilk; \$i<=\$son ;\$i++)**
- **{**
- **\$toplam=\$toplam + \$i;**
- **}**
- **return (\$toplam);**
- **}**
- **\$num1=100; \$num2=500;**
- **\$sonuc= sayitopla(\$num1, \$num2);**
- **echo "\$num1-\$num2 arası toplam =\$sonuc";**
- **?>**

Fonksiyonlar - İşlemler

- `<?php`
- `function topla($a, $b) { return ($a + $b); }`
- `function carp($a, $b) { return ($a*$b); }`
- `function mod_al($a, $b) { return ($a % $b); }`
- `$num1 = 8; $num2 = 3;`
- `$sonuctoplam = topla($num1, $num2);`
- `$sonuccarpim = carp($num1, $num2);`
- `$sonucmod = mod_al($num1, $num2);`
- `echo "Toplam sonucu: $sonuctoplam
";`
- `echo "Çarpım sonucu: $sonuccarpim
";`
- `echo "Mod sonucu: $sonucmod
";`
- `?>`

Fonksiyonlar – Harf Notu Bulma

- <?php
- function hesapla(\$not1, \$not2, \$not3)
- {
- \$sonuc = (\$not1 + \$not2 + \$not3)/3;
- if (\$sonuc>84 && \$sonuc<=100) \$harf='A';
- elseif (\$sonuc>=70 && \$sonuc<=84) \$harf='B';
- elseif (\$sonuc>=55 && \$sonuc<=69) \$harf='C';
- elseif (\$sonuc>=45 && \$sonuc<=54) \$harf='D';
- elseif (\$sonuc>=0 && \$sonuc<=44) \$harf='E';
- return(\$harf);
- }
- \$n1=90; \$n2=80; \$n3=100;
- \$sonucnot = hesapla(\$n1, \$n2, \$n3);
- echo "\$n1-\$n2-\$n3 Harf notu =\$sonucnot";
- ?>

Fonksiyonlar - Faktöriyel

- `<?php`
- `function faktoriyel($sayi)`
- `{`
- `if ($sayi==0) return(1);`
- `$fakt=1;`
- `for ($i=1; $i<=$sayi ; $i++)`
- `{ $fakt= $fakt * $i; }`
- `return ($fakt);`
- `}`
- `$num =6;`
- `$sonuc = faktoriyel ($num);`
- `echo $sonuc; ?>`

Dizinin Fonksiyona Gönderilmesi-Örnek

- <?
- function liste_yaz(\$gelendizi)
- {
- for (\$i=0;\$i<count(\$gelendizi);\$i++)
- echo \$gelendizi[\$i]."
";
- }
- \$gelendizi=array(6,8,10,7);
- liste_yaz(\$gelendizi);
- ?>

Tarih ve Saat Fonksiyonları

- Getdate()
- Date()
- Time()
- Mktime()

getdate() fonksiyonu

- Tarih/zaman bilgisi getirir.
- <?
- \$bugun=getdate();
- print \$bugun[mday]."-".\$bugun[mon]."-".\$bugun[year];
- //Bugünkü tarihi yazar
- ?>

Getdate fonksiyonunda kullanılan parametreler

"seconds"	Saniyelerin sayısal gösterimi	0'dan 59'a kadar
"minutes"	Dakikaların sayısal gösterimi	0'dan 59'a kadar
"hours"	Saatlerin sayısal gösterimi	0'dan 23'e kadar
"mday"	Ayın günlerinin sayısal gösterimi	1'den 31'e kadar
"wday"	Haftanın günlerinin sayısal gösterimi	(Pazar için) 0'dan (Cumartesi için) 6'ya kadar
"mon"	Ayın sayısal gösterimi	1 'den 12'ye kadar
"year"	Yılın 4 haneli sayısal tam gösterimi	Örnek: 1999 ya da 2003 gibi
"yday"	Yılın gününün sayısal gösterimi	0'dan 365'e kadar
"weekday"	Haftanın gününün metinsel tam gösterimi	Sunday'dan Saturday'ye
"month"	Ayın metinsel tam gösterimi, Ocak ya da Mart gibi	January'dan December'a kadar

date() fonksiyonu

- Yerel tarihi/saati biçimlendirir.

