

### Formlar

Kimi zaman web sitenizi ziyaret eden kullanıcılardan bilgi almanız gerekebilir. Bu bilgiler içinde siteyle ilgili düşünceler, kullanıcı kayıt verileri, anket cevapları ve ürün satın alma bilgileri yer alabilir. Farklı tipte bilgiler toplamaktan ziyaretçilerin siteyle etkileşime girmesini sağlamaya kadar pek çok işleve sahip olan formlar, veri elde etmenizi sağlayan kullanıcı arabirimleridir.

Formlar alan adı verilen ve ziyaretçinin bilgi girdiği bazı bileşenler içerir. Bu alanların arasında metin alanları, radyo düğmeleri, onay kutuları, menüleri ya da listeleri ve benzerleri yer almaktadır.

### Formun Oluşturulması

Bir sayfaya alanlar ve düğmeler gibi elemanlar eklemeyen önce bu elemanları içerecek olan formu oluşturmak gerekir. Form; alanlar, düğmeler, menüler ve ziyaretçilerin bilgi girmek ya da seçim yapmak için kullandığı diğer nesnelere için bir taşıyıcı görevi üstlenir. Formlar ayrıca gönderildiklerinde verilere ne olacağını belirler. Form nesnelere için çalışabilmesi için mutlaka bir form ile çevreli olmaları gereklidir. Bunun için sayfaya ilk olarak form etiketini ya da çevreleyicisini eklemelisiniz.

```
<form>
  Form Elemanları
</form>
```

Bir sayfaya birden fazla form yerleştirebilirsiniz. Ancak HTML’de bir formu diğer bir formun içine yerleştiremezsiniz.

### Form Elemanları

Eleman	İşlev
<b>&lt;form&gt;</b>	Kullanıcının veri girişi yapabileceği bir form alanı tanımlar
<b>&lt;input&gt;</b>	Veri girişi için bir kontrolcü tanımlar
<b>&lt;textarea&gt;</b>	Veri girişi için çok satırlı bir kontrolcü tanımlar
<b>&lt;label&gt;</b>	<input> elemanı için bir etiket tanımlar.
<b>&lt;fieldset&gt;</b>	Bir formdaki ilgili elemanları gruplandırır
<b>&lt;legend&gt;</b>	<fieldset> elemanı için bir resim yazısı tanımlar
<b>&lt;select&gt;</b>	Bir açılır liste tanımlar

Eleman	İşlev
<b>&lt;optgroup&gt;</b>	Açılır listedeki ilişkili seçenekleri gruplandırır
<b>&lt;option&gt;</b>	Açılır liste için bir seçenek tanımlar
<b>&lt;button&gt;</b>	Tıklanabilir bir buton tanımlar
<b>&lt;datalist&gt;</b>	Önceden tanımlanmış seçeneklerin listesini belirtir
<b>&lt;output&gt;</b>	Bir hesaplamanın sonucunu tanımlar

### Veri Giriş Türleri

Tür	İşlev
<b>text</b>	Bir satırlık metin giriş alanı tanımlar
<b>password</b>	Şifre alanı tanımlar
<b>submit</b>	Form verilerini göndermek için bir buton tanımlar
<b>textarea</b>	Veri girişi için çok satırlı bir kontrolcü tanımlar
<b>radio</b>	Seçeneklerden sadece birinin seçilmesine olacak sağlayan seçim listesi
<b>checkbox</b>	Seçeneklerin hiçbirini ya da birçoğunu seçmeye olanak sağlayan seçim listesi
<b>url</b>	Geçerli bir e-web adresi girilmesini sağlar
<b>email</b>	Geçerli bir e-posta adresi girilmesini sağlar
<b>file</b>	Dosya yüklemeleri için bir "Gözet" düğmesi tanımlar
<b>number</b>	Sayısal bir giriş alanı tanımlar
<b>date</b>	Giriş alanına bir tarih seçici tanımlar
<b>range</b>	Belirlenen bir aralıktan değer seçilmesini sağlayan kontrolcü tanımlar
<b>hidden</b>	Kullanıcılar tarafından görülemeyen bir alan tanımlar

### Form Verilerini İşleme

Bir formdan gelen verileri yakalamak ve işleme tabi tutmak için 2 farklı yöntem vardır. Bu yöntemlerden hangisinin kullanılacağına karar vermek için verilerin gönderildiği formun *method* öz niteliğine bakılmalıdır. Bu yöntemler POST ve GET metotlarıdır. Hangi verinin hangi isimle yakalanacağına da yine form içerisindeki alanların *name* öz niteliğine bakılarak karar verilir. Kullanıcı adı ve şifre alanlarından gelen verileri yakalamak için POST yöntemi şu şekilde kullanılır.

```
1. <?php
2.
3.     echo 'Kullanıcı Adı: '.$_POST["kullaniciAdi"];
4.
5.     echo '<br>';
6.
7.     echo 'Kullanıcı Adı: '.$_POST["sifre"];
8.
9. ?>
```

Aynı işlemin GET metoduyla yapılması ise şu şekildedir.

```
1. <?php
2.
3.     echo 'Kullanıcı Adı: '.$_GET["kullaniciAdi"];
4.
5.     echo '<br>';
6.
7.     echo 'Kullanıcı Adı: '.$_GET["sifre"];
8.
9. ?>
```

Her ne kadar benzer işlemlerin yapılıyor gibi görünse de verilen gönderilmesi birbirinden tamamen farklı 2 durum meydana getirmektedir. GET metodunda veriler bir URL aracılığı ile iletilirken POST metodunda veriler arka planda işlenecek sayfaya iletilirler. Bu nedenle bu örnekte olduğu gibi kullanıcı şifresi gibi gizli bilgiler aktarılacaksa o zaman GET metodu yerine POST metodu tercih edilmelidir.

### Kaynaklar

Duckett, J. (2014) HTML and CSS: Design and Build Websites (1st ed. pp. 144-176). Wiley Publishing.

HTML Forms - [https://www.w3schools.com/html/html\\_forms.asp](https://www.w3schools.com/html/html_forms.asp)

PHP Kılavuzu - <http://php.net/manual/tr/index.php>

PHP Tutorials - <https://www.w3schools.com/php/default.asp>