

WEB PROGRAMLAMA II

Öğr. Gör. M. Mutlu YAPICI

Ankara Üniversitesi
Elmadağ Meslek Yüksekokulu

Ders İzlencesi

Hafta	Modüller/İçerik/Konular
1. Hafta	Oturum yönetimi
2. Hafta	Cookies kullanımı ve oturum yönetimi
3. Hafta	Session kullanımı ve oturum yönetimi
4. Hafta	Sayfalama ve Arama İşlemleri
5. Hafta	JavaScript, JQuery ve PHP
6. Hafta	AJAX ve PHP
7. Hafta	AJAX ve PHP
8. Hafta	ARA SINAV
9. Hafta	PHP'de Nesne Yönelimli Programlamaya Giriş(347)
10. Hafta	PHP de PDO
11. Hafta	
12. Hafta	
13. Hafta	
14. Hafta	

Bu Ünite de Ele Alınan Konular

- Oturum yönetimi
- Cookies kullanımı ve oturum yönetimi
- Session kullanımı ve oturum yönetimi
- Sayfalama ve Arama İşlemleri

Ders Kazanımları

Bu bölümü Bitirdiğimizde,

- **Oturum yönetimi**
- Cookies kullanımı ve oturum yönetimi
- Session kullanımı ve oturum yönetimi
- Sayfalama ve Arama İşlemleri

öğrenmiş olacaksınız.

PHP'de SESSION Yönetimi

Daha öncede bahsettiğimiz gibi PHP de aslında internet yazılımlarının hepsinde iki farklı oturum yönetimi mevcuttur. Bunlar COOCKIE ve SESSION dır. Cookie yada çerez yönetimini anlattık şimdi de sıra SESSION dediğimiz oturum yönetiminde. Daha önce çerezlerin Client dediğimiz kişilerin kedi bilgisayarlarında saklandığını, buna karşılık SESSION ların ise Server dediğimiz sunucularda saklandığını anlatmıştık. Sunuculara dışardan erişimin kısıtlı ve kontrollü olması nedeniyle SESSION kullanımı COOKIE lere göre daha güvenlidir. Şimdi session kullanımı PHP de nasıl oluyor inceleyelim.

Öncelikle SESSION ların kullanıldığı yerlere bakalım. Oturum açmak kavramını duymuşsunuzdur. Bir sitedeki özel tanımlı alanlarda rahatça gezinebilmek için öncelikle o sitede oturum açmanız yani üye girişi yapmanız gerekmektedir. Oturum açtıktan sonra girdiğiniz site içerisinde dolaşırken o oturum bilgileri doğrultusunda web sitesinde rahatlıkla dolaşabilir ve belli başlı size ait işlemleri gerçekleştirebilirsiniz. Örneğin; Facebook sitesinde oturum yaptıktan sonra size ait fotoğrafları yükleyebilir, video paylaşabilir, yazılar yazabilirsiniz. Bu işlemleri gerçekleştirirken hem kendi oturum bilgileriniz dahilinde yaparsınız. Eğer başka bir kişinin bilgileriyle oturum açsaydınız. Başka kişiye ait bir takım işlemleri gerçekleştirebilirdiniz.

PHP'de SESSION Yönetimi

Peki ama internet siteleri bizim üye girişi yapıp yapmadığımızı nereden biliyorlar ve farklı sayfalara tıkladığımızda her sayfa üye girişi yapıp yapmadığımızı nasıl takip ediyor.

CEVAP = SESSION

SESSION dosyaları sayesinde biz oturum açtık mı, ne kadar açık kalacak, hangi bilgilerimiz bu dosyalarda tutulacak belirleye biliyoruz.

SESSION oturum yönetiminde bilmemiz gereken ve olmazsa olmazımız bazı fonksiyonlar mevcuttur. Bunlar ;

- session_start()
- session_id()
- session_destroy()

Şimdi bu fonksiyonları inceleyelim.

PHP'de SESSION Yönetimi

o `session_start()`

Bu fonksiyon kullanıcı sayfaya ilk defa giriyorsa kullanıcı için sunucuda bir SESSION dosyası yaratır ve kullanıma hazırlar.

Eğer daha önce girmişse onu kullanmaya hazırlar. Her bir kullanıcı için (yani sistemdeki her bir üye için) uzun bir kod üretir. Bu kod sayılardan ve harflerden oluşur. Bu koda SESSIONID de denir. İsterseniz bu idyi kendinize göre bir ön ekle de tanımlayabilirsiniz.

Bu `session_start()` fonksiyonunu mutlaka SESSION bilgisine ulaşmak istediğimiz her sayfanın başında tanımlamalıyız yoksa SESSION bilgilerine ulaşamayız ve SESSION fonksiyonları çalışmaz hata alırız.

PHP'de SESSION Yönetimi

● `bool session_start (void)`

Kullanım şekli yukarıdaki gibidir oturum başarılı bir şekilde başlatılırsa true başlatılamazsa false değeri döner

```
$oturum = session_start();
```

```
echo $oturum==true ? "SESSION Oturumu Başlatıldı" :  
"SESSION Oturumu Başarısız";
```

Yukarıdaki örnekte bir oturum başlattık ve başarılı ise bir mesaj verdirdik.

SESSION Dosyamızı oluşturduk ancak içine bir şey yazmadık dolayısıyla okuyamayacağız.

PHP'de SESSION Yönetimi

Session dosyası içerisine bir değişken ve bu değişkene bir değer atamak COOKIE işlemlerine çok benzemektedir. Hatta daha kolaydır diyebiliriz çünkü dizi işlemleri daha basittir burada.

```
$oturum = session_start();  
if($oturum==true){  
    echo "SESSION Oturumu Başlatıldı";  
    //Oturum Dosyamıza Veri Kaydedelim  
    $_SESSION['adi']="Mutlu YAPICI";  
}  
else  
    echo "SESSION Oturumu Başarısız";
```

Dosyamızın içerisine adi diye bir değişken oluşturup içerisine "Mutlu YAPICI" değerini yükler.