<i>biçim karakteri</i>	Açıklama	Örnek sonuç
Gün	---	---
<i>d</i>	Ay günlerinin sıfır dolgulu iki haneli gösterimi	<i>01'den 31'e</i>
<i>D</i>	Ay günlerinin üç harfli metinsel gösterimi	<i>Pzt'den Paz'a</i>
<i>j</i>	Ay günlerinin sıfır dolgunsuz gösterimi	<i>1'den 31'e</i>
<i>l</i> (küçük 'L' harfi)	Hafta günlerinin tam metinsel gösterimi	<i>Pazar'dan Cumartesi'ye</i>
<i>N</i>	Hafta günlerinin ISO-8601 standardında sayısal gösterimi (PHP 5.1.0'da eklenmiştir)	<i>1'den (Pazartesi için) 7'ye (Pazar için)</i>
<i>S</i>	Ay günleri için 2 karakterli İngilizce sıralama ekleri	<i>st, nd, rd ya da th. j ile uyumlu çalışmaktadır.</i>
<i>w</i>	Hafta günlerinin sayısal gösterimi	<i>0'dan (Pazar için) 6'ya (Cumartesi için)</i>
<i>z</i>	Yılın günleri (0'dan başlar)	<i>0'dan 365'e</i>

date() fonksiyonu

<i>Hafta</i>	---	---
<i>W</i>	ISO-8601 standartına göre yılın hafta numarası, hafta başlangıcı Pazartesi'dir	Örneğin: 42 (Yılın 42'inci haftası)
<i>Ay</i>	---	---
<i>F</i>	Ayın tam metinsel gösterimi, Ocak ya da Mart gibi	<i>January'den December'a</i>
<i>m</i>	Ayın sıfır dolgulu sayısal gösterimi	<i>01'den 12'ye</i>
<i>M</i>	Ayın üç karakterli, metinsel kısa gösterimi	<i>Jan'dan Dec'e kadar</i>
<i>n</i>	Ayın sıfır dolgusuz sayısal gösterimi	<i>1'den 12'ye kadar</i>
<i>t</i>	Belirtilen ayın gün sayısı	<i>28'den 31'e</i>

date() fonksiyonu

<i>Yıl</i>	---	---
<i>L</i>	Artık yıl olduğunda	Eğer artık yılsa 1, değilse 0.
<i>o</i>	ISO-8601 standardı yıl numarası. ISO hafta numaralarının (<i>W</i>) ait olduğu yılın gösterilmesi dışında <i>Y</i> ile aynı değere sahiptir.	Örnekler: 1999 ya da 2003 gibi
<i>Y</i>	Yılın 4 haneli sayısal, tam gösterimi	Örnekler: 1999 ya da 2003 gibi
<i>y</i>	Yılın iki haneli gösterimi	Örnekler: 99 ya da 03 gibi

date() fonksiyonu

<i>Saat</i>	---	---
<i>a</i>	Küçük harfli öğleden önce ve öğleden sonra	<i>am</i> ya da <i>pm</i>
<i>A</i>	Büyük harfli öğleden önce ve öğleden sonra	<i>AM</i> ya da <i>PM</i>
<i>B</i>	Swatch İnternet saati	000'dan 999'a
<i>g</i>	Saatın, 12-saatlik sıfır dolgusuz gösterimi	<i>1</i> 'den <i>12</i> 'ye
<i>G</i>	Saatın, 24-saatlik sıfır dolgusuz gösterimi	<i>0</i> 'dan <i>23</i> 'e
<i>h</i>	Saatın, 12-saatlik sıfır dolgulu gösterimi	<i>01</i> 'den <i>12</i> 'ye
<i>H</i>	Saatın, 24-saatlik sıfır dolgulu gösterimi	<i>00</i> 'dan <i>23</i> 'e
<i>i</i>	Sıfır dolgulu dakika gösterimi	00 ile 59 arasında
<i>s</i>	Sıfır dolgulu saniye gösterimi	00 ile 59 arasında
<i>u</i>	Mikrosaniye (PHP 5.2.2 sürümüyle eklenmiştir)	Örneğin: 654321

