



Burdur
Mehmet Akif Ersoy
Üniversitesi

MAKÜ|GMYO
GÖLHİSAR MESLEK YÜKSEKOKULU

İnternet Programcılığı I

Öğr. Gör. Hüseyin Şengün
hsengun@mehmetakif.edu.tr

8. Hafta

FORM VE ELEMENTLERİ



FORM VE ELEMENTLERİ

- İnternet tabanlı programlarda da kullanıcılardan bilgi girişi istenir ve girilen o bilgiler işlenerek işlem yapılır.
- Bu bilgi girişi internet tabanlı programlamada genellikle form ve elementleri ile yapılır.
- Form ve elementleri HTML (Hyper Text Markup Language) elementleridir.
- Örnek:
- `<form action="islem.php" method="post"></form>`



FORM VE ELEMENTLERİ

- Form elementinin parametrelerinden olan **method** genel olarak iki değer alabilir.
- Bunlar;
- **GET:** Formun method kısmına bu değer yazılırsa formdaki gönderilen bilgiler tarayıcının adres çubuğunda gözükür ve bu metodu kullanırken en fazla 255 karakter veri gönderilmesi tavsiye edilir.
- Bu karakter sayısı tarayıcıya (Internet Explorer, Mozilla Firefox veya Chrome gibi) ve sunucuya (Apache, IIS gibi) göre değişiklik gösterirler.
- Bu sınırlamalardan dolayı GET metodu ile veri gönderirken ve alırken verinin boyuna dikkat edilmelidir.



FORM VE ELEMENTLERİ

- **POST:** Formun method kısmına bu değer yazılırsa formdaki gönderilen bilgiler tarayıcının adres çubuğunda gözükmez ve sunucu ayarına göre sınırsız boyutta veri gönderilebilir. GET methoduna göre biraz daha güvenli ancak daha yavaştır.
- Kullanıcının veri gönderebilmesi için kullanılan html elementlerine form elementleri denir. Bu elementler kullanıcıların daha kolay veri gönderebilmesi için birçok tür vardır.

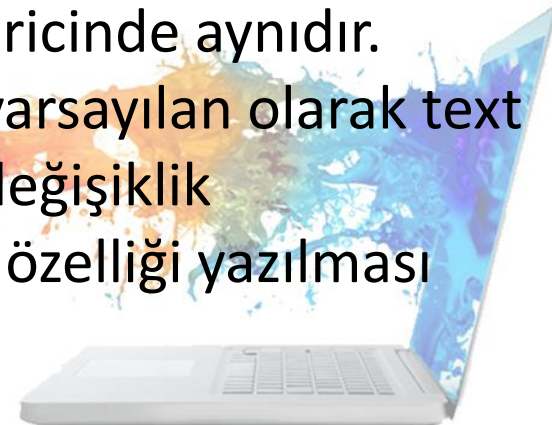


FORM VE ELEMENTLERİ

- **Form Elementleri**

tür	açıklama
text	Kısa metin girişi için
password	Şifre girişi için
submit	Form verilerinin gönderilmesi için kullanılan buton
textarea	Uzun metin girişi için

- Form elementlerinin kullanım şekli textarea haricinde aynıdır. Input elementlerinde type özelliği yazılmazsa varsayılan olarak text yazıldığı kabul edilir. Ancak bu tarayıcıya göre değişiklik gösterebileceği için Input elementlerinde type özelliği yazılması tavsiye edilir.





FORM VE ELEMENTLERİ

```
10 <form action="islem.php" method="post"/>
11   <input type="text" name="yazi"/>
12   <br/>
13   <input type="password" name="sifre"/>
14   <br/>
15   <input type="submit" value="Gönder"/>
16   <br/>
17   <textarea name="uzun yazi"/></textarea>
18 </form>
```

- input kullanım şekli:

`<input type="tür" name="nesne adı" value="değer" />`

- Textarea kullanım şekli:

`<textarea name="adı"> değer </textarea>`



Form elementlerindeki verilerin işlenmesi

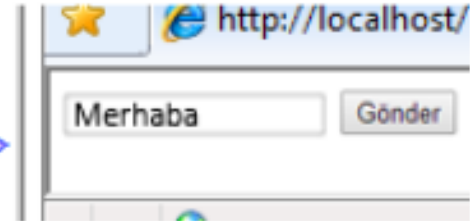
- Daha önceki örneklerimizde bazı form elementlerinin kullanım şekli gösterilmiştir.
- Şimdi de bu form elementlerine girilen değerlerin PHP tarafından nasıl işleneceğini görelim.
- Formdaki elementin değerini almak PHP'de “**\$_GET**” ve “**\$_POST**” **global değişkenleri** kullanılır ve bu global değişkenler **dizi** yapısındadır.
- Kullanım şekli:
 - **\$_GET** [input'un name özelliğinde yazılan ad]
 - **\$_POST** [input'un name özelliğinde yazılan ad]



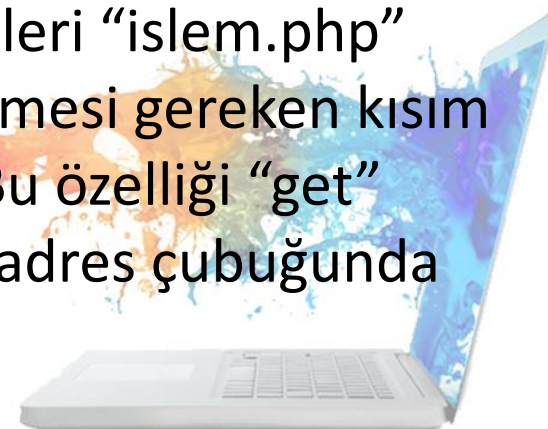
Form elementlerindeki verilerin işlenmesi

- Bir formun içerisine “text” alanı ekleyerek “GET” metodu ile girilen bu değeri ekrana yazdıralım.
- Bu örnek için önce “form.php” dosyası oluşturalım.

```
10 <form action="islem.php" method="get" />
11 <input type="text" name="yazi" />
12 <input type="submit" value="Gönder" />
13 </form>
```



- Yukardaki örnekte “yazi” adında bir text alanı oluşturduk ve “Gönder” butonuna tıklanınca doldurulan bilgileri “islem.php” dosyasına gönderilecek. Bu örnekte dikkat edilmesi gereken kısım formun “method” özelliğine “get” yazmamız. Bu özelliği “get” olarak ayarladığımız için gönderdiğimiz veriler adres çubuğunda görünecek.



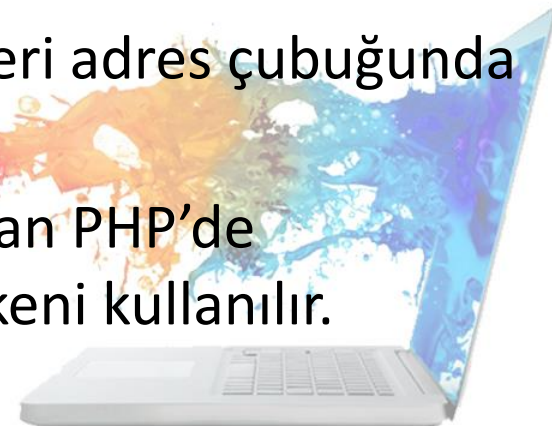
Form elementlerindeki verilerin işlenmesi

```
2 <?php
3     echo $_GET["yazi"];
4 ?>
```

http://localhost/islem.php?yazi=Merhaba

Merhaba

- Yukardaki örnekte formda “name” özelliği “yazi” olanın içeriğini ekrana yazdırdık.
- Formda doldurduğumuz “yazi” inputunun değeri adres çubuğunda gözükmemektedir.
- Oluşturulan formun metodu “get” olduğu zaman PHP’de gönderilen veriyi alırken “\$_GET” global değişkeni kullanılır.

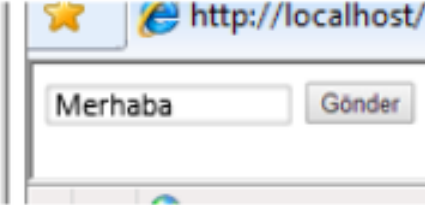




Form elementlerindeki verilerin işlenmesi

- Yukarıdaki örneği bu defa da “post” metodu ile yapalım.

```
10 <form action="islem.php" method="post"/>  
11 <input type="text" name="yazi"/>  
12 <input type="submit" value="Gönder"/>  
13 </form>
```



- Formumuzu oluşturduk. Bu sefer “method” kısmına “get” yerine “post” yazdık.