PHP'de SESSION Yönetimi

Şimdide başka bir sayfadan okuma işlemini yapalım.

```
$oturum = session_start();  
if($oturum==true){  
    //Okuma işlemi  
    echo $_SESSION['adi'];  
}else  
    echo "SESSION Oturumu Başarısız";
```

Yukarıdaki kod daha önce kaydettiğimiz veriyi okuyup ekrana yazacaktır. Burada dikkat etmeniz gereken en önemli nokta mutlaka veriyi okumandan önce ilk satırdaki `session_start()` fonksiyonunu yazmanız gerektiğidir. O fonksiyonu yazmazsanız oturum dosyanız hazırlanamayacağı için hata mesajı alırsınız.

PHP'de SESSION Yönetimi

Şimdide oturumumuza birden fazla veriyi yani bir diziyi yüklemeye bçalışalım.

```
$oturum = session_start();  
if($oturum==true){  
    //Okuma işlemi  
    echo $_SESSION['adi'];  
}else  
    echo "SESSION Oturumu Başarısız";
```

Yukarıdaki kod daha önce kaydettiğimiz veriyi okuyup ekrana yazacaktır. Burada dikkat etmeniz gereken en önemli nokta mutlaka veriyi okumandan önce ilk satırdaki **session_start()** fonksiyonunu yazmanız gerektiğidir. O fonksiyonu yazmazsanız oturum dosyanız hazırlanamayacağı için hata mesajı alırsınız.

PHP'de SESSION Yönetimi

```
string session_id ([ string $id ] )
```

Session id fonksiyonu o anki oturum dosyasının id sini öğrenmemize olanak sağlar.

```
$oturum = session_start();  
if($oturum==true){  
    //İdisini yazdıralım  
    echo session_id ();  
}else  
    echo "SESSION Oturumu Başarısız";
```

PHP'de SESSION Yönetimi

Şimdide başka bir sayfadan okuma işlemini yapalım.

```
$oturum = session_start();  
if($oturum==true){  
    //Okuma işlemi  
    echo $_SESSION['adi'];  
}else  
    echo "SESSION Oturumu Başarısız";
```

Yukarıdaki kod daha önce kaydettiğimiz veriyi okuyup ekrana yazacaktır. Burada dikkat etmeniz gereken en önemli nokta mutlaka veriyi okumandan önce ilk satırdaki `session_start()` fonksiyonunu yazmanız gerektiğidir. O fonksiyonu yazmazsanız oturum dosyanız hazırlanamayacağı için hata mesajı alırsınız.

PHP'de SESSION Yönetimi

Sessionlarda değişken üzerinde veri saklama işlemleri Cookielerde olduğundan biraz daha kolaydır. Özellikle birden fazla değişkende değer tutacaksak yani bir dizi kullanacaksak Sessionlarda daha kolaydır.

Hatırlayacak olursanız, Cookielerde dizi saklamak için diziyi önce serialize ile string ifadeye çeviriyorduk ve öyle saklıyorduk. Daha sonra kullanmak istediğimizde de tekrar unserialize fonksiyonu ile diziye dönüştürüyorduk. Sessionlarda direk dizi olarak saklayabiliyoruz verileri bu sebeple ekstra fonksiyonlarla dönüştürme işlemine pek gerek kalmıyor.

PHP'de SESSION Yönetimi

Session oluşturma dosyası

```
<?php
```

```
$ses = Session_start();
```

```
if($ses)
```

```
{
```

```
    echo "Oturum Başlatıldı";
```

```
    $_SESSION["mutlu"]= array("adi"=>"Mutlu YAPICI", "yas"=> 31, "meslek"=>
```

```
"Akademisyen");
```

```
}else
```

```
    echo "Oturum Başlatılamadı";
```

```
?>
```

```
<a href="sesoku.php">Session Dosyasını Oku</a>
```

```
<a href="sessil.php">Session Dosyasını Sil</a>
```

PHP'de SESSION Yönetimi

Session okuma dosyası

```
<?php
```

```
$ses = Session_start();  
if( isset( $_SESSION["mutlu"]))  
{  
    print_r( $_SESSION);  
    echo "<br>Adınız = ".$_SESSION["mutlu"]["adi"];  
}else  
    echo "Session Dosyası Bulunamadı";
```

```
?>
```

```
<a href="sesoku.php">Session Dosyasını Oku</a>
```

```
<a href="sessil.php">Session Dosyasını Sil</a>
```


PHP'de SESSION Yönetimi

Session silme dosyası

```
<?php
```

```
$ses = Session_start();  
unset($_SESSION["mutlu"]);
```

```
?>
```

```
<a href="sesoku.php">Session Dosyasını Oku</a>
```

```
<a href="sessil.php">Session Dosyasını Sil</a>
```

Session dosyasını tamamen sunucudan silmek istersek **Session_destroy()**; fonksiyonunu kullanabiliriz. Unset() fonksiyonu session dosyasındaki sadece ilgili değişkeni ve değerini siler. Örneğin bu örneğimizde biz session dosyasına **array("adi"=>"Mutlu YAPICI", "yas"=> 31, "meslek"=> "Akademisyen");** dizisi şeklinde bir sürü veri girdik. Unset fonksiyonu ile istediğimiz veri değerini silip diğerlerinin dosyada kalmasını sağlayabiliriz. Yani unset ile dosya tamamen silinmiş olmuyor.



ÖNEMLİ NOT : SESSION oturum dosyalarının COOKIE dosyalarından bir farkıda zaman kısıtlamalarının olmayışıdır. COOKIE dosyalarına 2 saat sonra otomatik silin gibi bir zaman sınırlaması koyabiliyorduk ama SESSION larda bu yok. Ya session_destroy() ile yok edeceğiz yada tüm SESSION dosyaları tarayıcı tamamen kapatıldığında otomatik silinir

PHP'de SESSION Yönetimi

Şimdide bir üye giriş ve çıkış, yani oturum işlemlerini yapabildiğimiz bir örnekle SESSION kullanımını anlatalım. Bu örnekte daha önceki Web Programlama I dersinde anlattığımız veritabanı işlemlerini de kullanacağız. Veritabanı oluşturup öncelikle üye bilgileri kaydedeceğiz. Daha sonra giriş sayfasında oturumu açmadan önce aldığımız verileri veritabanından kontrol edeceğiz eğer üye kayıtlı ise oturumu açtıracaktır.

