

PHP ile İnternet Programlama

Prof.Dr. Tolga GÜYER

Gazi Üniversitesi
Gazi Eğitim Fakültesi
Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü

3. BÖLÜM: Formlarla Çalışmak

HTML Formlarının Temelleri

Dinamik etkileşim araçlarından oluşan HTML formlarından elde edilen değerlerin işlenmesi, PHP'nin en işlevsel kullanım alanlarından birisini oluşturmaktadır.

Bu kesimde HTML formları hakkında temel seviyede bilgi verilmesi amaçlanmıştır. Bu bilgiler, ilerleyen bölümlerde ele alacağımız MySQL uygulamaları konusuna da temel oluşturacaktır.

İnternet üzerinde çalışan veritabanı uygulamalarında formlar, kullanıcının doğrudan etkileşim kurduğu en üst katmanda yer alırlar. Dolayısıyla kullanıcı-dostu olarak tasarlanmaları gerekir.

HTML Formlarının Temelleri

HTML formları, üzerlerinde bulunan metin alanları, radyo düğmeleri, işaretleme kutuları ya da liste kutuları gibi çeşitli etkileşim nesneleri yoluyla kullanıcıdan bilgi alınmasını sağlayan araçlardır.

Bu formlar,

```
<form>
```

```
...
```

```
...
```

```
</form>
```

etiketleri arasında oluşturulur.

HTML Formlarının Temelleri

Bu durumda metin tabanlı bir verinin girilmesini sağlayan form, aşağıdaki gibi olacaktır:

```
<form>  
  <p>Adınız: <input type="text" name="ad" ></p>  
  <p>Soyadınız: <input type="text" name="soyad" ></p>  
</form>
```

Burada `<input type="text" name="ad">` bildirimini, alanın bir girdi (input) alanı, türünün ise metin kutusu (text) olduğunu söylemektedir.

“name” özelliği ise, nesnenin programatik adını göstermektedir.

HTML Formlarının Temelleri

Tasarladığımız bu formu, bilgileri gönderme işlemini sağlayacak bir komut düğmesi ile tamamlayalım:

```
<form>
```

```
  <p>Adınız: <input type="text" name="ad" ></p>
```

```
  <p>Soyadınız: <input type="text" name="soyad" ></p>
```

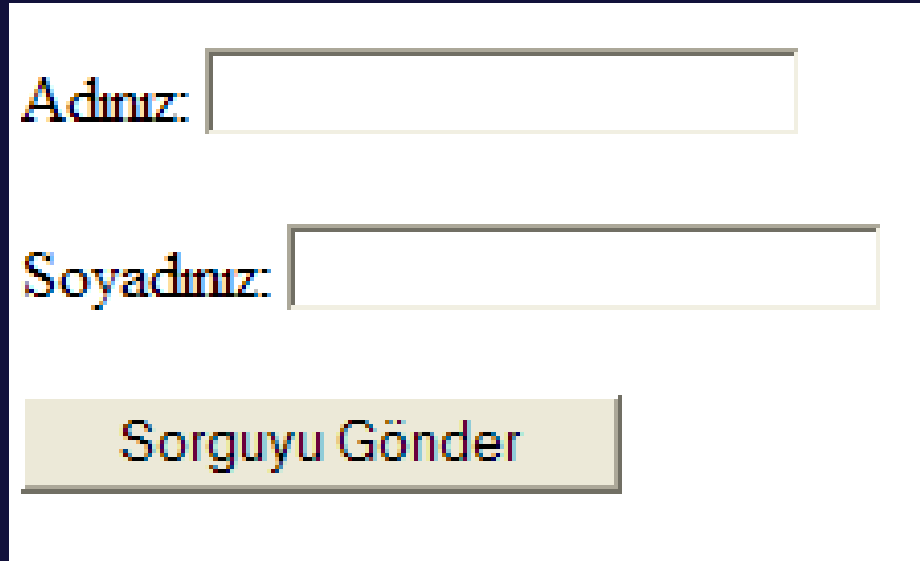
```
  <p><input type="submit" ></p>
```

```
</form>
```

Burada `<input type="submit">` bildirimini, yine alanın bir girdi (input) alanı, ancak türünün gönderme butonu (submit) olduğunu söylemektedir.

HTML Formlarının Temelleri

Bu koda ait ekran görüntüsü aşağıdaki gibi olacaktır:



The screenshot shows a simple web form on a white background. It consists of two text input fields and a submit button. The first input field is labeled "Adınız:" and the second is labeled "Soyadınız:". Below these fields is a button with the text "Sorguyu Gönder".

Adınız:

Soyadınız:

HTML Formlarının Temelleri

Sıkça kullanılan bazı nesnelerin yer aldığı örnek bir HTML formu aşağıdaki resimde görülmektedir.

Bu bir textarea kontrolüdür.
Çok satırlı metin yazmaya uygundur.