Form elementlerindeki verilerin işlenmesi

- Aynı işlemleri “post” metodu ile yaptığımızda gönderilen verinin adres çubuğunda gözükmez.
- Oluşturulan formun metodu “post” olduğu zaman PHP’de gönderilen veriyi alırken “\$_POST” global değişkeni kullanılır.

```
2 <?php
3     echo $_POST["yazi"];
4 ?>
```

http://localhost/islem.php

Merhaba



OTURUM İŞLEMLERİ

- Bir değişkene değer atadığımızda bu değer o sayfa için geçerlidir. Bu değişkeni başka sayfada kullanamayız.
- Bir sayfada tanımlanan değişkeni başka bir sayfalarda kullanmak için “**\$_SESSION**” global değişkeni kullanılır.
- **\$_SESSION** global değişkeni de **\$_GET** ve **\$_POST** değişkenleri gibi dizi yapısındadır. Bu global değişkeni tanımlamadan veya çağırmadan önce **session_start()** fonksiyonunun kullanımı zorunludur.

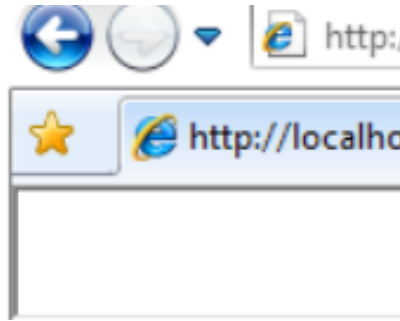


OTURUM İŞLEMLERİ

```
5 <?php
6     $suye_adi = "Ahmet";
7 ?>
```

- Yukardaki örnekte “**\$suye_adi**” adlı değişkene “Ahmet” değeri vererek tanımladık ve bu dosyayı “**tanimla.php**” adı ile kaydettik.
- Bu dosyada tanımladığımız değişkeni “**getir.php**” adındaki dosyadan çağırmayı deneyeceğiz.

```
5 <?php
6     echo $suye_adi;
7
8
9
10 ?>
```



OTURUM İŞLEMLERİ

- Yukardaki örnekte de görülebileceği gibi sıradan değişkenler bir sayfada tanımlandığında başka sayfadan çağırılamaz.
- Fakat “**\$_SESSION**” global değişkeni bu işlemi yapabilmektedir.

```
5 <?php
6     session_start();
7     $_SESSION["uye_adi"] = "Ahmet";
8 ?>
```

- Yukardaki örnekte **session_start()** fonksiyonu ile bu sayfada “**\$_SESSION**” global değişkeni ile işlem yapacağımızı belirttik ve “**\$_SESSION**” değişkenini tıpkı dizilerde olduğu gibi index vererek tanımladık. Bu sayfayı “tanimla.php” olarak kaydettik. Şimdide bu sayfada tanımladığımız değişkeni “getir.php” adından sayfa oluşturarak bu sayfada ekrana yazdıralım.



OTURUM İŞLEMLERİ



```
5 <?php
6 session start();
7 echo $_SESSION["uye_adi"];
8
9
10
11 ?>
```

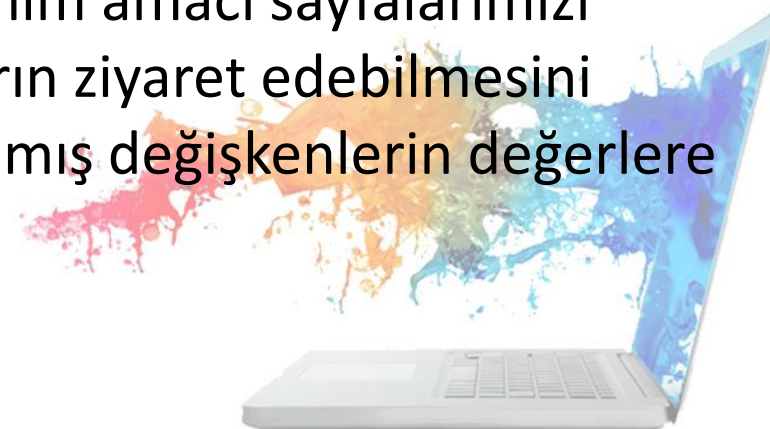
http://
http://localhost
Ahmet
Yerel intranet

- Yukardaki örnekte de görülebileceği gibi “tanımla.php” de oluşturduğumuz bir değeri daha sonra “getir.php” sayfasında ekrana yazdırdık.