Şimdi öncelikle aşağıdaki gibi 'ogrenci' isimli tabloyu 'okul' isimli bir veritabanına oluşturalım ve bir iki tanede öğrenci verisi girelim.

+ Seçenekler

		id	numara	adi	sifre	bolum
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	1	1234567	Mutlu YAPICI	1234	Bilgisayar Teknolojileri
<input type="checkbox"/>	 Düzenle  Kopyala  Sil	2	3456678	Ayşe ŞEN	mutlu	Elektronik ve Otomasyon sistemleri

Tümünü Seç Seçilileri:  Değiştir  Sil  Dışa Aktar

Göster : Başlangıç satırı: Satır sayısı: Her satırda bir başlıklar

PHP'de SESSION Yönetimi

Üye giriş ekranımız aşağıdaki gibi olacaktır. Dosya ismimiz uyegiris.php

```
<form action="giris.php" method="POST">
  <table width="400" cellpadding="10">
  <tr>
    <td colspan="2" align="center">
      <h3>ÜYE GİRİŞ SAYFASI</h3>
    </td>
  </tr>
  <tr>
    <th>Kullanıcı Adınız </th>
    <td align="right">
      <input type="text" name="kadi" id="kadi" placeholder="Kullanıcı Adınız"/>
    </td>
  </tr>
  <tr>
    <th>Kullanıcı Şifreniz </th>
    <td align="right">
      <input type="password" name="ksfr" id="ksfr" placeholder="Kullanıcı Şifreniz"/>
    </td>
  </tr>
  <tr>
    <td colspan="2" align="right"> <button name="btn"> Giriş Yap</button> </td>
  </tr>
</form>
```

ÜYE GİRİŞ SAYFASI

Kullanıcı Adınız

Kullanıcı Şifreniz

SESSION Giriş Sayfası

Şimdide verilerin veritabanından alınıp kontrol edilerek giriş işlemlerinin yapıldığı `giris.php` sayfasını yapalım.

```
<?php
//önce veriler geliyormu kontrol edelim
if(!empty($_POST['kadi']) && !empty($_POST['ksfr']))
{
    //gerekli veriler geldiğine göre alıp değişkenlere yükleyelim
    $adi=$_POST['kadi'];    $sfr=$_POST['ksfr'];

    //veri tabanında böyle bir kullanıcı varmı bağlanıp bakalım.

    $vt=@new mysqli("localhost","root","usbw","okul");

    if($vt->connect_error)|
        die("<h3 style='color:red;'>Veritabanı Bağlantısı Başarısız</h3>");

    $sql="SELECT * FROM `ogrenci` WHERE `adi`=? AND `sifre`=?";
    $mysqlHazirla=$vt->prepare($sql);
    if($vt->error)
        die("<h3 style='color:red;'>SQL Sorgu Hatası</h3>");
```

SESSION Giriş Sayfası (devam)

Şimdide verilerin veritabanından alınıp kontrol edilerek giriş işlemlerinin yapıldığı **giris.php** sayfasını yapalım.

```
$mysqlHazirla->bind_param('ss', $adi, $sfr);
$mysqlHazirla->execute();
$sonuc=$mysqlHazirla->get_result();
$veriler=$sonuc->fetch_assoc();
    if($sonuc->num_rows>0)
    {
        echo "<h3 style='color:green;'>Üye Girişi Başarılı</h3>";

        //Eğer üye girişi başarılı ise Oturum İşlemlerini Başlatalım
        $ses=session_start();
        if($ses)
        {
            $_SESSION['OturumVerileri']=array("oturumDurumu"=>"Başarılı",
                "adi"=> $veriler['adi'],
                "numara"=> $veriler['numara'],
                "bolum"=> $veriler['bolum']
            );

            //giris başarılı ise anasayfaya yönlendirelim
            header("refresh:3; url=anasayfa.php");
            exit;
        }
    }
```

SESSION Giriş Sayfası (devam)

Şimdide verilerin veritabanından alınıp kontrol edilerek giriş işlemlerinin yapıldığı `giris.php` sayfasını yapalım.

```
}else  
    echo "<h3 style='color:red;'>Üye Girişi Başarısız</h3>";
```

```
$mysqlHazirla->free_result();
```

```
}else  
    echo "<h3 style='color:red;'>Üye Girişi İçin Gerekli Alanları Doldurun</h3>";
```

```
$vt->close();
```

```
header("refresh:5; url=uyegiris.php");
```

?>

SESSION Giriş Sayfası (devam)

Şimdi de üyegirişi başarılı olduğunda yönlendirileceğimiz anasayfa.php sayfasını yapalım. Bu sayfaya sadece üye girişi yapmış yani oturum açmış kişiler girebilecek, diğer kişiler otomatik olarak uyegiris.php sayfasına yönlendirilecekler.

Sayın Üyemiz Formumuza Hoş Geldiniz

[Çıkış Yapın](#)

```
<?php
//eğer üye girişi yapmışsa bu sayfaya girebilsin yoksa uyarı verip
// giriş sayfasına yönlendirelim
session_start();
if(!empty($_SESSION['OturumVerileri']))
    if($_SESSION['OturumVerileri']['oturumDurumu']=='Başarili')
    {

    ///Üye Girişi Başarılı İse Kullanıcıların Neler Yapabileceğini
    //Burada Tanımlayacağız.

    <h1> Sayın Üyemiz Formumuza Hoş Geldiniz </h1>
    <a href="cikis.php">Çıkış Yapın</a>

<?php
exit;
}

///üye Girişi yapmadıysa giriş sayfasına yönlendirelim
echo "<h3 style='color:red;'>Bu sayfaya Giriş Yetkiniz Yok. Lütfen Üye Girişi Yapın.</h3>";
header("refresh:3; url=uyegiris.php");
```

SESSION Giriş Sayfası (devam)

Şimdi de üye çıkışı yapabilmesi için çıkış yapın butonuna tıkladığında gideceği cikis.php sayfasını yapalım.