date() fonksiyonu

<i>Zaman dilimi</i>	---	---
<i>e</i>	Zaman dilimi belirteci	Örnekler: <i>UTC, GMT, Europe/Istanbul</i>
<i>I</i> (büyük <i>i</i>)	Yaz saati uygulaması var mı?	Varsa <i>1</i> , yoksa <i>0</i> .
<i>O</i>	Saat olarak Greenwich zamanı (GMT) farkı	Örneğin: <i>+0200</i>
<i>P</i>	Saat ve dakika olarak Greenwich zamanı (GMT) farkı	Örneğin: <i>+02:00</i>
<i>T</i>	Zaman dilimi kısaltması	Örnekler: <i>EST, EET</i> gibi
<i>Z</i>	Saniye cinsinden saat farkı. UTC'nin batısı daima negatif, doğusu ise daima pozitif değerlidir.	<i>-43200'den 50400'e</i>

date() fonksiyonu

- <?
- `print date("d-m-Y");`
- `//13-03-2011` gibi sistem tarihini yazar.
- ?>

time() fonksiyonu

- Sunucunun o anki zaman bilgisini saniye olarak döndürür. (1 Ocak 1970 den başlayarak)

- <?php

```
$gelecekHafta = time() + (7 * 24 * 60 * 60);
```

```
    // 7 gün; 24 saat; 60 dakida; 60 saniye
```

```
echo 'Şimdi:      '. date('d-m-Y') ."\n";
```

```
echo 'Gelecek Hafta:'. date('d-m-  
Y', $gelecekHafta) ."\n";
```

```
?>
```


mktime() fonksiyonu

- mktime() fonksiyonu, kendisine verilen verilen tarih ile 1 Ocak 1970 arasındaki farkı hesaplar.
- Örnek olarak 1 Ocak 1970 ile 22 Kasım saat 08:00 arasındaki farkı şu şekilde bir sayı ile geri döndürür:
- **1227340850**
- Burda görüldüğü üzere 1 Ocak 1970den bu yana ne kadar saniye geçtiğidir!

mktime() fonksiyonu

- Genel kullanımı;
- \$saat = 08;
- \$dakika = 0;
- saniye = 0;
- \$ay = 11;
- \$gun = 22;
- \$yil = 2008;
- \$sonuc = mktime(\$saat, \$dakika, \$saniye, \$ay, \$gun, \$yil);
- echo \$sonuc; //1227340850

mktime() fonksiyonu

- ```
<?php
echo date("M-d-Y", mktime(0, 0, 0, 12, 32, 1997));
echo date("M-d-Y", mktime(0, 0, 0, 13, 1, 1997));
echo date("M-d-Y", mktime(0, 0, 0, 1, 1, 1998));
echo date("M-d-Y", mktime(0, 0, 0, 1, 1, 98));
?>
```
- "Jan-01-1998" tarihini yazar.



# Matematiksel Fonksiyonlar

- `abs()`
- `floor()`
- `ceil()`
- `round()`
- `pow()`
- `sqrt()`
- `max()`
- `min()`
- `log10()`

# abs() fonksiyonu

- abs fonksiyonu matematikdeki **mutlak değerin** işini görür.
- `<?php`
- `$abs = abs(-4.2); // $abs = 4.2; (double/float)`
- `echo $abs."<br>";`
- `$abs2 = abs(5); // $abs2 = 5; (integer)`
- `echo $abs2;`
- `?>`