OTURUM İŞLEMLERİ

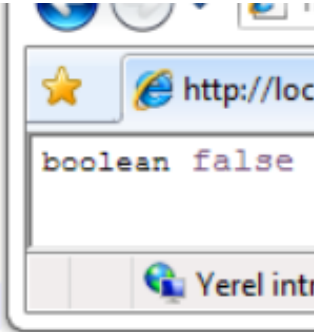
- PHP bu işlemi gerçekleştirmek için önce tarayıcımıza eşsiz bir değer (T.C. Kimlik numaralarında olduğu gibi) gönderiyor ve sunucumuzda bu değer adında bir dosya oluşturuyor.
- Biz “\$_SESSION” değişkenine değer atadığımızda bilgileri bu dosyaya kaydediyor. Kişinin tarayıcısında bu eşsiz değer varsa PHP o dosyadaki değeri bize veriyor biz de istediğimiz işlemleri gerçekleştirebiliyoruz.
- “\$_SESSION” değişkenininin genel kullanım amacı sayfalarımızı sadece bizim belirlediğimiz kullanıcıların ziyaret edebilmesini sağlamak ve farklı sayfalarda tanımlanmış değişkenlerin değerlere erişilebilmesidir.



OTURUM İŞLEMLERİ

- Bu işlemlerden önce **isset()** fonksiyonunu kullanımını anlatılacaktır. Bu fonksiyon değişken tanımlandı ise doğru “true” tanımlanmadı ise yanlış “false” yanıtını döndürür.
- Kullanım şekli: **isset(değişken)**;

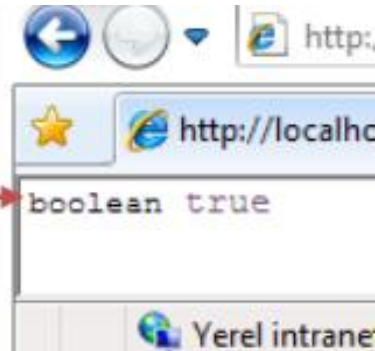
```
5 <?php
6     session_start();
7     var_dump(isset($_SESSION["isim"]));
8
9
10
11 ?>
```



OTURUM İŞLEMLERİ

- Yukardaki örnek çalıştırılmadan önce tarayıcı geçmişini silmeyi unutmayınız. Yukardaki örnekte “\$_SESSION[“isim”]” değişkeni daha önce tanımlanmadığı için isset() fonksiyonu yanlış “false” yanıtını döndürdü. Şimdi de “\$_SESSION[“isim”]” adındaki değişken tanımlayıp bu işlemi tekrar yapalım.

```
5 <?php
6     session_start();
7     $_SESSION["isim"] = "Ahmet";
8     var_dump(isset($_SESSION["isim"]));
9
10
11 ?>
```



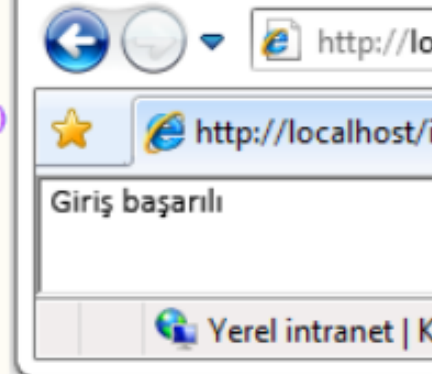
- Yukardaki örnekte “\$_SESSION[“isim”]” adlı değişken tanımlandığı için isset fonksiyonu doğru “true” yanıtını döndürdü.



OTURUM İŞLEMLERİ

- Yukarda gösterilen işlemleri kullanarak bir sayfamızdaki verileri sadece “\$_SESSION[“isim”]” değişkeni tanımlanmışsa ekrana “Giriş başarılı” tanımlanmamışsa “Giriş başarısız” yazdıralım.

```
5 <?php
6     session_start();
7     if(isset($_SESSION["isim"]))
8     {
9         echo "Giriş başarılı";
10    }
11    else
12    {
13        echo "Giriş başarısız";
14    }
15 >
```