Seçenek-1

Seçenek-2

Seçenek-3

Seçim-1

Seçim-2

Seçim-3

seçim-1

seçim-1
seçim-2
seçim-3

HTML Formlarının Temelleri

Bu forma ait HTML kodları ise aşağıda verilmiştir.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/>
<title>HTML Form Örneği</title>
</head>

<body>
<form id="form1" name="form1" method="post" action="">
  <p>
    <textarea name="textarea" id="textarea" cols="45"
rows="5"></textarea>
  </p>
  <p>
    <input type="checkbox" name="checkbox1"
id="checkbox">Seçenek-1
  </p>
  <p>
    <input type="checkbox" name="checkbox2"
id="checkbox2">Seçenek-2
  </p>
```


HTML Formlarının Temelleri

```
<p>
    <input type="checkbox" name="checkbox3"
id="checkbox3">Seçenek-3
  </p>
  <p>
    <input name="radio" type="radio" id="radio" value="1"
checked="checked" />Seçim-1
  </p>
  <p>
    <input type="radio" name="radio" id="radio" value="2"
/>Seçim-2
  </p>
  <p>
    <input type="radio" name="radio" id="radio" value="3"
/>Seçim-3
  </p>
  <p>
    <select name="select" id="select">
      <option value="seçim-1">seçim-1
      <option value="seçim-2">seçim-2
      <option value="seçim-3">seçim-3
    </select>
  </p></form></body></html>
```

HTML Formlarının Temelleri

İlk alan bir Textarea'dır. Bunun metin kutusundan farkı, çok satırlı olarak daha uzun metinleri alabilen bir nesne olmasıdır.

HTML formlarında işaretleme yapma amacı ile iki tip nesne kullanılır: Checkbox ve Radiobutton. Checkbox nesnesi çoklu seçimlerin yapılmasına olanak tanır. Tek seçim yapılmasını istediğimiz durumlarda ise Radiobutton nesnesini kullanırız.

Farklı Radiobutton nesnelerinin aynı grup içerisinde değerlendirildiği bu durumu, her nesneye aynı id (kimlik) adının verilmesi sağlamaktadır.

En son nesnemiz ise üç seçenek içeren Combobox türünde bir Select nesnesidir.

PHP Sayfasına Veri Gönderme

Gönderme butonuna basıldığında gerçekleşecek eylemin belirlenmesi için form etiketinin action özelliği kullanılır. Buraya yazılacak veri, herhangi bir script kodunun çalıştırılması ile ilgili olabileceği gibi, doğrudan bir php sayfasının adresi de yazılabilir.

```
<form action="verileri_al.php" method="post" >  
  <p>Adınız: <input type="text" name="ad" ></p>  
  <p>Soyadınız: <input type="text" name="soyad" ></p>  
  <p><input type="submit" ></p>  
</form>
```

PHP Sayfasına Veri Gönderme

Burada gönderme butonuna basıldığı takdirde, internet tarayıcı tarafından verileri_al.php dosyası yüklenecektir.

Bu değer boş bırakıldığı takdirde, yani,

```
<form action=" " method="post" >  
  <p>Adınız: <input type="text" name="ad" ></p>  
  <p>Soyadınız: <input type="text" name="soyad" ></p>  
  <p><input type="submit" ></p>  
</form>
```

durumunda, gönderme butonu mevcut sayfanın yeniden yüklenmesini sağlayacaktır. Bu durumda çalışmasını istediğimiz php kodlarının da aynı sayfa üzerinde olması gerekmektedir.

PHP Sayfasına Veri Gönderme

Bilgilerin form üzerinden gönderilmesi için iki farklı yöntemden birisi seçilebilir. Bunlardan ilki “post” , diğeri ise “get” yöntemidir. Gönderme yöntemi, form etiketinin method özelliği kullanılarak belirlenir.

Bilgilerin alınış yöntemi de, gönderme için seçilen yönteme göre değişecektir.

Öncelikle “post” metodunu inceleyelim.

PHP Sayfasına Veri Gönderme

POST Metodu

Aşağıdaki HTML formunu göz önüne alalım:

```
<html>
<body>
<form action="verileri_al.php" method="post">
  <p>Adınız: <input type="text" name="ad"></p>
  <p>Soyadınız: <input type="text" name="soyad" >
</p>
  <p><input type="submit"></p>
</form>
</body>
</html>
```

PHP Sayfasına Veri Gönderme

POST Metodu

Bu kodu taşıyan dosyayı, “verileri_gonder.html” adıyla kaydedelim. Şimdi, action özelliğinde belirtilen “verileri_al.php” dosyasının içereceği kodları yazalım:

```
<?php
    $kullaniciAdi = $_POST['ad'];
    $kullaniciSoyadi = $_POST['soyad'];
    echo "Merhaba $kullaniciAdi $kullaniciSoyadi";
?>
```

PHP Sayfasına Veri Gönderme

POST Metodu

İnternet tarayıcısı tarafından görüntülenecek dosya, “verileri_gonder.html” adıyla kaydettiğimiz form dosyası olacaktır. Bu form üzerinde yer alan veri gönderme butonuna basıldığında internet tarayıcısı tarafından “verileri_al.php” dosyası yüklenecek ve PHP çözümleyicisi tarafından çalıştırılacaktır.

Sonuç olarak ekranda aşağıdaki gibi bir görüntü olacaktır:

Merhaba Tolga Güyer

PHP Sayfasına Veri Gönderme

POST Metodu

Kullanıcı tarafından verileri_gonder.html dosyasında girilen bilgiler, post metodu kullanılarak verileri_al.php dosyasına gönderilmiştir ve kullanıcı gönderilen verileri ancak bu PHP kodları çalıştırıldığında görebilmiştir. Veriler, \$_POST fonksiyonu kullanılarak alınmıştır. Parametre olarak da form üzerindeki alanın adı kullanılmıştır.

Get metodunda ise durum biraz farklıdır.

PHP Sayfasına Veri Gönderme

GET Metodu

Aynı örneği bu defa defa get metodunu kullanarak yapalım:

```
<html>
<body>
<form action="verileri_al.php" method="get">
  <p>Adınız: <input type="text" name="ad"></p>
  <p>Soyadınız: <input type="text"
name="soyad"></p>
  <p><input type="submit"></p>
</form>
</body>
</html>
```

PHP Sayfasına Veri Gönderme

GET Metodu

Bu durumda HTML dosyasında değişen kısım form etiketinin method özelliğinin değeri olmuştur. PHP dosyasında ise verilerin alınmasında kullanılan `$_POST` fonksiyonu yerine `$_GET` fonksiyonu kullanılmıştır.

```
<?php
    $kullaniciAdi = $_GET['ad'];
    $kullaniciSoyadi = $_GET['soyad'];