Sayın Üyemiz Formumuza Hoş Geldiniz

[Çıkış Yapın](#)

```
<?php
```

```
session_start();
```

```
session_destroy();
```

```
echo "<h3 style='color:green;'>Üye Çıkışı Başarılı. Yine Bekleriz</h3>";
```

```
header("refresh:3; url=uyegiris.php");
```

```
?>
```


SESSION İşlemleri

Bundan sonra yapmamız gerekense diğer sayfalarımızı oluşturup hangi sayfalara üye girişi yapanların hangi sayfalara ise herkesin erişebileceğini belirlemek olacaktır. Bunun için öncelik birden fazla sayfa yapmalıyız. Tabi bu sayfalar arasında gezinmek için bir de menü yapmalıyız ve bu menüyü her sayfanın baş tarafına yerleştirmeliyiz. Burada amaç tasarımdan çok bir web sayfasının arka plan kodlaması dediğimiz BACK END geliştirmek olduğu için sayfa tasarımına çok dikkat etmeyeceğim. Basitçe menüler yapıp back end kodlama kısmını anlatacağım.

Öncelikle Hakkımızda, ürünler, iletişim, sipariş ver gibi 4 sayfa daha yapalım. Hakkımızda Sayfası Aşağıdaki Gibi Görünecek.

Hakkımızda Sayfasına Hoş Geldiniz.

Hakkımızda bölümü için gerekli yazıyı buraya yazacağız. Ankara Üniversitesi Elmadağ Meslek Yüksekokulu Bilgisayar Teknolojileri Bölümü Bilgisayar Programcılığı Programı Uzaktan Eğitim Web Programlama II Dersi.

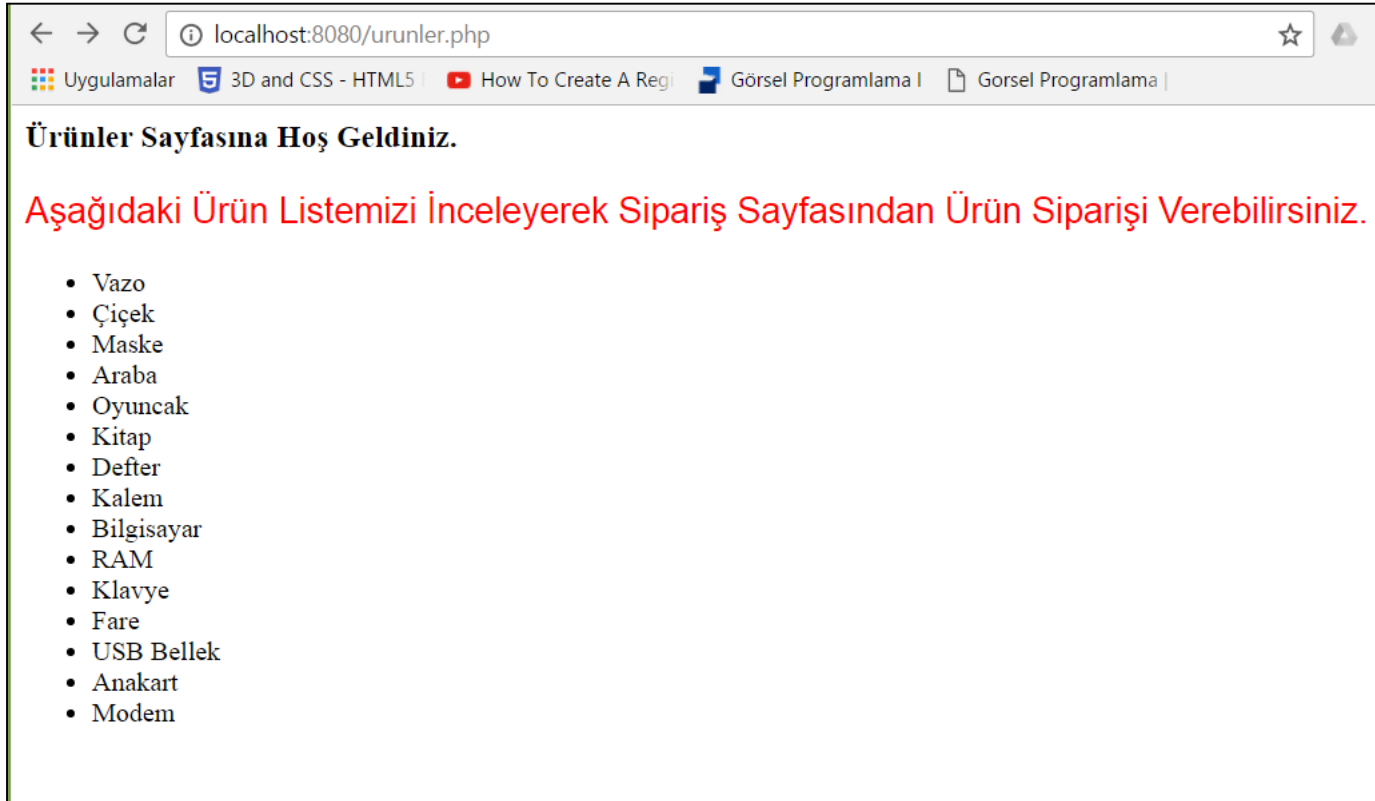
SESSION İşlemleri

Hakkımızda Sayfasına Ait Kodlar Aşağıdadır ve Sayfa Adı 'hakkimizda.php' dir.

```
<html>
  <head>
    <title>Hakkımızda Sayfası</title>
  </head>
  <body>
    <h3> Hakkımızda Sayfasına Hoş Geldiniz.</h3>
    <p style="font-family:arial;
      font-size:12pt;
      color:red;">
      Hakkımızda bölümü için gerekli yazıyı buraya yazacağız.
      Ankara Üniversitesi Elmadağ Meslek Yüksekokulu Bilgisayar
      Teknolojileri Bölümü Bilgisayar Programcılığı
      Programı Uzaktan Eğitim Web Programlama II Dersi.
    </p>
  </body>
</html>
```

SESSION İşlemleri

Ürün Listele Sayfası Aşağıdaki Gibi Görünecek.