OTURUM İŞLEMLERİ

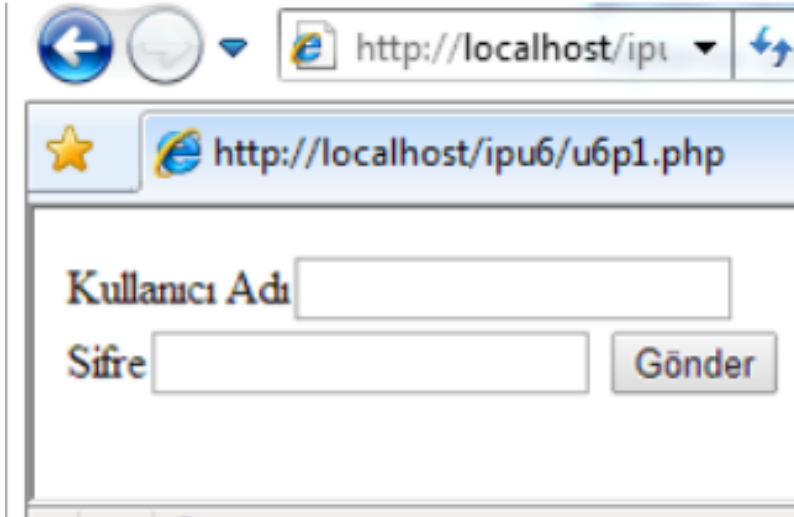
- Şimdiye kadar yaptığımız işlemleri birleştirerek önce bir form oluşturalım. Bu formda kullanıcı adını ve şifresini girsin. Eğer adı alanına “Ahmet” şifre alanına da “123” yazarsa “**islem.php**” sayfasında “Hoşgeldin Ahmet”, farklı bilgiler yazarsa da “Yetkiniz yoktur” mesajı ekrana gelsin.
- İlk olarak formu oluşturalım ve “login.php” adıyla kaydedelim.

```
17 <form action="islem.php" method="post"/>
18     Kullanıcı Adı<input type="text" name="isim"/>
19     <br/>
20     Sifre<input type="password" name="sifre"/>
21     <input type="submit" value="Gönder"/>
22 </form>
```



OTURUM İŞLEMLERİ

- Yukardaki örnekte bir tane “text” tipinde kullanıcı adını yazmak için bir tane de “password” tipinde şifre yazmak için iki tane veri alanı oluşturduk. Dosyamızın adını da “login.php” yapıp kaydettik. Oluşturduğumuz formun ekran görüntüsü aşağıda verilmiştir.



The screenshot shows a web browser window with the address bar containing 'http://localhost/ipu6/u6p1.php'. Below the address bar, there is a login form with two input fields: 'Kullanıcı Adı' (Username) and 'Sifre' (Password). A 'Gönder' (Send) button is located to the right of the password field.

login.php ekran çıktısı



OTURUM İŞLEMLERİ

- Şimdi de sadece bizim belirlediğimiz kişilerin görebileceği sayfayı oluşturalım. Bu dosyaya **islem.php** adını vererek kaydedelim.

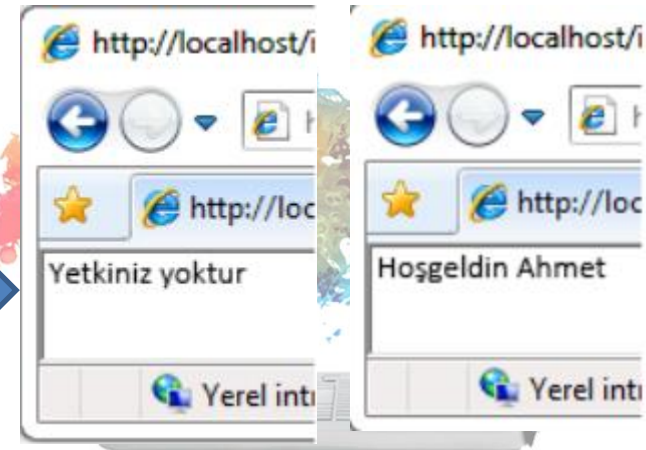
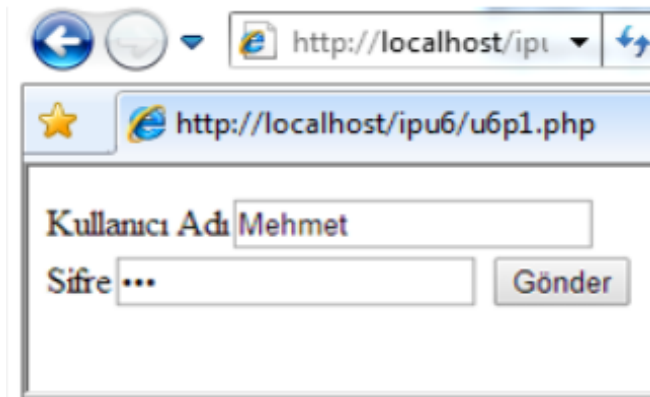
```
5 <?php
6     session_start();
7     if($_POST["isim"] == "Ahmet" AND $_POST["sifre"] == "123")
8     {
9         $_SESSION["isim"] = $_POST["isim"];
10        $_SESSION["sifre"] = $_POST["sifre"];
11
12        echo "Hoşgeldin " . $_SESSION["isim"];
13    }
14    else
15    {
16        echo "Yetkiniz yoktur";
17    }
18 ?>
```