    echo "Merhaba $kullaniciAdi $kullaniciSoyadi";
?>
```

PHP Sayfasına Veri Gönderme

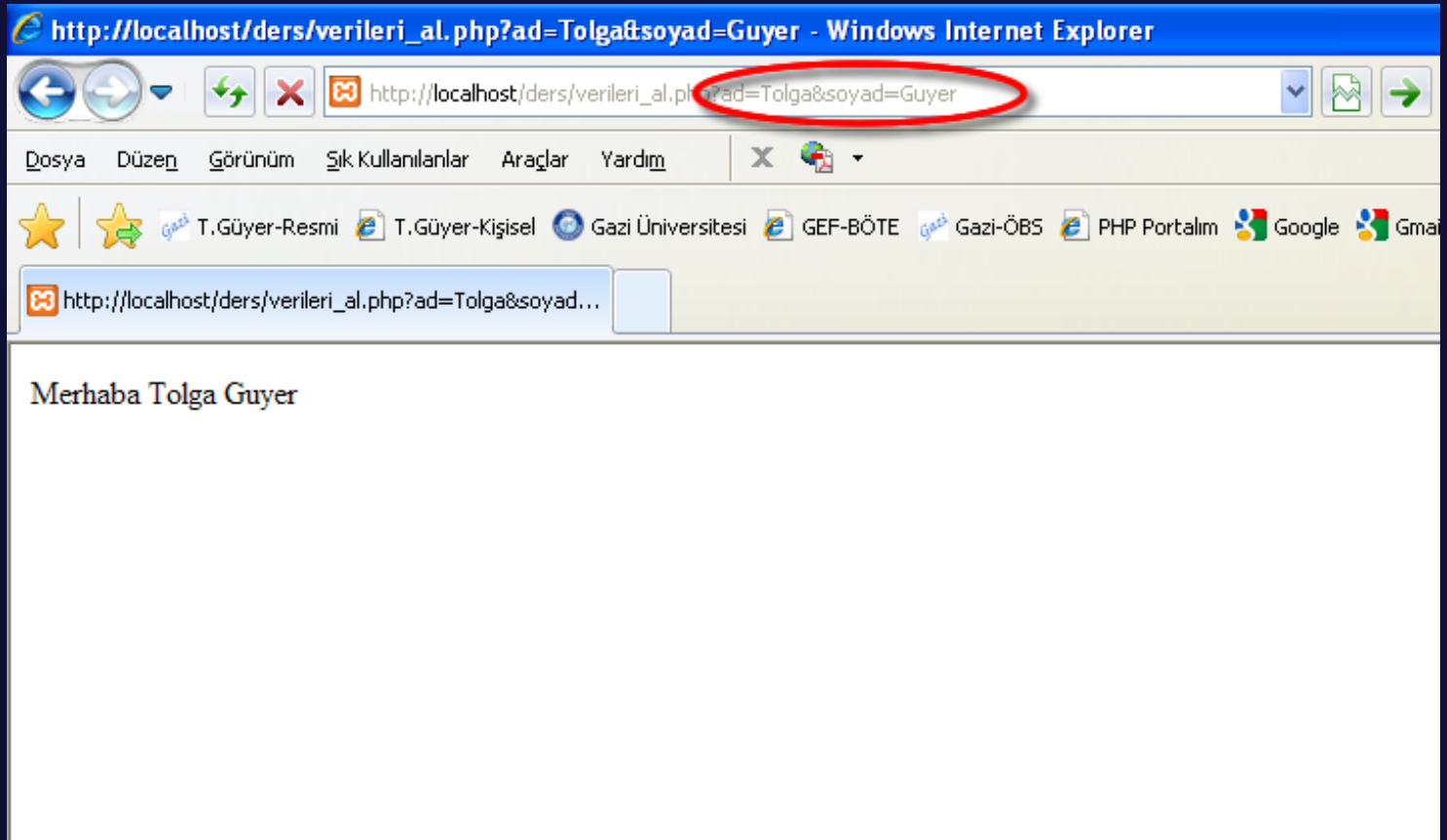
GET Metodu

Aynı amaçla kullanılan bu iki yöntem arasındaki en önemli fark, get metodunda verilerin gönderiliş biçiminden kaynaklanmaktadır. Bu durumda veriler, doğrudan internet tarayıcısının adres çubuğu kullanılarak aktarılırlar.

HTML formu doldurulup gönderme butonuna basıldığı anda, internet tarayıcısında resimdeki gibi bir durum oluşacaktır.

PHP Sayfasına Veri Gönderme

GET Metodu



PHP Sayfasına Veri Gönderme

GET Metodu

Resimde de görüldüğü gibi internet tarayıcısının adres çubuğunda yer alan

`http://.../verileri_al.php?ad=Tolga&soyad=Guyer`

adres bilgisi, gönderilen parametreleri de içermektedir.

Dolayısıyla gizlilik içeren bilgilerin get metodu kullanılarak gönderilmesi önerilmez. Bu şekilde gönderilecekse, verilerin şifrelenmesi ve kullanılmadan önce şifrelerinin çözülmesi bu güvenlik problemini ortadan kaldıracaktır.

PHP Sayfasına Veri Gönderme

GET Metodu

Ayrıca parametre sayısı çok fazla olduğunda adres bilgisi çok uzun olacağından ve bu bilgi için 1 Kb üst sınırı bulunduğundan, bu durumda post metodu tercih edilmelidir.

Diğer yandan post metodu kullanıldığında bilgiler HTTP tarafından paketlenerek gönderildiğinden, get metoduna oranla daha yavaş bir transfer olacağı söylenebilir.

PHP Sayfasına Veri Gnderme

Aynı Sayfa zerinden Ykleme Yapmak

Aynı rneęi tek dosya zerinde alıřacak řekilde tasarlayalım. Bu durumda hem HTML formu, hem de PHP kodu aynı dosya zerinde olacaktır ve gnderme butonuna basıldıęında bu dosya tekrar yklenecektir.

PHP Sayfasına Veri Gönderme

Aynı Sayfa Üzerinden Yükleme Yapmak

