

PHP Nedir?

PHP, betik tabanlı bir programlama dilidir. Web tabanlı yazılım geliştirme için güçlü bir araçtır ve genellikle bu amaçla kullanılır.

Yazar: Rasmus Lerdorf
İlk Sürüm: 1995

Kaynak kodu açık ve ücretsizdir. Sunucu tarafında çalışır. HTML ile birlikte kullanılabilir. Geniş bir işletim sistemi, web sunucusu ve veritabanı yazılım desteği vardır.

Adım Adım Çalışma Süreci

1. Yazılan kod sunucu tarafında yorumlanır.
2. Çıktı HTML web sunucusuna yönlendirilir.
3. Web sunucusu çıktıyı tarayıcıya yönlendirir.

PHP ile Neler Yapılabilir?

1. Veritabanı uygulamaları
2. E-ticaret siteleri
3. Sunucu, Tartışma forumları, Arama motorları, CMS (...)

PHP Kullanımı

"echo, print, printf" komutları metni ekrana yazdırır.

Yorum Satırları:

```
<?php
// Bu bir yorum satırıdır.
echo("PHP Merhaba Dünya!");
/*
 * Test satırı 1
 * Test satırı 2
 */
echo "PHP Merhaba Dünya!";
?>
```

Kaçış Karakterleri:

```
<?php
echo ("\"PHP\" dünyası merhaba"); // çıktısı "PHP" dünyası merhaba
// \n Yeni satır, \t Sekme, \r Satır başı
?>
```

Değişken Türleri

```
<?php
    $sayi1; // Doğru
    $isim2; // Doğru
    $sayi1!; // Yanlış
    $1sayi; // Yanlış
    $isim // Doğru
    isim // Yanlış
?>
```

Veri Türleri:

Tamsayılar:

- `$number = 10;`
- `$age = 25;`

Ondalık sayılar:

- `$pi = 3.14;`
- `$price = 12.50;`

Dizeler:

- `$name = "Bard";`
- `$message = "Merhaba Dünya!";`

Boole:

- `$true = true;`
- `$false = false;`

Diziler:

- `$colors = ["kırmızı", "yeşil", "mavi"];`
- `$numbers = [1, 2, 3, 4, 5];`

Nesneler:

- `$user = new User();`
- `$car = new Car();`

Null:

- `$value = null;`

Boş:

- `$text = "";`

Değişkenleri Birleştirme

```
<?php
    $var1 = "Test1";
    $var2 = "Test2";
    $var3 = "Test3";

    $oneLineVar = $var1.$var2.$var3;
?>
```

Aritmetik Operatörler

- **Toplama (+):** İki operandı toplar. `$result = 5 + 3; // $result 8 olacak`
- **Çıkarma (-):** İkinci operandı birinciden çıkarır. `$result = 10 - 2; // $result 8 olacak`
- **Çarpma (*):** İki operandı çarpar. `$result = 4 * 5; // $result 20 olacak`
- **Bölme (/):** İlk operandı ikinciye böler. `$result = 12 / 3; // $result 4 olacak (genellikle ondalık sonuç verir)`
- **Modülüs (%):** Bölme işleminden sonra kalanı verir. `$result = 10 % 3; // $result 1 olacak`

Tanımlama

```
define('PI', 3.14159);
define('MAX_USERS', 100);
define('DEBUG_MODE', true);

echo PI; // Çıktı: 3.14159
echo MAX_USERS; // Çıktı: 100
echo DEBUG_MODE; // Çıktı: 1 (boolean true)
```

Tür Dönüşümü

```
<?php

// Tür Dönüşümü Örnekleri:

$number_string = "30"; // Bir sayı içeren dize
$age = $number_string + 5; // Toplama için otomatik dönüşüm
echo $age; // Çıktı: 35 (tamsayı)

$message = "Yaş: " . 22; // Dize birleştirme için kullanılan sayı
echo $message; // Çıktı: "Yaş: 22" (dize)

$is_admin = true;
$admin_level = $is_admin * 10; // Çarpma için boolean 1 olarak kabul edilir
echo $admin_level; // Çıktı: 10 (tamsayı)

// Açık Dönüşüm Örnekleri:
```

```
$user_age = "28";  
$actual_age = (int)$user_age; // Tamsayıya açık dönüşüm  
echo $actual_age; // Çıktı: 28 (tamsayı)  
  
$pi = 3.14;  
$pi_as_string = (string)$pi; // Dizeye açık dönüşüm  
echo $pi_as_string; // Çıktı: "3.14" (dize)  
  
?>
```

gettype() ve settype()

```
$x = 10;  
$y = "Merhaba";  
$z = true;  
  
echo gettype($x); // Çıktı: integer  
echo gettype($y); // Çıktı: string  
echo gettype($z); // Çıktı: boolean
```

```
$age = "25"; // Dize değeri  
  
// Hesaplamalar için tamsayıya dönüştür  
settype($age, "integer");  
  
$total = $age + 10;  
echo $total; // Çıktı: 35 (tamsayı)
```

