

WEB TASARIM I

Öğr. Gör. M. Mutlu YAPICI

Ankara Üniversitesi
Elmadağ Meslek Yüksekokulu

Ders İzlenesi

Hafta	Modüller/İçerik/Konular
1. Hafta	PHP Tanımı ve Sunucu Kurulumları
2. Hafta	PHP Yazım Notasyonu ve Değişkenler
3. Hafta	PHP de Karar kontrol yapıları ve Döngüler
4. Hafta	Dizi ve Dizi işlemleri
5. Hafta	Fonksiyon, Sınıf ve Nesne Kavramları
6. Hafta	HTML ve PHP
7. Hafta	PHP ile Veritabanı İşlemleri
8. Hafta	MYSQL, MYSQLİ
9. Hafta	AJAX
10. Hafta	
11. Hafta	
12. Hafta	
13. Hafta	
14. Hafta	

Bu Ünite de Ele Alınan Konular

- HTML, CSS, JavaScript ve PHP
- PHP ile HTML Metin ve Css İşlemleri
- PHP ile HTML Form ve Css İşlemleri
- PHP ile HTML Form Elemanları İşlemleri
- PHP ile HTML Form Veri Gönderme İşlemleri
- PHP ile HTML Gizli Veri Gönderme İşlemleri

Ders Kazanımları

Bu bölümü Bitirdiğimizde,

- HTML, CSS, JavaScript ve PHP
- PHP ile HTML Metin ve Css İşlemleri
- PHP ile HTML Form ve Css İşlemleri
- PHP ile HTML Form Elemanları İşlemleri
- PHP ile HTML Form Veri Gönderme İşlemleri
- PHP ile HTML Gizli Veri Gönderme İşlemleri

öğrenmiş olacaksınız.

HTML ve PHP

HTML ve PHP birbirini tamamlayan dillerdir. HTML istemci tarafındaki tasarım ve görevleri yerine getirirken, PHP de sunucu tarafındaki işlemleri yerine getirmektedir.

Bu düzen içerisinde bazen HTML tagleri arasında PHP, bazen de PHP kodları arasında HTML kodları yazıp işleme almamız gerekmektedir.

Web sayfası ile veritabanı arasındaki veri alışverişini yapabilmemiz için HTML ve PHP yi birlikte ve etkileşimli şekilde kullanmak zorundayız.

HTML ve PHP

PHP kodu içerisinde HTML taglerini direkt olarak kullanamamaktayız. PHP kodları sunucu tarafında yorumlanıp çalıştığı ve HTML kodlarında istemci tarafında internet tarayıcılarında yorumlandığı için, PHP içerisinde direkt yazılan HTML kodları yorumlanamayıp çalışmıyor ve hata veriyorlar. Biz sunucunun bu HTML kodlarını dikkate almadan sanki bir metinmiş gibi tarayıcıya gönderilmesini ve tarayıcı tarafından yorumlanmasını sağlamalıyız.

Tabi bunu yapabilmek için de bu kodları PHP sayfasında normal metinleri tanımladığımız gibi çift tırnak " " arasında yazmak zorundayız.

HTML ve PHP

Yanlış kullanım :

<?php

<h1> Ben H1 HTML Tagiyim </h1>

?>

Doğru kullanım ;

<?php

Echo "<h1> Ben H1 HTML Tagiyim </h1>";

?>

HTML ve PHP

PHP 'de meta tag gibi üst başlık tanımlamaları header() fonksiyonu ile olmaktadır. Yönlendirme için `header('location:http://www.emyo.ankara.edu.tr');` kullanılır. Yine aynı şekilde karakter türünü tanımlamak için de 'charset' parametresi kullanılır.

Hatalı yazım.

```
<?php  
    echo "Merhaba PHP Geliştiricileri";  
?>
```

Doğru kullanım ;

```
<?php  
header('content-type: text/HTML; charset= utf-8');  
    echo "Merhaba PHP Geliştiricileri";  
?>
```


HTML ve PHP

PHP 'de meta tag gibi üst başlık tanımlamaları header() fonksiyonu ile olmaktadır. Yönlendirme için `header('location:http://www.emyo.ankara.edu.tr');` kullanılır. Yine aynı şekilde karakter türünü tanımlamak için de 'charset' parametresi kullanılır.

```
<?php
    if($_GET['sifre']==1234)
    {
        header('location:admin.php');
    }else{

        header('location:giris.php');
    }
?>
```