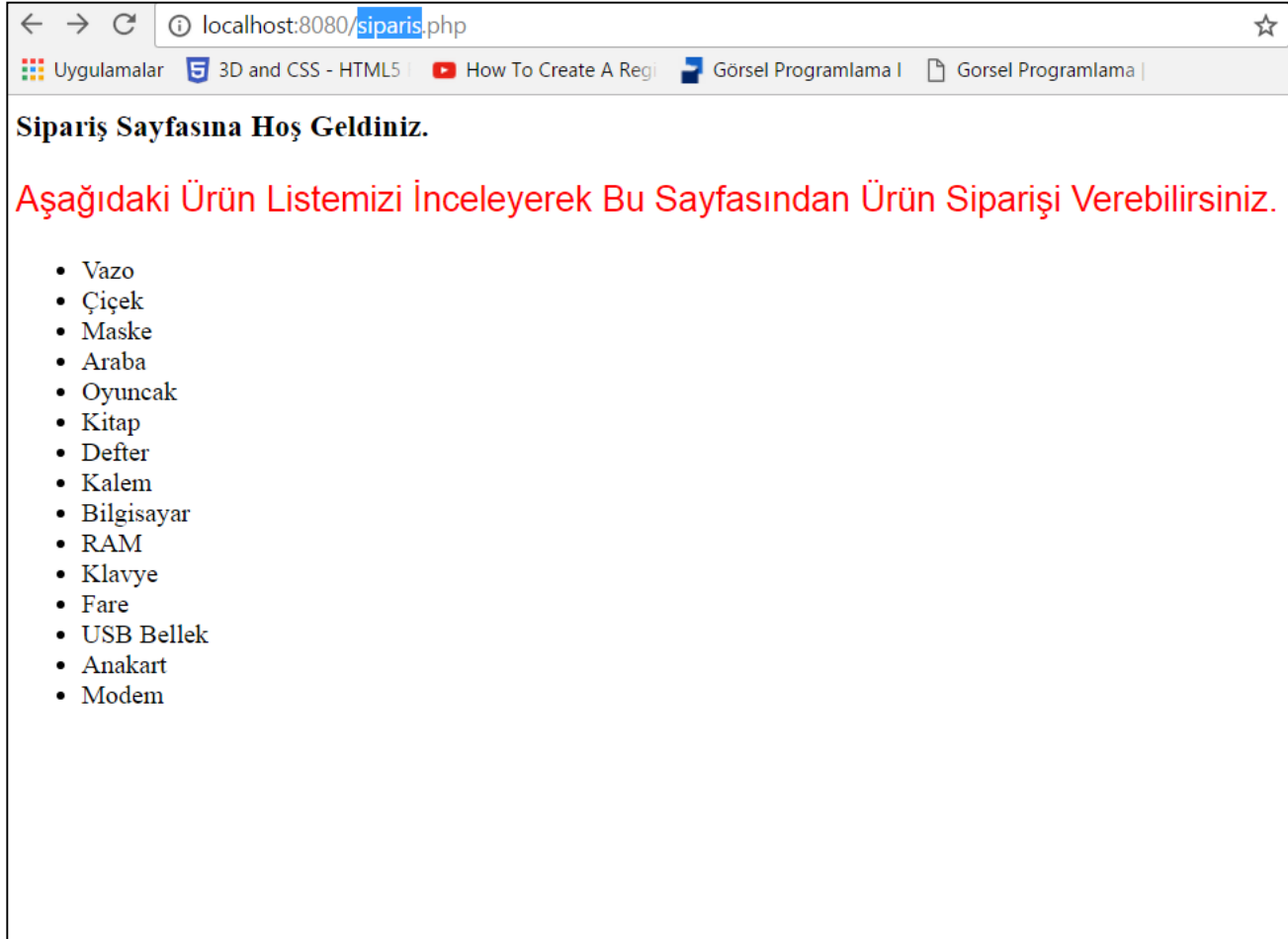
SESSION İşlemleri

Ürün Listele Sayfasına Ait Kodlar Aşağıdadır ve Sayfa Adı 'urunler.php' dir.

```
<html>
  <head>
    <title>Ürünler Sayfası</title>
  </head>
  <body>
    <h3> Ürünler Sayfasına Hoş Geldiniz.</h3>
    <p style="font-family:arial;
      font-size:16pt;
      color:red;">
      Aşağıdaki Ürün Listemizi İnceleyerek Sipariş Sayfasından Ürün Siparişi Verebilirsiniz.
    </p>
    <ul style="disc">
      <li>Vazo</li>
      <li>Çiçek</li>
      <li>Maske</li>
      <li>Araba</li>
      <li>Oyuncak</li>
      <li>Kitap</li>
      <li>Defter</li>
      <li>Kalem</li>
      <li>Bilgisayar</li>
      <li>RAM</li>
      <li>Klavye</li>
      <li>Fare</li>
      <li>USB Bellek</li>
      <li>Anakart</li>
      <li>Modem</li>
    </ul>
  </body>
</html>
```

SESSION İşlemleri

Ürün Sipariş Sayfası Aşağıdaki Gibi Görünecek.