# floor() fonksiyonu

- Ondalık sayıların küçük sayıya yuvarlanması için kullanılır.
- ```
<?php  
echo floor(4.3); // 4  
echo floor(9.999); // 9  
echo floor(-3.14); // -4  
?>
```


ceil() fonksiyonu

- Ondalık sayıların büyük sayıya yuvarlanması için kullanılır.
- ```
<?php
echo ceil(4.3); // 5
echo ceil(9.999); // 10
echo ceil(-3.14); // -3
?>
```

# round() fonksiyonu

- Ondalık sayıların yuvarlanması için kullanılır.
- Kullanım Şekli
- round(yuvarlanacak sayı,noktadan sonra kaç basamak kullanılacak)
- ```
<?php
echo round(3.4); // 3
echo round(3.5); // 4
echo round(3.6); // 4
echo round(3.6, 0); // 4
echo round(1.95583, 2); // 1.96
echo round(1241757, -3); // 1242000
echo round(5.045, 2); // 5.05
echo round(5.055, 2); // 5.06
?>
```


pow() fonksiyonu

- Bir sayının üstel değerlerini bulmak için kullanılır.
- Kullanım Şekli
- `pow(sayı,kuvvet değeri)`

- `<?php`
- `echo pow(2, 8); // 256`
- `echo pow(-1, 20); // 1`
- `echo pow(0, 0); // 1`
- `echo pow(-1, 5.5); // NAN(tanımsız)`
- `?>`

sqrt() fonksiyonu

- Bir sayının karekökünü alır.
- ```
<?php
echo sqrt(9); // 3
echo sqrt(10); // 3.16227766 ...
?>
```

# max() fonksiyonu

- Verilen değerler içinde en büyüğünü bulur
- `echo max(1, 3, 5, 6, 7); // 7`  
`echo max(array(2, 4, 5)); // 5`

```
echo max(0, 'hello'); // 0
echo max('hello', 0); // hello
```

```
echo max('42', 3); // '42'
```

```
// Farklı arraylarda uzun olanı bulur
$val = max(array(2, 2, 2), array(1, 1, 1, 1)); // array(1, 1, 1, 1)
```

```
// Aynı uzunluktaki arraylerde elemanlarına bakılır.
```

```
Örnek: 2 == 2, fakat 4 < 5
```

```
$val = max(array(2, 4, 8), array(2, 5, 7)); // array(2, 5, 7)
```

```
?>
```



# min() fonksiyonu

- Verilen değerler içinde en küçük olanı bulur.

- <?php

```
echo min(2, 3, 1, 6, 7); // 1
```

```
echo min(array(2, 4, 5)); // 2
```

```
?>
```



# log10() fonksiyonu

- 10 tabanına göre logaritma alır.
- < ?PHP
- echo log10(100);
- ?>
  
- Sonuç:2

# Karakter Kümesi Fonksiyonları

explode()

ltrim()

rtrim()

trim()

number\_format()

printf()

sprintf()

str\_pad()

str\_repeat()

str\_replace()

strcmp()

strlen()

strtolower()

strtoupper()

substr()

substr\_replace()

wordwrap()



# explode() fonksiyonu

- Bir dizgeyi bir ayrıca göre bölüp bir dizi haline getirir.

- <?php

```
$pizza = "dilim1 dilim2 dilim3 dilim4 dilim5 dilim6";
```

```
$dilimler = explode(" ", $pizza);
```

```
echo $dilimler[0]; // dilim1
```

```
echo $dilimler[1]; // dilim2
```

```
?>
```



# ltrim() fonksiyonu

- Dizgenin başındaki boşlukları (veya diğer karakterleri) budar. Aşağıdaki karakterleri budar.
- " " (ASCII 32 (0x20)), sıradan boşluk.
- "\t" (ASCII 9 (0x09)), sekme.
- "\n" (ASCII 10 (0x0A)), satırsonu.
- "\r" (ASCII 13 (0x0D)), satırbaşı.
- "\0" (ASCII 0 (0x00)), NUL-bayt.
- "\xB" (ASCII 11 (0x0B)), dikey sekme.

```
<?php
$metin = "\t\tBir kaç kelam :) ... ";
print "\n";
$kirpik = ltrim($metin);
echo $kirpik
$kirpik = ltrim($metin, " \t.");
echo $kirpik;
?>
```