HTML ve PHP

İstersek PHP kod ve değişkenleri ile etkileşimli bir şekilde de kullanabiliriz ,

```
<?php
```

```
$adim="Mutlu";
```

```
$sadam="Yapıcı";
```

```
Echo "<font> Merhaba Sahip ".$adim." ". $sadam."
```

```
Sayfana Hoş Gelmişsen </font>";
```

```
?>
```

HTML ve PHP

Biliyorsunuz ki HTML taglerine parametre ekleyip renk boyut ve font gibi ayarlar yaparak zenginlik kazandırabiliyoruz. Bu parametrelere değer atarken de, bu değerleri biz `color=" red "` de olduğu gibi çift tırnak içerisine yazıyoruz. Ancak PHP de çift tırnağı metinleri yazmak için kullanıyoruz. İşte PHP de HTML tagini tırnak içerisine yazıp birde parametre eklediğimizde bu parametrenin tırnaklarıyla PHP nin tırnak işaretleri karışıyor.

Anlatırken bile cümleler karıştı :D örneklerle açıklama daha kolay olacak sanırım.

Örneğin aşağıdaki kodu yazarsanız SYNTAX (yazım) hatası alırsınız. Çünkü ECHO dan sonra tırnak açmışsınız metin yazacaksınız ama yazdığınız metinde parametreye değer atamak için yine tırnak kullandınız (kırmızı tırnaklar) oysa PHP onu dikkate almasın istiyordunuz. Ama PHP onu sadece parametreler için yazdığınızı bilmiyor. Ve hata oluşuyor.

`<?php`

`$adim="Mutlu";`

`$sadi="Yapıcı";`

`Echo " Merhaba Sahip ".$adim." ". $sadi." Sayfana Hoş Gelmişsen ";`

`?>`

HTML ve PHP

Bu gibi durumlarda yani aslında PHP de anlamlı olarak kullanılan karakterlerin dikkate alınmasını istemiyorsanız kaçış karakteri olan **tesr slash** \ karakterini kullanmalısınız.

Doğru kullanım şekli;

```
<?php
```

```
$adim="Mutlu";
```

```
$sadem="Yapıcı";
```

```
Echo "<font face=\\" arial \\" colro=\\" red \\" > Merhaba Sahip  
\".$adim.\" \". $sadem.\" Sayfana Hoş Gelmişsen </font>\";
```

```
?>
```

HTML ve PHP

Diğer bir kullanım şeklide bu gibi durumlarda çift tırnağı PHP için kullandıysanız tek tırnağı da HTML parametreleri için kullanabilirsiniz.

Doğru kullanım şekli 2;

```
<?php
```

```
$adim="Mutlu";
```

```
$sadam="Yapıcı";
```

```
Echo "<font face=' arial ' colro='red' > Merhaba Sahip  
". $adim. " ". $sadam. " Sayfana Hoş Gelmişsen </font>";
```

```
?>
```

HTML ve PHP

PHP içerisinde HTML kullanmayı gördük peki ama CSS kodlarını nasıl kullanacağız?

HTML ile aynı mantıkta kullanabiliriz. Yani tüm CSS kodlarını sanki normal bir metin gibi PHP arasında kullanabiliriz.

Kullanım şekli ;

```
<?php  
echo "
```

```
    <style>  
        .aslanlar{  
            font-family:tahoma;  
            background-color:#00FFFF;  
            border:solid 5px #FF00FF;  
            color:red;  
        }  
        .aslanlar:hover{  
            background-color:red;  
            border:solid 5px #FFFFFF;  
            color:#00FFFF;  
        }  
    </style>
```

```
";
```

```
$adim="Mutlu";
```

```
$sadam="Yapıcı";
```

```
Echo "<font face=' arial ' colro='red' class='aslanlar' > Merhaba Sahip ".$adim." ". $sadam." Sayfana Hoş Gelmişsen  
</font>";
```

```
?>
```

HTML ve PHP

PHP içerisindeki javascript kullanımı da aynı şekilde olmaktadır.