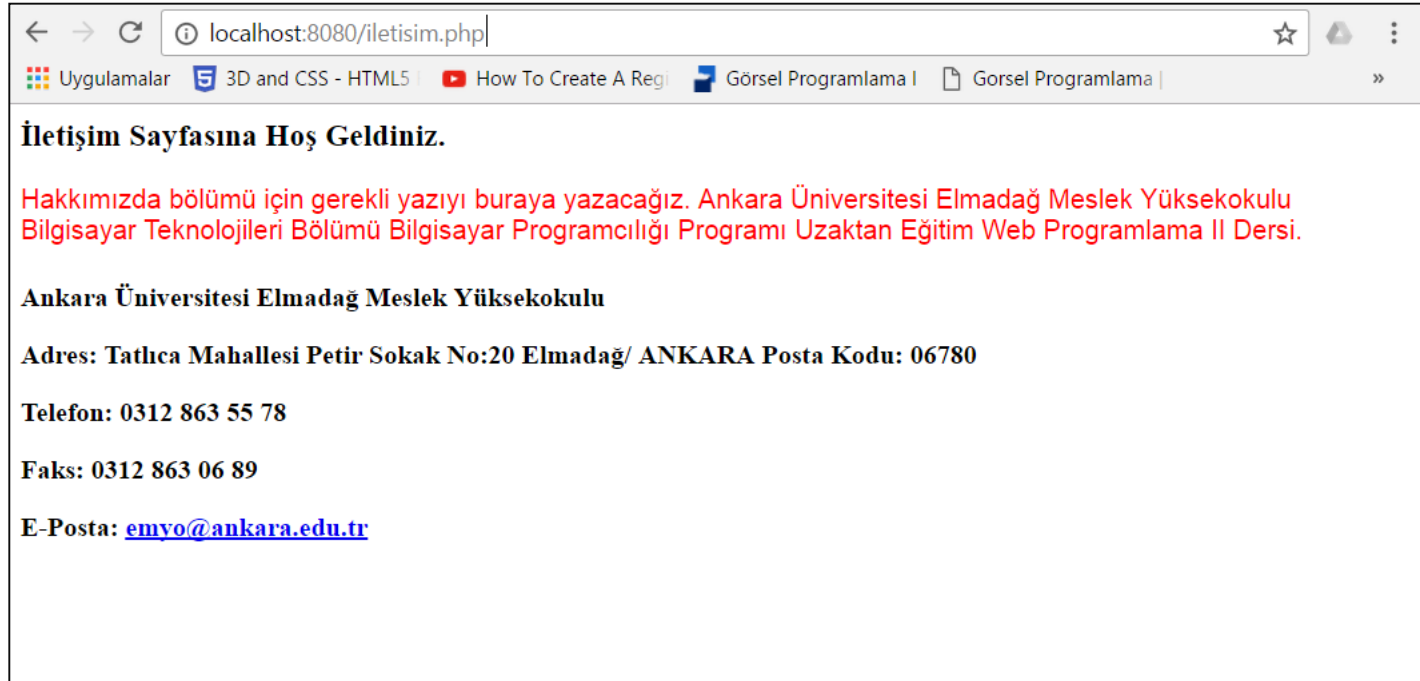
SESSION İşlemleri

Ürün Siparişi Sayfasına Ait Kodlar Aşağıdadır ve Sayfa Adı 'siparis.php' dir.

```
<html>
  <head>
    <title>Sipariş Sayfası</title>
  </head>
  <body>
    <h3> Sipariş Sayfasına Hoş Geldiniz.</h3>
    <p style="font-family:arial;
      font-size:16pt;
      color:red;">
      Aşağıdaki Ürün Listemizi İnceleyerek Bu Sayfasından Ürün Siparişi Verebilirsiniz.
    </p>
    <ul style="disc">
      <li>Vazo</li>
      <li>Çiçek</li>
      <li>Maske</li>
      <li>Araba</li>
      <li>Oyuncak</li>
      <li>Kitap</li>
      <li>Defter</li>
      <li>Kalem</li>
      <li>Bilgisayar</li>
      <li>RAM</li>
      <li>Klavye</li>
      <li>Fare</li>
      <li>USB Bellek</li>
      <li>Anakart</li>
      <li>Modem</li>
    </ul>
  </body>
</html>
```

SESSION İşlemleri

İletişim Sayfası Aşağıdaki Gibi Görünecek.