```
<html>
<body>
<form action="" method="post">
<p>Adınız: <input type="text" name="ad"></p>
<p>Soyadınız: <input type="text" name="soyad"></p>
<p><input type="submit" name="dugme"></p>
</form>

<?php
    if(isset($_POST["dugme"])) {
        $kullaniciAdi = $_POST['ad'];
        $kullaniciSoyadi = $_POST['soyad'];
        echo "Merhaba $kullaniciAdi $kullaniciSoyadi"; }
?>
</body>
</html>
```

PHP Sayfasına Veri Gönderme

Aynı Sayfa Üzerinden Yükleme Yapmak

Örneğin bu biçimde, aynı sayfa tekrar yüklendiğinden, ilk yüklenişinde çalışmasını istemediğimiz bölüm bir koşul ile kontrol altına alınmıştır. Şöyle ki, bu koşulda yer alan “isset” fonksiyonu, “dugme” adlı butona basıldığında sağlanmaktadır. Dolayısıyla sayfanın ilk yüklenişinde kırmızı renkli kod otomatik olarak çalışmayacaktır.

Örnek Uygulamalar

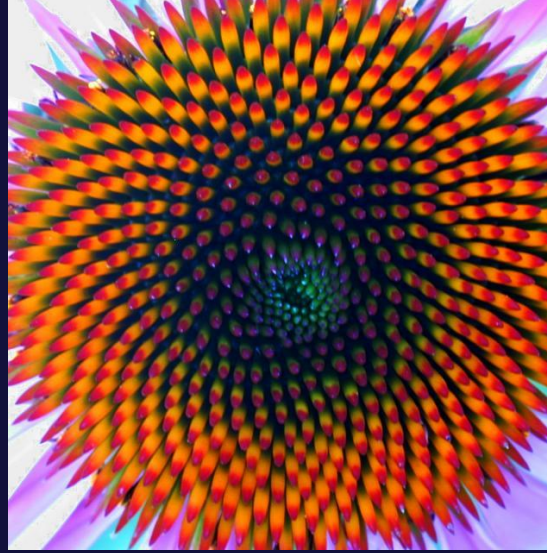
Örnek-1: Fibonacci Dizisi

Fibonacci dizisi, 1,1,2,3,5,8,13,21,... biçiminde, 1,1 sayıları ile başlayıp her sayının kendisinden önceki iki sayının toplamı olduğu sayı dizisidir. Leonardo Fibonacci (1170-1250) tarafından bir tavşan çiftliğinde her üreme döneminde elde edilecek toplam tavşan sayısını bulmak için geliştirilen dizi, bulunduğundan bu yana doğada daha pek çok yerde insanların karşısına çıkmıştır.

Örneğin; dizideki her bir sayı, Fibonacci sayısı adını alır ve çiçeklerin helezoni biçimindeki tohum dağılımlarında, her bir helezonda yer alan tohum sayısı bir Fibonacci sayısına karşılık gelmektedir. Sonraki slaytta yer alan resim bu durumu göstermektedir.

Örnek Uygulamalar

Örnek-1: Fibonacci Dizisi



Fibonacci dizisinin bir diğer ilginç özelliği de, ardışık iki Fibonacci sayısının oranının, dizi ilerledikçe, sanat ve mimaride binlerce yıldır bilinen ve “altın oran” adı verilen sayıya yakınsamasıdır.

Örnek Uygulamalar

Örnek-1: Fibonacci Dizisi

Yazacağımız program kullanıcı tarafından girilecek bir adım sayısına göre Fibonacci dizisini ve altın orana yakınsama özelliğini listelemektedir.

Öncelikle kullanıcıdan adım sayısını alacağımız HTML formunu tasarlayalım.

```
<form action="fibonacci.php" method="post">
  <p>Adım sayısını giriniz: <input type="text"
name="adim" ></p>
  <p><input type="submit" value="DİZİYİ OLUŞTUR"></p>
</form>
```

Örnek Uygulamalar

Örnek-1: Fibonacci Dizisi

fibonacci.php dosyasının içeriği ise aşağıdaki gibi olacaktır.

```
<?php
$adim=$_POST['adim'];
$a=1;
$b=1;
echo "$a<br>$b<br>";
for ($sayac=1;$sayac<$adim-1;$sayac++)
{
    $c = $a + $b;
    echo "$c - ".$c/$b. "<br>";
    $a=$b;
    $b=$c;
}
?>
```

Örnek Uygulamalar

Örnek-2: Algoritmik Karekök Alma

Girilecek bir sayının karekökünü, belirlenecek bir adım sayısına göre;

$$A_0 = N \text{ (Adım Sayısı)}$$

$$A_k = \frac{A_{k-1} + \frac{N}{A_{k-1}}}{2}$$

yaklaşımını kullanarak algoritmik olarak hesaplayacak programı yazalım.

Örnek Uygulamalar

Örnek-2: Algoritmik Karekök Alma

```
<form action="" method="post">
  <p><b>Karekökü alınacak sayıyı giriniz      : <input type="text"
name="sayi" size="4"></b></p>
  <p><b>Adım sayısını giriniz              : <input type="text"
name="adim" size="4"></b></p>
  <p><input name="dugme" type="submit" value="BAŞLAT!"></p>
</form>
<?php
  if(isset($_POST["dugme"]))
  {
    $N=$_POST['sayi'];
    $a=$N;
    $adim_sayisi=$_POST['adim'];
    while ($say<=$adim_sayisi)
    {
      $a = ($a + $N/$a)/2;
      $say++;
    }
    echo "Karekökü = $a";
  }
?>
```


Örnek Uygulamalar

Örnek-3: Dört İşlem

Bu örneğimizde, form üzerinden iki sayı ve işlem kodu alınarak sonuç bir kullanıcı-tanımlı fonksiyon kullanılarak hesaplanıp ana programa geri gönderilmektedir.

```
<form action="" method="post">
  <p>Birinci Sayı : <input type="text" name="sayi_1" size="4"></p>
  <p>İkinci Sayı : <input type="text" name="sayi_2" size="4"></p>
  <p>İşlem Kodu : <select size="1" name="kod">
    <option value="-1" selected="selected">== S E Ç İ N İ Z
  ==</option>
  <option value="1">Toplama</option>
    <option value="2">Çıkarma</option>
    <option value="3">Bölme</option>
    <option value="4">Çarpma</option>
  </select>
  </p>
  <p><input name="dugme" type="submit" value="Hesapla"></p>
</form>
```

Örnek Uygulamalar

Örnek-3: Dört İşlem

```
<?php
if(isset($_POST["dugme"])){
    function islemYap($sayi1,$sayi2,$islemKodu) {
        switch ($islemKodu) {
            case 1:
                $sonuc=$sayi1+$sayi2;
                break;
            case 2:
                $sonuc=$sayi1-$sayi2;
                break;
            case 3:
                $sonuc=$sayi1/$sayi2;
                break;
            case 4:
                $sonuc=$sayi1*$sayi2;
                break; }
        return $sonuc; }
    $s1=(int)$_POST['sayi_1'];
    $s2=(int)$_POST['sayi_2'];
    $kod=$_POST['kod'];
    echo islemYap($s1,$s2,$kod);
}
?>
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

Geliştireceğimiz bu uygulama, İnternet üzerinde bilgisayar parçaları siparişi vermek için tasarlanmıştır.