Kullanım şekli ;

```
<?php  
echo "
```

```
    <script>  
        function tiklama(a)  
        {  
            alert(a+' Tıklamasana Kardeşim.');        }  
    </script>
```

```
";
```

```
$adim="Mutlu";  
$sadam="Yapıcı";  
Echo "<font color='red' onclick='tiklama(\"".$adim." ".$sadam."\\");' >  
Merhaba Sahip ".$adim." ". $sadam." Sayfana Hoş Gelmişsen AMA SAKIN  
TIKLAMA </font>";
```

```
?>
```

HTML ve PHP

Yukarıdaki örneklere baktığımızda PHP içerisinde HTML, CSS, JavaScript kullanırken PHP ve diğer kodların çift tırnakları karışmasın diye ya tek tırnak kullandık ya da kaçış / karakterini kullandık. Sizde fark etmişsinizdir ki bu oldukça can sıkıcı ve işi karmaşıklaştıran bir yöntemdir. İsterseniz bunu yerine PHP nin Blok yazım yöntemini kullanabilirsiniz.

Kullanım şekli ;

`<?php`

`echo <<<java`

```
<script>
    function tiklama(a)
    {
        alert(a+' Tıklamasana Kardeşim.');
```

```
</script>
```

`java;`

//üstteki bitiş kodunun başına ya da sonuna hiçbir şekilde boşluk yada karakter koymayın

`$adim="Mutlu";`

`$sadam="Yapıcı";`

`Echo <<<font`

``

`Merhaba Sahip $adim $sadam Sayfana Hoş Gelmişsen AMA SAKIN TIKLAMA`

``

`font;`

//üstteki bitiş kodunun başına ya da açıklama satırı bile koymayın

`?>`

HTML ve PHP

Evet şimdiye kadar PHP içerisinde HTML yazmayı gördük, peki ama HTML içerisinde PHP nasıl kullanılır şimdi de ona bakalım. Arkadaşlar HTML içerisinde PHP kullanırken dikkat etmemiz gerek en önemli konu bir **kere kesinlikler dosyanızın uzantısı PHP olmalıdır**. HTML kodu yazsanız bile dosya uzantınız PHP olmalıdır. Daha sonra her php kodu yazacağınız ayrı bölümlere mutlaka PHP yazdığınızı belirten `<?php ?>` başlangıç ve bitiş kodlarını yazmalısınız.

Kullanım şekli ;

```
<html>
  <head>
    <title> HTML içerisinde PHP kodu</title>
    <script>
      function tiklama(a)
      {
        alert(a+' Tıklamasana Kardeşim.');
```

```
      }
    </script>
  </head>
  <body bgcolor="yellow">
    <h1 align="center"> Merhaba Arkadaşlar </h1>
    <?php
      $adim="Mutlu";
      $sadam="Yapıcı";
    ?>
    <font face="arial" color="red" size='6' onclick="tiklama(' <?php echo $adim.' ".$sadam; ?> ');"> Merhaba
    Sahip <?php echo $adim.' ".$sadam; ?> Sayfana Hoş Gelmişsen AMA SAKIN TIKLAMA </font>
  </body>
</html>
```

HTML ve PHP

HTML ve PHP birlikte kullanımına bir örnek daha. Örneğin bir diziyi tabloya dönüştürmek.

Kullanım şekli ;

```
<html>
  <head>
    <title> HTML içerisinde PHP kodu</title>
    <meta charset="utf-8">
  </head>
  <body bgcolor="yellow">
    <h1 align="center"> Merhaba Arkadaşlar Diziyi Tabloya Dönüştürelim</h1>
    <?php
      $dizi= array("adi"=>"Mutlu YAPICI", "Bolumu"=>"Bilgisayar Teknolojileri", "no"=>12345
    );
    echo <<<tablo
      <table border="2">
        <tr> <th> Adı</th> <th> Bölümü</th> <th> Numarası</th>
        <tr>
          <td> Mutlu YAPICI </td>
          <td> Bilgisayar Teknolojileri </td>
          <td> 12345 </td>
        </tr>
      </table>
    </?php>
  </body>
</html>
```

HTML ve PHP

HTML ve PHP birlikte kullanımına bir örnek daha. Örneğin bir diziyi tabloya dönüştürmek.