isset, unset ve empty

```
<?php  
  
$name = "Alice"; // Değeri olan değişken  
$age = null; // Null değeri olan değişken  
$message = ""; // Boş dize  
$numbers = []; // Boş dizi  
  
// **isset**, varlık ve değeri kontrol eder (null dahil)  
if (isset($name)) {  
    echo "Merhaba, $name!";  
}  
  
// **empty**, çeşitli "boş" koşulları kontrol eder  
if (empty($message)) {  
    echo "Bir mesaj yok.";  
}
```

```
if (empty($numbers)) {
    echo "Dizi boş.";
}

// **isset**, $age null olduğu için yazdırmaz
if (isset($age)) {
    echo "Yaş: $age";
}

// **unset** kullanarak $age'i kaldır
unset($age);

// Kaldırılan değişkene erişmeye çalışmak hata verir
if (isset($age)) {
    echo "Yaş: $age"; // Bu bir hata oluşturur
}

?>
```

Ortam Değişkenleri

- `echo $HTTP_COOKIE_DATA;` kullanılan bir çerez hakkında bilgi gösterir.
- `echo $HTTP_USER_AGENT;`

Form aracılığıyla veri almak

```
<!DOCTYPE html>
<html lang="tr">
<head>
    <meta charset="UTF-8">
    <title>Tek Kodda $_POST ve $_GET</title>
</head>
<body>
    <h1>Bilgilerinizi Girin</h1>
    <form method="post">
        <label for="isim">Adınız:</label>
        <input type="text" name="isim" id="isim">
        <br>
        <label for="soyisim">Soyadınız:</label>
        <input type="text" name="soyisim" id="soyisim">
        <br>
        <br>
        <button type="submit">Gönder</button>
    </form>

    <?php

    // $_POST ve $_GET ile gelen verileri alalım
    $isim = $_POST["isim"] ?? $_GET["isim"];
    $soyisim = $_POST["soyisim"] ?? $_GET["soyisim"];
```

```
// Verileri ekrana yazdıralım
if ($isim && $soyisim) {
    echo "Merhaba, " . $isim . " " . $soyisim . "!";
} else {
    echo "Lütfen formdaki alanları doldurun veya URL'ye parametre ekleyin.";
}

?>
</body>
</html>
```

\$_REQUEST

```
<?php

// Form verileri (POST kullanılarak gönderilen ad ve e-posta alanları olan bir
form varsayalım)
$name = $_POST["name"];
$email = $_POST["email"];

// URL parametresi (örnek: ?message=Merhaba)
$message = $_GET["message"];

// Çerez verisi (varsayılan olarak "user_id" adında bir çerez ayarlanmış
varsayalım)
$user_id = $_COOKIE["user_id"];

// $_REQUEST tüm kaynakları birleştirir (GÜVENLİK nedenleriyle ideal olmayabilir)
$data = $_REQUEST;

// $_REQUEST'dan belirli verilere erişmek (kaynak bilinmiyorsa)
$combined_name = $_REQUEST["name"]; // Formdan veya URL'den gelebilir
?>
```

Önemli Not: Güvenlik endişeleri nedeniyle \$_REQUEST kullanırken dikkatli olun. Mümkün olduğunda \$_POST veya \$_GET gibi belirli değişkenleri kullanmayı düşünün.

MANTIKSAL KONTROL VE DÖNGÜLER

```
<?php

// Mantıksal Operatörler
$age = 20;
$has_id = true;

// && (ve): Her iki koşul da doğru
if ($age >= 18 && $has_id) {
    echo "Oy kullanabilirsiniz.";
} else {
```

```
    echo "Üzgünüz, henüz oy kullanamazsınız.";
}

// || (veya): En az bir koşul doğru
$is_admin = false;
$is_moderator = true;

if ($is_admin || $is_moderator) {
    echo "Kontrol paneline erişiminiz var.";
} else {
    echo "Erişiminiz yok.";
}

// ! (değil): Doğruluk değerini tersine çevirir
$logged_in = false;

if (! $logged_in) {
    echo "Lütfen giriş yapın.";
}

// Kontrol Akışı İfadeleri
$grade = 85;

// if/else
if ($grade >= 90) {
    echo "Harika!";
} else if ($grade >= 80) {
    echo "Harika iş!";
} else {
    echo "Çalışmaya devam edin!";
}

// switch-case
$day = "Salı";

switch ($day) {
    case "Pazartesi":
        echo "Haftanın başlangıcı!";
        break;
    case "Cuma":
        echo "Hafta sonu geldi!";
        break;
    default:
        echo "Bir hafta içi.";
}

// while döngüsü
$count = 1;
while ($count <= 3) {
    echo "Sayı: $count ";
    $count++;
}
echo "\n"; // Döngüden sonra yeni satır
```

```
// for döngüsü
for ($i = 0; $i < 5; $i++) {
    echo "Sayı: $i ";
}
echo "\n"; // Döngüden sonra yeni satır

// foreach döngüsü
$fruits = ["elma", "muz", "portakal"];

foreach ($fruits as $fruit) {
    echo "$fruit seviyorum! ";
}
echo "\n"; // Döngüden sonra yeni satır

// do-while döngüsü
$x = 10;
do {
    echo "Değer: $x ";
    $x--;
} while ($x > 0);
echo "\n"; // Döngüden sonra yeni satır
?>
```