# rtrim() fonksiyonu

- Dizgenin sonundaki boşlukları (veya diğer karakterleri) budar.
- Aşağıdaki karakterleri budar.
- " " (ASCII 32 (0x20)), sıradan boşluk.
- "\t" (ASCII 9 (0x09)), sekme.
- "\n" (ASCII 10 (0x0A)), satırsonu.
- "\r" (ASCII 13 (0x0D)), satırbaşı.
- "\0" (ASCII 0 (0x00)), NUL-bayt.
- "\x0B" (ASCII 11 (0x0B)), dikey sekme.

```
<?php
$metin="Bir kaç kelam :) ... ";
echo rtrim($metin);
?>
```



# trim() fonksiyonu

- Dizgenin başındaki ve sonundaki boşlukları (veya diğer karakterleri) budar. Aşağıdaki karakterleri budayacaktır.
- " " (ASCII 32 (0x20)), sıradan boşluk.
- "\t" (ASCII 9 (0x09)), sekme.
- "\n" (ASCII 10 (0x0A)), satırsonu.
- "\r" (ASCII 13 (0x0D)), satırbaşı.
- "\0" (ASCII 0 (0x00)), NUL-bayt.
- "\xB" (ASCII 11 (0x0B)), dikey sekme.

```
<?php
$metin=" Bir kaç kelam :) ... ";
echo trim($metin);
?>
```



# number\_format()

- Sayıyı binlik bölümlere ayırır.
- Kullanımı:
- **number\_format** (sayı , ondalık\_hane, ondalık\_ayracı, binlik\_ayracı )
- ```
<?php  
$sayi = 1234.56;  
$strformat = number_format($sayi, 2, ',', '.');  
echo $strformat;
```
- ```
// 1.234,56
?>
```

# printf() fonksiyonu

- Biçimli bir dizgi çıkartır.
- `<?php`
- `$num=1000;`
- `$kelime="emyo";`
- `printf('Sayı:%d',$num);`
- `printf('Sayı:%s',$kelime);`
- `?>`



# sprintf() fonksiyonu

- Biçimli bir dizge çıkartır.
- `<?php`
- `$num=1000;`
- `$kelime="emyo";`
- `echo sprintf('Sayı:%d',$num);`
- `echo sprintf('Sayı:%s',$kelime);`
- `?>`



# str\_pad fonksiyonu

- Bir dizgeyi belli bir uzunlukta diğer dizgeyle doldurur.
- Kullanımı: **str\_pad** (girdi , dolgulu\_uzunluk [, dolgu\_dizgesi = " " [, dolgu\_türü = *STR\_PAD\_RIGHT* ] ] )
- 
- *dolgu\_türü*: Seçimlik *dolgu\_türü* olarak **STR\_PAD\_RIGHT** (sağ), **STR\_PAD\_LEFT** (sol) veya **STR\_PAD\_BOTH** sabiti (her iki taraf) belirtilebilir. *dolgu\_türü* belirtilmezse **STR\_PAD\_RIGHT** değeri öntanımlıdır.

# str\_pad fonksiyonu-örnek

- ```
<?php
$input = "Ayran";
echo str_pad($input, 10); // "Ayran  " üretilir
echo str_pad($input, 10, "-=", STR_PAD_LEFT); // "-==--Ayran" üretilir
echo str_pad($input, 10, "_", STR_PAD_BOTH); // "__Ayran__" üretilir
echo str_pad($input, 6, "___"); // "Ayran_" üretilir
?>
```


str_repeat fonksiyonu

- Bir dizgeyi yineler.
- ```
<?php
echo str_repeat("-", 10);
?>
```
- Çıktısı:
- -----



# str\_replace fonksiyonu

- str\_replace() fonksiyonu istediğimiz karakterleri farklı bir karaktere dönüştürmemizi sağlar.
- Kullanımı:
- str\_replace(bul, değiştir, değişken)

```
<?php
$veri = "İstanbul güzel bir şehir?";
echo str_replace("güzel","pahalı",$veri);
?>
```