Aynı örneği implode() fonksiyonu ile yapalım

```
<?php
```

```
$dizi= array("adi"=>"Mutlu YAPICI", "Bolumu"=>"Bilgisayar Teknolojileri",  
"no"=>12345 );
```

```
echo <<<tablo  
<table border="2">  
<tr> <th> Adı</th> <th> Bölümü</th> <th> Numarası</th> </tr>  
<tr>  
    <td>
```

```
tablo;
```

```
echo implode("</td><td>", $dizi);
```

```
echo <<<tablo2  
    </td>  
</tr>  
</table>
```

```
tablo2;
```

```
?>
```

HTML Form İşlemleri ve PHP

HTML Formlarının PHP ile birlikte kullanılması interaktif bir web sitesinin olmazsa olmazıdır. Buraya kadar PHP ve HTML kodlarının nasıl birlikte kullanılacağını gördük. Form elamanları da tasarım bölümünde aynı şekilde kullanılmaktadır. Farklılık HTML den PHP ye veri aktarımı kısmında ortaya çıkmaktadır.

HTML formlarından PHP ye URL'ye eklenerek ya da arkaplanda gizli olarak iki farklı şekilde veri aktarımı olmaktadır.

Öncelikler verileri aktara bileceğimiz bir form sayfasını oluşturalım. Bunu direk HTML sayfası olarak tasarlayabilirsiniz ancak uzantısı php olmalı.

Örnek kod:

```
<html><head> <title> HTML içerisinde PHP kodu</title>      <meta charset="utf-8"> </head>
  <body bgcolor="yellow">
  <h1 align="center"> Kullanıcı Girişi </h1>
    <form action="giris.php" Method="GET" name="k_giris">
      Kullanıcı Adı :
      <input type="text" name="kadi" value="" placeholder="Kullanıcı adını gir"/>
      <br/>
      Kullanıcı Şifresi :
      <input type="text" name="ksfr" value="" placeholder="Kullanıcı şifreni gir"/>
      <br/>
      <input type="submit" name="btn" value="Gönder Gitsin"/>
    </form>
  </body>
</html>
```

HTML Form İşlemleri ve PHP

Bu sayfada kırmızı olarak gösterilen parametreler önemli. Form tagi içerisindeki **action='giris.php'** verilerin gönderileceği sayfayı tanımlamaktadır. **Method='GET'** verilerin nasıl gönderileceğini tanımlamaktadır. GET metodu ile göndermekteyiz yani gönderilen tüm verileri URL de görebileceğiz. Input taglerinde tanımlanan **name=''** verileri çok önemli çünkü biz verilerimizi burada tanımladığımız isimle göndereceğiz yani **her name bir değişken** olacak. En sondaki input bir butondur ve tipi submit olmaz ise verileri gönderemeyiz.

Örnek kod:

```
<html><head> <title> HTML içerisinde PHP kodu</title>      <meta charset="utf-8"> </head>
  <body bgcolor="yellow">
    <h1 align="center"> Kullanıcı Girişi </h1>
      <form action="giris.php" Method="GET" name="k_giris">
        Kullanıcı Adı :
        <input type="text" name="kadi" value="" placeholder="Kullanıcı adını gir"/>
          <br/>
        Kullanıcı Şifresi :
        <input type="text" name="ksfr" value="" placeholder="Kullanıcı şifreni gir"/>
          <br/>
        <input type="submit" name="btn" value="Gönder Gitsin"/>
      </form>
    </body>
  </html>
```

HTML Form İşlemleri ve PHP

Gördünüz gibi bu formdan giris.php sayfasına veriler gönderildi ve aslında gönderdiğimiz tüm verileri URL de değişken adları ile birlikte görebiliyoruz. Dikkat ettiyseniz giriş.php dosya yolundan sonra soru işareti ? gelmiştir bunun anlamı sonraki gelen veriler değişken ve içerikler demektir. adi=Mutlu verisi URL de gidiyor ve yine dikkat edilirse verileri ayırmak için her veri arasında & işareti kullanılmaktadır. **kadi=Mutlu&ksfr=1234&btn=Gönder+Gitsin**. Değişken adlarına baktığımızda hepsinin aslında input taginin name parametresine yazdığımız değerler olduğu görülmektedir. Bu yüzden name parametrelerine verilen isimler değişken tanımlama kurallarına uygun olmalıdır.