SESSION İşlemleri

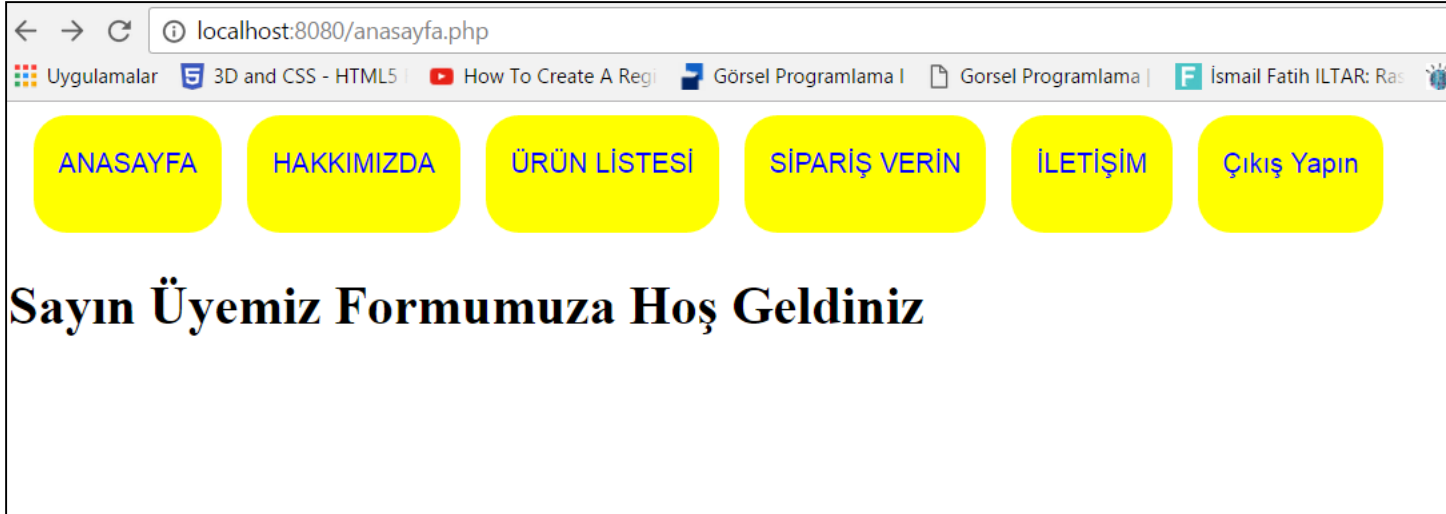
İletişim Sayfasına Ait Kodlar Aşağıdadır ve Sayfa Adı 'iletisim.php' dir.

```
<html>
  <head>
    <title>İletişim Sayfası</title>
  </head>
  <body>
    <h3> İletişim Sayfasına Hoş Geldiniz.</h3>
    <p style="font-family:arial;
      font-size:12pt;
      color:red;">
      Hakkımızda bölümü için gerekli yazıyı buraya yazacağız. Ankara Üniversitesi
      Elmadağ Meslek Yüksekokulu Bilgisayar Teknolojileri Bölümü Bilgisayar Programcılığı
      Programı Uzaktan Eğitim Web Programlama II Dersi.
    </p>

    <h4> <div class="post-content"> <!-- confines heading font to this content -->
    <p>Ankara Üniversitesi Elmadağ Meslek Yüksekokulu</p>
<p>Adres: Tatlıca Mahallesi Petir Sokak No:20 Elmadağ/ ANKARA      Posta Kodu: 06780</p>
<p>Telefon: 0312 863 55 78</p>
<p>Faks: 0312 863 06 89</p>
<p>E-Posta: <a href="mailto:emyo@ankara.edu.tr">emyo@ankara.edu.tr</a></p>
</h4>
  </body>
</html>
```


SESSION İşlemleri

Sayfalarımız hazır ama bu sayfalar arasında geçi yapabileceğimiz bi menü yok şimdi bir menü yapıp her sayfanın üst kısmına yerleştirelim. Öncelikle anasayfa ile başlayacağız. Anasayfayı daha önce kodlamıştık ve bu sayfanın içeriğini sadece üye girişi yapanlar görebiliyordu. Öncelikle üye girişi yaptım ve sayfanın üst kısmına aşağıdaki gibi menü kodlarını ekledim. Anasayfanın son hali ;



SESSION İşlemleri

Anasayfa eklediğimiz menü kodları; Bu kodları var olan ana sayfanın üst kısmına ekledim.

```
<style>
```

```
a{ font-family:arial; font-size:12pt; color:blue; text-decoration:none; transition:all 0.3s; }
```

```
div{ background-color:yellow; float:left; width:auto; height:50px; padding:20px 15px 0px 15px;  
border-radius:20px; margin-left:15px; transition:all 0.3s; }
```

```
div: hover{ transform: rotate(110deg); }
```

```
a: hover{ font-size:16pt; color:red; }
```

```
a: active{ font-size:14pt; color:yellow; }
```

```
</style>
```

```
<div class="menu"> <a href="anasayfa.php">ANASAYFA</a> </div>
```

```
<div> <a href="hakkimizda.php">HAKKIMIZDA</a> </div>
```

```
<div> <a href="urunler.php">ÜRÜN LİSTESİ</a> </div>
```

```
<div> <a href="siparis.php">SİPARİŞ VERİN</a> </div>
```

```
<div> <a href="iletisim.php">İLETİŞİM</a> </div>
```

```
<div> <a href="cikis.php">Çıkış Yapın</a> </div>
```

```
<br><br> <br><br>
```

```
{  
    //Üye Girişi Başarılı İse Kullanıcıların Neler Yapabileceğini  
    //Burada Tanımlayacağız.  
>  
    <h1> Sayın Üyemiz Formumuza Hoş Geldiniz </h1>  
    <a href="cikis.php">Çıkış Yapın</a>  
<?php  
    exit;  
}
```

SESSION İşlemleri

Evet menüyü ana sayfaya ekledik ama daha ekleyeceğim dört sayfa daha var tüm menü kodlarını o sayfalarda eklemesek, anasayfadan diğer sayfalara geçiş yapabiliriz, ama diğer sayfalarda menü olmadığı için bir birleri arasında geçiş yapamayız. Bu nedenle menüyü tüm sayfalara ekleyeceğiz. Menüyü tüm sayfalara kopyala yapıştır yapmak yerine bir önceki dönem anlattığım gibi yeni bir menü sayfası oluşturup o sayfayı tüm menülere yüklersek daha etkili olacaktır. Böylece menüde yapacağımız her değişiklik tüm sayfalara yansıtacak ve tek tek değiştirmek zorunda kalmayacağız. Bunun için menu.php adında bir sayfa oluşturup tüm menü kodlarını o sayfaya taşıyalım.