Uygulama iki dosyadan oluşmaktadır. İlki, sipariş formunun yer aldığı HTML uygulaması, ikincisi ise sipariş bilgilerini alarak hesaplamaları ve raporlamayı gerçekleştiren PHP uygulamasıdır.

Öncelikle HTML uygulamasını verelim.

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<html>
<head>
  <title>Bilgisayar Parçaları Satışı</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-9"></head>
<body>
<h1>Bilgisayar Parçaları Satışı</h1>
<h2>Sipariş Formu</h2>
<form action="siparisi_al.php" method=post>
<table border=1>
<tr bgcolor=#BEE9E9>
  <td width=150>Parça Adı</td>
  <td align=center width=110>Birim Fiyat</td>
  <td width=15>Adet</td>
</tr>
<tr bgcolor=#FFFFAA>
  <td>Kasa</td>
  <td align=right>85.50 TL</td>
  <td align=center><input type="text" name="kasa" size=3 maxlength=3></td>
</tr>
<tr>
  <td>Anakart</td>
  <td align=right>150.00 YTL</td>
  <td align=center><input type="text" name="anakart" size=3 maxlength=3></td>
</tr>
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<tr bgcolor=#FFFFFFAA>
  <td>İşlemci</td>
  <td align=right>275.30 YTL</td>
  <td align=center><input type="text" name="islemci" size=3 maxlength=3></td>
</tr>
<tr bgcolor=#FFFFFFAA>
  <td>Sabit Disk</td>
  <td align=right>120.80 YTL</td>
  <td align=center><input type="text" name="sabitdisk" size=3 maxlength=3></td>
</tr>
<tr bgcolor=#FFFFFFAA>
  <td>CD/DVD Sürücüsü</td>
  <td align=right>90.00 YTL</td>
  <td align=center><input type="text" name="cddvd" size=3 maxlength=3></td>
</tr>
<tr bgcolor=#FFFFFFAA>
  <td>Bellek</td>
  <td align=right>134.20 YTL</td>
  <td align=center><input type="text" name="bellek" size=3 maxlength=3></td>
</tr>
<tr bgcolor=#BEE9E9>
  <td colspan=3 align=left><input type=submit value="Siparişi Al"></td>
</tr>
</table> </form> </body> </html>
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

Form çalıştırıldığında devreye siparisi_al.php dosyası girecektir. Bu dosya birim fiyat sabitlerini ve formdan gelen parça adetleri verisini kullanarak toplam tutarı hesaplayacak, string fonksiyonlarının kullanıldığı bir yöntemle bu tutarı yazı ile de yazacaktır.