Veriler ~~Değişkenler~~

Array ([kadi] => Mutlu [ksfr] => 1234 [btn] => Gönder Gitsin)

Kullanıcı Girişi

Kullanıcı Adı :

Kullanıcı Şifresi :

HTML Form İşlemleri ve PHP

Formdan veri gönderimi bu şekilde olmaktadır. Peki gönderdiğimiz verileri karşı taraftan nasıl alacağız. `giris.php` sayfasına yazacağımız kod gelen verileri alıp kullanmamız sağlamaktadır. Tabi buradaki önemli nokta gönderdiğimiz sayfadaki form tagi nin `METHOD` parametresinin değeridir. Veriyi hangi methodla gönderdiysek karşıdan da onunla almalıyız.

```
<form action="giris.php" Method="GET" name="k_giris">
```

Baktığımızda metodumuzun `GET` olduğu görülmektedir o zaman bu verileri de `$_GET` değişkeni ile almamız gerekmektedir. Gönderilen tüm veriler `$_GET` değişkeninde dizi halinde gelmektedir. Bunları `print_r($_GET);` diyerek dizilerde olduğu gibi ekrana yazdırabilir ya da `Echo $_GET['kadi'];` gibi tek tek değişken ismine göre de alabiliriz.

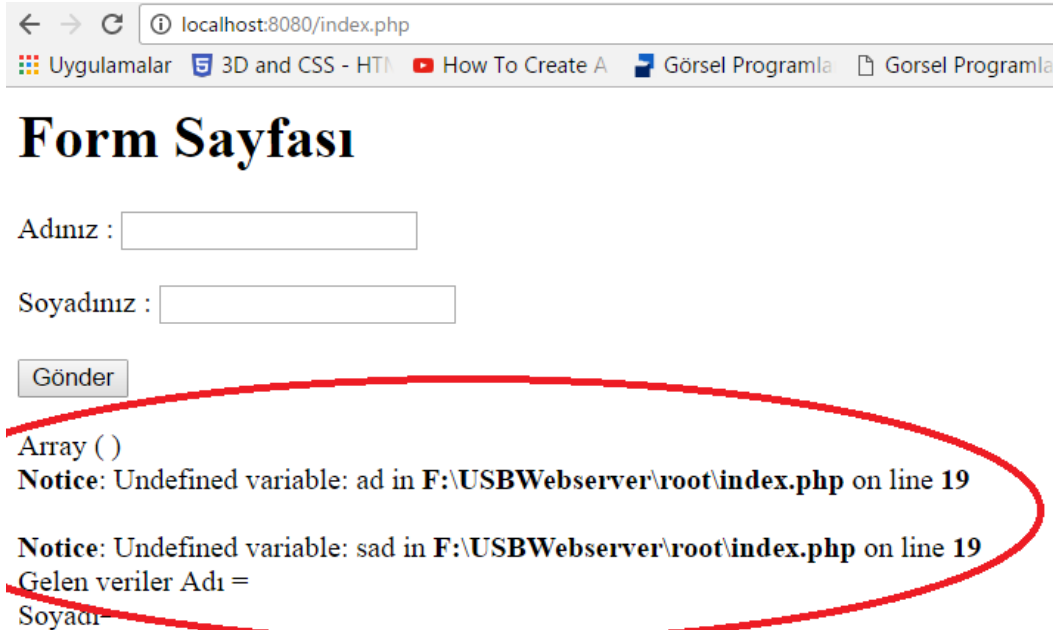
Örnek kod:

```
<?php  
    print_r($_GET);//Gelen tüm verileri Dizi halinde yazdıralım  
    echo $_GET["kadi");//sadece kullanıcı adını alalım  
?>
```

HTML Form İşlemleri ve PHP

PHP de verileri gönderdiğiniz sayfa farklı bir sayfa olmak zorunda değildir. İsterseniz aynı sayfada hem formu bulundurup hem de verileri aynı sayfaya gönderebilirsiniz.

Bu durum dikkat etmeniz gereken nokta , sayfa ilk kez açıldığında veriler gönderilmemiş olacağından, ancak aynı sayfada verileri okumaya çalışan PHP kodları olacağından ilk açılışta aşağıdaki gibi hata mesajı çıkabilir.