SESSION İşlemleri

Menü Sayfası;

```
<style>
a{ font-family:arial;
font-size:12pt;
color:blue;
text-decoration:none;
transition:all 0.3s;
}
div{background-color:yellow;
float:left;
width:auto;
height:50px;
padding:20px 15px 0px 15px;
border-radius:20px;
margin-left:15px;
transition:all 0.3s;}
div:hover{
transform: rotate(110deg);
}
a:hover{
font-size:16pt;
color:red;
}
a:active{
font-size:14pt;
color:yellow;
}
</style>
```

```
<div> <a href="anasayfa.php">ANASAYFA</a> </div>
<div> <a href="hakkimizda.php">HAKKIMIZDA</a> </div>
<div> <a href="urunler.php">ÜRÜN LİSTESİ</a> </div>
<div> <a href="siparis.php">SİPARİŞ VERİN</a> </div>
<div> <a href="iletisim.php">İLETİŞİM</a> </div>
<div> <a href="cikis.php">Çıkış Yapın</a> </div>
<br><br> <br><br>
```

SESSION İşlemleri

Menü Sayfasını tüm sayfalarımızın en üstüne aşağıdaki gibi yüklüyoruz.

```
Dosya Düzen Arama Görünüm Kodlama Dil Ayarlar Makro Çalıştır Eklentiler Pencere ?
deneme.html x hakkımızda.php x urunler.php x siparis.php x menu.php x anasayfa.php x
1 <?php
2     //menüyü bu sayfaya yüklüyoruz
3     include_once ("menu.php");
4 ?>
5 <html>
6     <head>
7         <title>Hakkımızda Sayfası</title>
8     </head>
9     <body>
10        <h3> Hakkımızda Sayfasına Hoş Geldiniz.</h3>
11        <p style="font-family:arial;
12            font-size:12pt;
13            color:red;">
14            Hakkımızda bölümü için gerekli yazıyı buraya yazacağız.
15            Ankara Üniversitesi Elmadağ Meslek Yüksekokulu Bilgisayar
16            Teknolojileri Bölümü Bilgisayar Programcılığı
17            Programı Uzaktan Eğitim Web Programlama II Dersi.
18        </p>
19    </body>
20 </html>
```

SESSION İşlemleri

Menü Sayfasını tüm sayfalarımızın en üstüne aşağıdaki gibi yüklüyoruz.

```
Dosya Düzen Arama Görünüm Kodlama Dil Ayarlar Makro Çalıştır Eklentiler Pencere ?
deneme.html x hakkımızda.php x urunler.php x siparis.php x iletisim.php x menu.php x anasayfa.php x
1 <?php
2 //menüyü bu sayfaya yüklüyoruz
3 include_once("menu.php");
4 ?>
5 <html>
6 <head>
7 <title>Hakkımızda Sayfası</title>
8 </head>
9 <body>
10 <h3> Hakkımızda Sayfasına Hoş Geldiniz.</h3>
11 <p style="font-family:arial;
12 font-size:12pt;
13 color:red;">
14 Hakkımızda bölümü için gerekli yazıyı buraya yazacağız.
15 Ankara Üniversitesi Elmadağ Meslek Yüksekokulu Bilgisayar
16 Teknolojileri Bölümü Bilgisayar Programcılığı
17 Programı Uzaktan Eğitim Web Programlama II Dersi.
18 </p>
19 </body>
20 </html>
```

SESSION İşlemleri

Evet projemizin bu haliyle sadece anasayfaya üye girişi yapma kısıtı eklemiş olduk. Diğer bütün sayfalara herkes giriş yapıp içeriğini görebiliyor. Şimdi istediğim ürün listeler ve sipariş sayfalarına da üye girişi yapmayanların girişini engellemek. Bunun için oturum açılmış mı diye kontrol şartını bu sayfalara ekleyeceğiz. Eğer Oturum açılmamışsa daha önce yaptığımız gibi giriş sayfasına yönlendirecek.

```
<?php
session_start(); ///önce oturumu hazırlıyoruz.
if(empty($_SESSION['OturumVerileri']))///oturum dosyası var mı bakıyoruz
{////oturum dosyası yoksa yani üye girişi yapılmadıysa
///kişiye uyarı verip giriş sayfasına yönlendiriyoruz.
echo "<h3 style='color:red;'>Bu sayfaya Giriş Yetkiniz Yok. Lütfen Üye Girişi Yapın.</h3>";
header("refresh:3; url=uyegiris.php");
exit;
}else{///oturum dosyası varsa üye hala aktif mi kodtrol edelim

    if($_SESSION['OturumVerileri']['oturumDurumu']!=='Basarili')
    {
        ////aktif değilse yine girişi sayfasına gönderiyoruz
        echo "<h3 style='color:red;'>Bu sayfaya Giriş Yetkiniz Yok. Lütfen Üye Girişi Yapın.</h3>";
        header("refresh:3; url=uyegiris.php");
        exit;
    }

}
```

```
//menüyü bu sayfaya yüklüyoruz
include_once("menu.php");
```

```
?>
```

```
<html>
```

SESSION İşlemleri

Bu kısmı sadece üye girişi yapanların görmesini istediğimiz her sayfaya ekleyerek sınırlama yapabiliriz.

```
<?php
session_start(); ///önce oturumu hazırlıyoruz.
if(empty($_SESSION['OturumVerileri']))///oturum dosyası var mı bakıyoruz
{/////oturum dosyası yoksa yani üye girişi yapılmadıysa
///kişiye uyarı verip giriş sayfasına yönlendiriyoruz.
echo "<h3 style='color:red;'>Bu sayfaya Giriş Yetkiniz Yok. Lütfen Üye Girişi Yapın.</h3>";
header("refresh:3; url=uyegiris.php");
exit;
}else{///oturum dosyası varsa üye hala aktif mi kontrol edelim

        if($_SESSION['OturumVerileri']['oturumDurumu']!= 'Basarili')
        {
            ///aktif değilse yine girişi sayfasına gönderiyoruz
            echo "<h3 style='color:red;'>Bu sayfaya Giriş Yetkiniz Yok. Lütfen Üye Girişi Yapın.</h3>";
            header("refresh:3; url=uyegiris.php");
            exit;
        }

    }

    ///menüyü bu sayfaya yüklüyoruz
include_once ("menu.php");
```

?>

<html>

SESSION İşlemleri

Sessionları kullanarak üye girişi işlemlerini yaptık peki ama bu projeyi bir adım daha ilerletip, sayfalara göre yetkilendirme işlemlerini nasıl yapabiliriz.

Örneğin; bir sayfaya Yönetici giriş yaptığında farklı menüler görsün, Moderatör giriş yaptığında farklı ve Normal üyeler giriş yaptığında farklı menüler görsün. Bunu nasıl yapabiliriz?



KAYNAKLAR

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