# strcmp() fonksiyonu

- İkili olarak dizge karşılaştırması yapar.
- `<?php`
- `$isim1="Ali";`
- `$isim2="Ahmet";`
- `if(strcmp($isim1,$isim2)==0)`
- `print "iki isim aynıdır";`
- `elseif(strcmp($isim1,$isim2)>0)`
- `print "Birinci isim büyük";`
- `else`
- `print "ikinci isim büyük";`
- `?>`

# strlen() fonksiyonu

- Dize uzunluğunu döndürür.

- ```
<?php
$str = 'abcdef';
echo strlen($str); // 6
```

```
$str = ' ab cd ';
echo strlen($str); // 7
?>
```


strtolower fonksiyonu

- Bir dizgenin harflerini küçük harfe çevirir.
- `<?php`
- `$str = "Mini MİNİ bir KUŞ uçmuştu PENCEREME konmuştu.";`
- `$str = strtolower($str);`
- `echo $str;`
- `// Çıktı: mini mini bir kuş uçmuştu pencereme konmuştu.`
- `?>`

strtoupper

- Bir dizgenin harflerini büyük harfe çevirir.
- `<?php`
- `$str = "Mini MİNİ bir KUŞ uçmuştu PENCEREME konmuştu.";`
- `$str = strtoupper($str);`
- `echo $str;`
- `// Çıktısı: MINI MINI BIR KUS UÇMUSTU PENCEREME KONMUSTU.`
- `?>`

substr() fonksiyonu

- Dizgenin bir kısmını döndürür.
- Kullanımı:
- **substr** (dizge , başlangıç [, uzunluk])

- <?php
- echo substr('abcdef', 1); // bcdef döndürür.
- echo substr('abcdef', 1, 3); // bcd döndürür.
- echo substr("abcdef", -1); // "f" döndürür.
- echo substr("abcdef", -2); // "ef" döndürür.
- echo substr("abcdef", -3, 1); // "d" döndürür.
- ?>

substr_replace() fonksiyonu

- Bir dizgenin belli bir bölümünü değiştirir.
- Kullanımı:
- **substr_replace** (dizge, yenisi , başlangıç [, uzunluk])
- ```
<? $metin = "En Faydalı Türkçe Kaynak";
```
- ```
echo substr_replace($metin, 'Yararlı', 3,7);
```
- ```
//En Yararlı Türkçe Kaynak ?>
```

# wordwrap() fonksiyonu

- Bir dizgeyi istenilen yerden alt satıra geçirir.
- Kullanımı:
- **wordwrap** ( dizge [, genişlik [, satırsonu [, kes = *false/true* ]]) )
- ```
<?php
$metin = "Dağ başını duman almış, gümüş dere durmaz akar.";
$yenisi = wordwrap($metin, 4, "<br />");
echo $yenisi;
?>
```

wordwrap(\$metin, 4, "
", **true**);
Yukarıdaki gibi sona true eklersek kelimelere dikkate almadan böler.

Php Yardımıyla mail gönderme

- **Mail() Fonksiyonu:**

PHP'de Mail göndermemize yarayan fonksiyondur. Kullanımı aşağıdaki gibidir.

mail ("mailadresini" , "mailkonusu" , "mailiçeriđi" , "Mailbaşlık")

mailadresini kısmı;mail yollanacak Kişi

mailkonusu;Mailin konusu

mailiçeriđi;Mailin içeriđi

Mailbaşlık;Mail başlığı

Php Yardımıyla mail gönderme

- Bir mailin php ile yollanması aşağıdaki gibidir.

```
< ?
```

```
$adres = "deneme@hotmail.com";
```

```
$konu = "Mail konusu";
```

```
$mesaj = "PHP Mail() Fonksiyonu";
```

```
mail("$adres", "$konu", "$mesaj");
```

```
echo "mail gönderildi";
```

```
?>
```