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<html>
<head> <title> Bilgisayar Parçaları Satışı</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-9"></head>
<body>
<h1>Bilgisayar Parçaları Satışı</h1>
<h2>Sipariş Bilgileri</h2>
<?php
function birler_oku ($Gelen_Sayi){
switch ($Gelen_Sayi) {
case 0:
    $bir="";
    break;
case 1:
    $bir="BİR";
    break;
case 2:
    $bir="İKİ";
    break;
case 3:
    $bir="ÜÇ";
    break;
case 4:
    $bir="DÖRT";
    break;
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
case 5:
    $bir="BEŞ";
    break;
case 6:
    $bir="ALTI";
    break;
case 7:
    $bir="YEDİ";
    break;
case 8:
    $bir="SEKİZ";
    break;
case 9:
    $bir="DOKUZ";
    break;
}
return $bir;
}
```


Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
function onlar_oku ($Gelen_Sayi)
{
switch ($Gelen_Sayi) {
case 0:
    $on="";
    break;
case 1:
    $on="ON";
    break;
case 2:
    $on="YİRMİ";
    break;
case 3:
    $on="OTUZ";
    break;
case 4:
    $on="KIRK";
    break;
case 5:
    $on="ELLİ";
    break;
case 6:
    $on="ALTMİŞ";
    break;
}
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
case 7:
    $on="YETMİŞ";
    break;
case 8:
    $on="SEKSEN";
    break;
case 9:
    $on="DOKSAN";
    break;}
return $on;}
function oku ($sayi){
$basamak_sayisi=strlen($sayi);
    switch ($basamak_sayisi) {
case 1:
        $sonuc=birler_oku((int)$sayi);
        break;
case 2:
        $sonuc=onlar_oku((int)$sayi[0]).birler_oku((int)$sayi[1]);
        break;
case 3:
        if ($sayi[0]=="1")
        {
            $sonuc="YÜZ".onlar_oku((int)$sayi[1]).birler_oku((int)$sayi[2]);
        }
    }
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
else
{

    $sonuc=birler_oku((int)$sayi[0])."YÜZ".onlar_oku((int)$sayi[1]).birler_oku((int)$sayi[2]);
}
break;
case 4:
    if ($sayi[0]=="1")
    {
        if ($sayi[1]=="0" and $sayi[2]=="0")
        {
            $sonuc="BİN".birler_oku((int)$sayi[3]);
        }
        elseif ($sayi[1]=="0")
        {
            $sonuc="BİN".onlar_oku((int)$sayi[2]).birler_oku((int)$sayi[3]);
        }
        else
        {
            if ($sayi[1]=="1")
            {
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
$sonuc="BİNYÜZ".onlar_oku((int)$sayi[2]).birler_oku((int)$sayi[3]);
    }
    else
    {

$sonuc="BİN".birler_oku((int)$sayi[1])."YÜZ".onlar_oku((int)$sayi[2]).birler_oku(
(int)$sayi[3]);
    }
    }
    else
    {
        if ($sayi[1]=="0" and $sayi[2]=="0")
        {
            $sonuc=birler_oku((int)$sayi[0])."BİN".birler_oku((int)$sayi[3]);
        }
        elseif ($sayi[1]=="0")
        {

            $sonuc=birler_oku((int)$sayi[0])."BİN".onlar_oku((int)$sayi[2]).birler_oku((in
t)$sayi[3]);
        }
        else
        {
            if ($sayi[1]=="1") {
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
$sonuc=birler_oku((int)$sayi[0])."BİNYÜZ".onlar_oku((int)$sayi[2]).birler_oku(
(int)$sayi[3]);
    }
    else {

$sonuc=birler_oku((int)$sayi[0])."BİN".birler_oku((int)$sayi[1])."YÜZ".onlar_oku(
(int)$sayi[2]).birler_oku((int)$sayi[3]);
    }
    }

    break;
}
return $sonuc;
}

function TL_Oku ($para)
{
    $noktayeri=strpos($para,".");
    if ($noktayeri<>0)
    {
        $tl=substr($para,0,$noktayeri);
        $kr=substr($para,$noktayeri+1,strlen($para)-$noktayeri);
        return oku($tl)." TL  ".oku($kr)." Kr";  }
}
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
else
{
    return oku($para)." TL";
}
}
echo "<p>Siparişiniz "; // Siparişle ilgili bütün detaylar bu php kodu
                        // tarafından hesaplanmaktadır.

echo date("j/m/y");
echo " tarihinde, saat ";
echo date("H:i");
echo " itibarı ile işleme konmuştur.";
echo "<p>Siparişinizle ilgili detaylar aşağıda verilmiştir:";
echo "<p>";
$kasa = (int)$_POST['kasa'];
$anakart = (int)$_POST['anakart'];
$islemci = (int)$_POST['islemci'];
$sabitdisk = (int)$_POST['sabitdisk'];
$cddvd = (int)$_POST['cddvd'];
$bellek = (int)$_POST['bellek'];
$kasa_ucreti = (double)$_POST['kasabirim'];
$anakart_ucreti = (double)$_POST['anakartbirim'];
$islemci_ucreti = (double)$_POST['islemcibirim'];
$sabitdisk_ucreti = (double)$_POST['sabitdiskbirim'];
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
$cddvd_ucreti = (double)$_POST['cddvdbirim'];
$bellek_ucreti = (double)$_POST['bellekbirim'];
define("kasa_ucreti", 85.50);
define("anakart_ucreti", 150.0);
define("islemci_ucreti", 275.30);
define("sabitdisk_ucreti", 120.80);
define("cddvd_ucreti", 90.0);
define("bellek_ucreti", 134.20);
define("kdv_orani", 15);

$toplam_parca = 0;
$toplam_tutar = 0.00;

$toplam_parca = $kasa + $anakart + $islemci + $sabitdisk + $cddvd + $bellek;
$toplam_birim_fiyat = kasa_ucreti + anakart_ucreti + islemci_ucreti
                    + sabitdisk_ucreti + cddvd_ucreti + bellek_ucreti;
$toplam_tutar = $kasa * kasa_ucreti
               + $anakart * anakart_ucreti
               + $islemci * islemci_ucreti
               + $sabitdisk * sabitdisk_ucreti
               + $cddvd * cddvd_ucreti
               + $bellek * bellek_ucreti;

?>
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<table border=1>
  <tr bgcolor=#C399C3>
    <td width=150>Parça Adý</td>
    <td width=25>Adet</td>
    <td width=110 align=center>Birim Fiyat</td>
    <td width=110 align=center>Tutar</td>
    <td width=110 align=center>KDV (%<?php echo kdv_orani ?>)</td>
    <td width=110 align=center>KDV Dahil Tutar</td>
  </tr>
  <tr bgcolor=#D4FFD4>
    <td>Kasa</td>
    <td align=right><?php echo $kasa ?></td>
    <td align=right> <?php echo number_format(kasa_ucreti,2)." TL" ?></td>
    <td align=right> <?php echo number_format($kasa * kasa_ucreti,2)." TL"
?></td>
    <td align=right> <?php echo number_format($kasa * kasa_ucreti *
$kdv_orani,2)." TL" ?></td>
    <td align=right> <?php echo number_format($kasa * kasa_ucreti * (1 +
$kdv_orani),2)." TL" ?></td></tr>
  <tr bgcolor=#D4FFD4>
    <td>Anakart</td>
    <td align=right><?php echo $anakart ?></td>
    <td align=right>
```


Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<?php echo number_format(anakart_ucreti,2). " TL" ?></td>
    <td align=right> <?php echo number_format($anakart * anakart_ucreti,2). "
TL" ?></td>
    <td align=right> <?php echo number_format($anakart * anakart_ucreti *
kdv_orani,2). " TL" ?></td>
    <td align=right> <?php echo number_format($anakart * anakart_ucreti * (1 +
kdv_orani),2). " TL" ?></td></tr>
    <tr bgcolor=#D4FFD4>
        <td>Ýþlemci</td>
        <td align=right><?php echo $islemci ?></td>
        <td align=right> <?php echo number_format(islemci_ucreti,2). " TL" ?></td>
        <td align=right> <?php echo number_format($islemci * islemci_ucreti,2). "
TL" ?></td>
        <td align=right> <?php echo number_format($islemci * islemci_ucreti *
kdv_orani,2). " TL" ?></td>
        <td align=right> <?php echo number_format($islemci * islemci_ucreti * (1 +
kdv_orani),2). " TL" ?></td></tr>
    <tr bgcolor=#D4FFD4>
        <td>Sabit Disk</td>
        <td align=right><?php echo $sabitdisk ?></td>
        <td align=right> <?php echo number_format(sabitdisk_ucreti,2). " TL" ?></td>
        <td align=right> <?php echo number_format($sabitdisk *
sabitdisk_ucreti,2). " TL" ?></td>
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<td align=right> <?php echo number_format($sabitdisk * sabitdisk_ucreti *
kdv_orani,2)." TL" ?></td>
    <td align=right> <?php echo number_format($sabitdisk * sabitdisk_ucreti *
(1 + kdv_orani),2)." TL" ?></td>
</tr>
<tr bgcolor=#D4FFD4>
    <td>CD/DVD Sürücüsü</td>
    <td align=right><?php echo $cddvd ?></td>
    <td align=right> <?php echo number_format(cddvd_ucreti,2)." TL" ?></td>
    <td align=right> <?php echo number_format($cddvd * cddvd_ucreti,2)." TL"
?></td>
    <td align=right> <?php echo number_format($cddvd * cddvd_ucreti *
kdv_orani,2)." TL" ?></td>
    <td align=right> <?php echo number_format($cddvd * cddvd_ucreti * (1 +
kdv_orani),2)." TL" ?></td>
</tr>
<tr bgcolor=#D4FFD4>
    <td>Bellek</td>
    <td align=right><?php echo $bellek ?></td>
    <td align=right> <?php echo number_format(bellek_ucreti,2)." TL" ?></td>
    <td align=right> <?php echo number_format($bellek * bellek_ucreti,2)." TL"
?></td>
```

Örnek Uygulamalar

Örnek-4: Bilgisayar Parçaları Satışı

```
<td align=right> <?php echo number_format($bellek * bellek_ucreti *  
kdv_orani,2)." TL" ?></td>  
    <td align=right> <?php echo number_format($bellek * bellek_ucreti * (1 +  
kdv_orani),2)." TL" ?></td></tr>  
    <tr bgcolor=#FFFFFFAA>  
        <td>TOPLAM</td>  
        <td align=right><?php echo $toplam_parca ?></td>  
        <td align=right> <?php echo number_format($toplam_birim_fiyat,2)." TL"  
?></td>  
        <td align=right> <?php echo number_format($toplam_tutar,2)." TL" ?></td>  
        <td align=right> <?php echo number_format($toplam_tutar * kdv_orani,2)." TL" ?></td>  
        <td align=right> <?php echo number_format($toplam_tutar * (1 +  
kdv_orani),2)." TL" ?></td></tr></table>  
<?php  
    echo "<p>YAZI İLE : "; echo TL_Oku((string)(number_format($toplam_tutar * (1 +  
kdv_orani),2)));  
    echo "<p>TEŞEKKÜRLER..."; ?>  
</body>  
</html>
```