islem.php



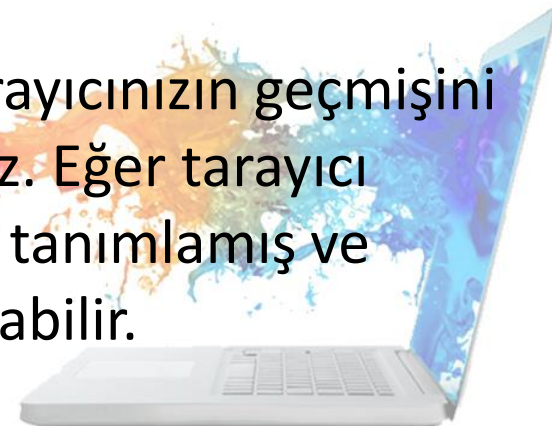
OTURUM İŞLEMLERİ

- Yukardaki örnekte “login.php” sayfasından gelen “\$_POST[“isim”]” değişkeni “Ahmet” ve “\$_POST[“sifre”]” değişkeni “123” ise girilen bilgileri “\$_SESSION” değişkenine atandı ve ekrana “Hoşgeldin Ahmet”, değilse ekrana “Yetkiniz yoktur” yazıldı.
- Tarayıcıdan **login.php** sayfasını açtığımızda ekran görüntüsü aşağıdaki resimdeki gibi olacaktır. Ekrana gelen formda “Kullanıcı Adı” kısmına “Mehmet” “Sifre” kısmına da “123” yazarak “Gönder” butonuna tıklayalım.



OTURUM İŞLEMLERİ

- Kullanıcı grupları oluşturarak farklı kullanıcıları farklı sayfalara yetkilendirme yapabiliriz.
- Yeni bir örnek yapalım ve bu örneğimizde “**ahmet.php**”, “**mehmet.php**”, “**diger.php**” dosyaları oluşturalım.
- Ahmet kullanıcısı “ahmet.php”, Mehmet kullanıcısını ise “mehmet.php” dosyasının içeriğini görebilsin. Formu dolduran herhangi bir kişi (Ahmet ve Mehmet dahil) “**diger.php**” dosyasının içeriğini görebilsin.
- Aşağıdaki işlemleri gerçekleştirmeden önce tarayıcınızın geçmişini siliniz veya kodlarınızı gizli pencerede deneyiniz. Eğer tarayıcı geçmişini silmezseniz daha önden \$_SESSION ile tanımlamış ve atamış olduğunuz değişkenler karışıklığa yol açabilir.



OTURUM İŞLEMLERİ

- Kişilerin adlarını girebilmesi için “**login.php**”dosyasını oluşturarak içerisine aşağıdaki kodları yazalım.

```
20 <form action="islem.php" method="post"/>  
21     Kullanıcı Adı<input type="text" name="isim"/>  
22     <input type="submit" value="Gönder"/>  
23 </form>
```



OTURUM İŞLEMLERİ

- Kullanıcının girdiği bilgiyi “\$_SESSION” değişkenimize atayalım ve oluşturacağımız sayfalara kolay erişim sağlamak için link verelim.