The screenshot shows a web browser window with the address bar displaying "localhost:8080/index.php". The browser tabs include "Uygulamalar", "3D and CSS - HT", "How To Create A", "Görsel Programla", and "Gorsel Programla". The page title is "Form Sayfası". Below the title, there are two input fields: "Adınız :" and "Soyadınız :". A "Gönder" button is located below the input fields. The page content is partially obscured by a red oval highlighting the error messages. The error messages are:

```
Array ()  
Notice: Undefined variable: ad in F:\USBWebserver\root\index.php on line 19  
  
Notice: Undefined variable: sad in F:\USBWebserver\root\index.php on line 19  
Gelen veriler Adı =  
Soyadı =
```


HTML Form İşlemleri ve PHP

Ancak gönder butonuna basıp verileri gönderdiğinizde hata kaybolacaktır.

```
<html>
```

```
<body> <h1> Form Sayfası</h1>
```

```
<form action="index.php" method="GET" name="frm1">
```

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

```
Soyadınız : <input type="text" name="sad" /><br><br>
```

```
<button value="a" name="btn">Gönder</button>
```

```
</form>
```

```
</body>
```

```
</html>
```

```
<?Php
```

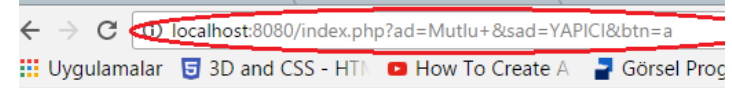
```
print_r($_GET);
```

```
extract($_GET);
```

```
echo"<br>Gelen veriler Adı = $ad
```

```
<br> Soyadı= $sad";
```

```
?>
```



Form Sayfası

Adınız :

Soyadınız :

```
Array ( [ad] => Mutlu [sad] => YAPICI [btn] => a )
```

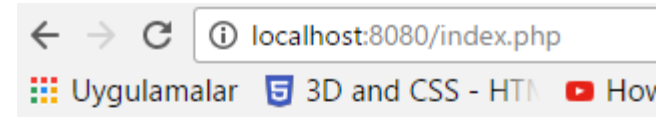
```
Gelen veriler Adı = Mutlu
```

```
Soyadı= YAPICI
```

HTML Form İşlemleri ve PHP

İlk açılışta aslında `echo"
Gelen veriler Adı = $ad
 Soyadı= $sad";` satırında bulunan `$ad` ve `$sad` değişkenlerinin değerleri olmadığı için hata veriyor. Biz bu değişkenleri `isset()` ile kontrol ettirip eğer değer var ise ekrana yazma işlemi yaparak hatayı engelleyebiliriz.

```
<html>
  <body> <h1> Form Sayfası</h1>
  <form action="index.php" method="GET" name="frm1">
    Adınız : <input type="text" name="ad" /><br><br>
    Soyadınız : <input type="text" name="sad" /><br><br>
    <button value="a" name="btn">Gönder</button>
  </form>
</body>
</html>
<?Php
extract($_GET);
if(isset($ad) && isset($sad)){
  print_r($_GET);
  echo"<br>Gelen veriler Adı = $ad <br>
    Soyadı= $sad";
}
?>
```



Form Sayfası

Adınız :

Soyadınız :

Gönder

HTML Form İşlemleri ve PHP

İsterseni bu değişkenleri `empty()` ile kontrol ettirip eğer değişken boş değilse ekrana yazma işlemi yaparak hatayı engelleyebiliriz.

```
<html>
```

```
  <body> <h1> Form Sayfası</h1>
```

```
  <form action="index.php" method="GET" name="frm1">
```

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

```
    Soyadınız : <input type="text" name="sad" /><br><br>
```

```
    <button value="a" name="btn">Gönder</button>
```

```
  </form>
```

```
</body>
```

```
</html>
```

```
<?Php
```

```
  extract($_GET);
```

```
  if( ! empty($ad) && ! empty($sad)){
```

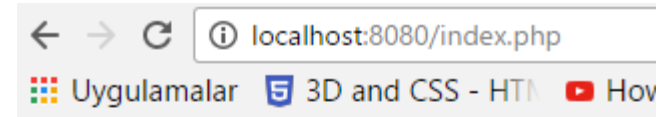
```
    print_r($_GET);
```

```
    echo"<br>Gelen veriler Adı = $ad <br>
```

```
      Soyadı= $sad";
```

```
  }
```

```
?>
```



Form Sayfası

Adınız :

Soyadınız :

Gönder

KAYNAKLAR

- İnternet ortamı
- PHP ve AJAX Haydar TUNA
- A'dan Z'ye PHP Rıza ÇELİK