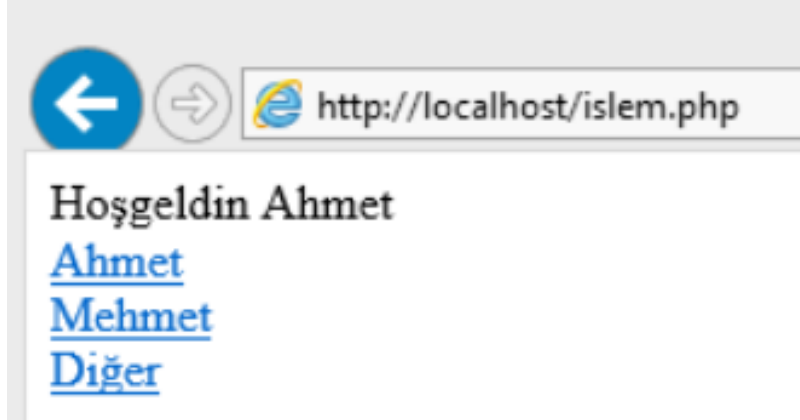
```
5  <?php
6      session_start();
7      $_SESSION["isim"] = $_POST["isim"];
8      .....
9      echo "Hoşgeldin " . $_SESSION["isim"];
10     echo "<br/>";
11
12     echo "<a href='ahmet.php'>Ahmet</a>";
13     echo "<br/>";
14     echo "<a href='mehmet.php'>Mehmet</a>";
15     echo "<br/>";
16     echo "<a href='diger.php'>Diğer</a>";
17  ?>
```

islem.php



OTURUM İŞLEMLERİ

- Yukardaki örnekte formdan gelen veriyi \$_SESSION değişkenine atadık ve ekrana “Hoşgeldin ” formdan gelecek isim yazdırdık. Kullanıcı adına “Ahmet” yazıp “Gönder” butonuna tıklayınca aşağıdaki ekran görüntüsü oluşacaktır.



islem.php ekran çıktısı



OTURUM İŞLEMLERİ

- Yukardaki örnekte “Ahmet”, “Mehmet” veya “Diğer” yazılarının üzerine tıklanınca sayfalar hata verecektir. Çünkü henüz o sayfaları oluşturmadık. Şimdi “ahmet.php” sayfasını oluşturalım.

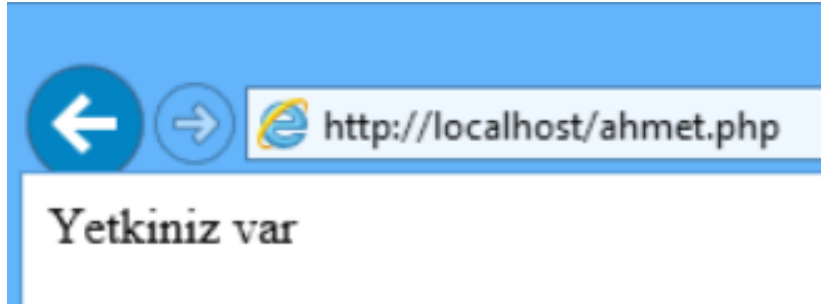
```
5 <?php
6     session_start();
7     if($_SESSION["isim"] == "Ahmet")
8     {
9         echo "Yetkiniz var";
10    }
11    else
12    {
13        echo "Bu sayfayı sadece Ahmet görebilir";
14    }
15 ?>
```

ahmet.php



OTURUM İŞLEMLERİ

- Yukardaki örnekte eğer önceki formda kullanıcı adı olarak “Ahmet” yazılmışsa ekrana “Yetkiniz var” farklı bir şey yazılmışsa “Bu sayfayı sadece Ahmet görebilir” yazısı gelecek. Ben daha önceden kullanıcı adı kısmına “Ahmet” yazdığım içi ekran görüntüsü aşağıdaki gibi olacaktır.



ahmet.php ekran çıktısı



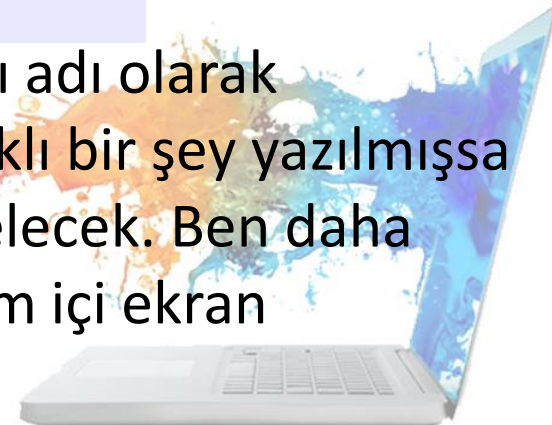


OTURUM İŞLEMLERİ

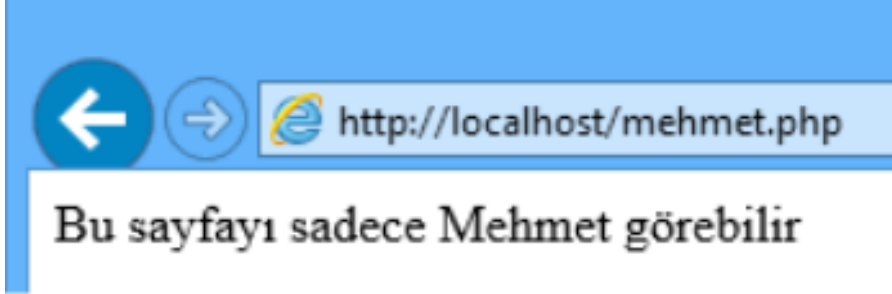
- Şimdi de “mehmet.php” adlı sayfanın kodlarını yazalım.

```
5 <?php
6     session_start();
7     if($_SESSION["isim"] == "Mehmet")
8     {
9         echo "Yetkiniz var";
10    }
11    else
12    {
13        echo "Bu sayfayı sadece Mehmet görebilir";
14    }
15 >
```

- Yukardaki örnekte eğer önceki formda kullanıcı adı olarak “Mehmet” yazılmışsa ekrana “Yetkiniz var” farklı bir şey yazılmışsa “Bu sayfayı sadece Mehmet görebilir” yazısı gelecek. Ben daha önceden kullanıcı adı kısmına “Ahmet” yazdığım içi ekran görüntüsü aşağıdaki gibi olacaktır.



OTURUM İŞLEMLERİ



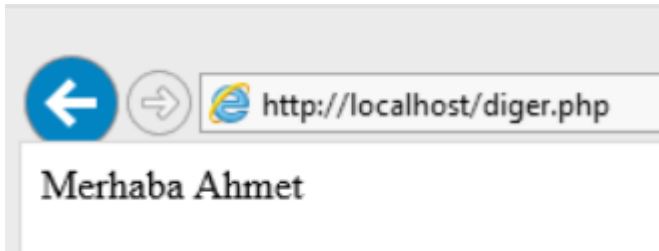
- mehmet.php ekran çıktısı
- Şimdi de "diger.php" adlı sayfanın kodlarını yazalım.

```
5 <?php
6     session_start();
7     if(isset($_SESSION["isim"]))
8     {
9         echo "Merhaba " . $_SESSION["isim"];
10    }
11    else
12    {
13        echo "Formu doldurunuz";
14    }
15 ?>
```



OTURUM İŞLEMLERİ

- Yukardaki örnekte eğer önceki formda kullanıcı adı olarak “Ahmet” yazılmışsa ekrana “Merhaba Ahmet”, “Mehmet” yazılmışsa “Merhaba Mehmet” veya farklı bir isim yazılmışsa “Merhaba” formda yazılan isim yazacak. Ancak form doldurulmadan direk bu sayfaya erişmek isteyen kişinin ekranında “Formu doldurunuz ” yazacak. Ben daha önceden kullanıcı adı kısmına “Ahmet” yazdığım içi ekran görüntüsü aşağıdaki gibi olacaktır.



- diger.php ekran çıktısı



OTURUM İŞLEMLERİ

- Son olarak “`$_SESSION`” değişkeninde bulunan bütün değerleri silmek için **`session_destroy()`** fonksiyonu kullanılır.
- Kullanım şekli: **`session_destroy()`**

```
17 <?php  
18     session_destroy();  
19 ?>
```





Uygulama



Bireysel Etkinlik

- Farklı sayfalar oluşturarak bu sayfalara farklı kişiler yetkilendirin ve bu kişilerin login yapabilmesi için form oluşturun.



Kaynaklar

- PHP Kılavuzu - <http://php.net/manual/tr/index.php>
- PHP Tutorials - <https://www.w3schools.com/php/default.asp>
- PHP Videolu Temel Dersler İçin: Youtube/Php Türkiye
- <https://php-mysql.org/>
- Atatürk Üniversitesi Açık Öğretim Fakültesi

