

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 Ünitede 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 Cookie Yönetimi

Türkçede çerez de dediğimiz Cookie'ler web programlamada sayfamıza gelen kullanıcıların bilgilerini kendi bilgisayarları üzerinde tutmak istediğimizde sıklıkla kullanılmaktadır. Üye giriş işlemleri, online alışveriş işlemleri ve sepet uygulamaları gibi yerlerde sıklıkla kullanılmaktadır. SetCookie() fonksiyonu ile bir bilgiyi çerez dosyaları içerisine kaydedebilir ve daha sonra oradan okuyarak kullanabiliriz. Cookie kullanımında dikkat edilmesi gereken en önemli nokta bu dosyalar kullanıcının kendi bilgisayarında tutulduğu ve ufak bir kod yardımı ile istenilen internet sayfasından kolayca ulaşılabildiği için şifre gibi önemli bilgiler bu dosyalarda tutulmamalıdır. Fonksiyonun kullanım şekli;

Bool Setcookie (*degisken adı, degeri, zaman ayarı, erişim yolu, erişecek adres, https, httponly*);

- **Değişken adı** : Çereze kaydedilecek değişkenin ismi
- **Değeri** : Değişkenin değeri. Burada şifre gibi güvenlik önemi olan değerler saklanmamalı.
- **Zaman ayarı** : Zaman aşımını kontrol eden değerdir. oturumun hangi süre içerisinde açık kalacağını belirler.
- **Erişim yolu** : Bu parametre bize server üzerinde hangi kısımlarda oturumun etkili olacağını belirtir. Üyelerin sitedeki belli bir alana girmesine izin veriyorsak dizini belirtmeliyiz. Aksi taktirde '/' yazarak tüm sitede geçerli kılabiliriz.
- **Erişim yolu** : Çerezin kullanılabilceği site adını tanımlar. Sitenin subdomainli alt sitelerinde oturumun geçerli olmasını bu şekilde sağlarız.
- **https** : Çerezin güvenli bağlantı üzerinden kullanıcıya veri aktarımını istiyorsak bu değere *TRUE* yazmalıyız. Aksi taktirde bu değer *FALSE* döner.
- **Httponly** : Bu alan *TRUE* değer alırsa veri sadece http üzerinden aktarılır. Yani çereze JS gibi script dilleri ile erişim gerçekleşmez. Bu XSS saldırılarında kimlik hırsızlığı riskini azaltmaya yöneliktir. bkz([XSS](#))

PHP'de Cookie Yönetimi

//Çerezlerin kaydedildiği sayfa içi cookieKayit.php

```
Setcookie ("adi", "M. Mutlu YAPICI");  
echo "<a href='cookieOku.php'>Çerezi Okumak İçin Tıklayın</a>";
```

[Çerezi Okumak İçin Tıklayın](#)

//Çerezlerin okunacağı sayfa içi cookieOku.php

```
if( isset($_COOKIE['adi']))  
{  
    echo "<h2>Hoş Geldin {$_COOKIE['adi']}"; //çerezin değerini okuyoruz  
}else  
    echo "Kayıtlı Çerez Bulunamadı";
```

Hoş Geldin M. Mutlu YAPICI

Yukarıdaki kodları incelediğimizde ilk sayfada 'adi' değişkenine değer olarak 'M. Mutlu YAPICI' verisi kaydediliyor. Daha sonra bağlantıya tıklayarak gittiğimiz cerezoku.php sayfasında `$_COOKIE['adi']` komutu ile çerez içindeki 'adi' değişkeninin değeri okunup ekrana yazdırılıyor. Bu örneği bir de zaman sınırı koyarak deneyelim 10 saniye içinde okumak istersek okunsun yoksa çerez silinsin istiyoruz. Bunun için setcookie fonksiyonuna zaman kısıtını da ekleyeceğiz. Öncelikle time() fonksiyonu ile o anki zamanı alıp üzerine 10 saniye ekleyip çereze gönderelim.

//Çerezlerin kaydedildiği sayfa içi cookieKayit.php

```
$zaman = time()+10;  
Setcookie ("adi", "M. Mutlu YAPICI" , $zaman);
```

```
echo "<a href='cookieOku.php'>Çerezi Okumak İçin Tıklayın</a>";
```

Eğer 10 saniyeden sonra girerseniz

Kayıtlı Çerez Bulunamadı

PHP'de Cookie Yönetimi

Bir çerezin içerisine istersek birden fazla veride yükleyebiliriz. Bunun için tahmin edebileceğiniz gibi DİZİ leri kullanacağız. Çerezlerde dizileri kullanmanın iki yolu vardır.

Birinci yolu çereze yükleyeceğimiz dizinin içeriğini **SERIALIZE()** PHP fonksiyonu ile bir string değişkeni haline getirip çereze normal bir değişkenmiş gibi yüklemek. Tabi bu şekilde yaparsak çerezden okuduğumuzda da yine o veriyi string halinden dizi haline dönüştürüp kullanmamız gerekir. Bunun içinde **UNSERIALIZE()** PHP fonksiyonunu kullanırız.

İkinci yolu ise dizi içerisine yazdığımız değişkeni sanki bir diziymiş gibi tanımlamalıyız, ancak tırnak içerisinde tanımlamalıyız. Evet bu biraz garip bir yöntem, açıkçası çerezlerde dizi kullanımı olmadığı için bir anlamda bu yöntemle çerezi kandırmış gibi oluyoruz.

Bu iki yöntemi de örnekler üzerinde görelim şimdi.

```
$suan = time();
```

```
//birinci yöntemle önce yükleyeceğimiz verileri bir diziye atalım
```

```
    $dizi= array('adi'=>'Mutlu YAPICI','yas'=>31,'meslek'=>'Akademisyen');
```

```
// şimdi bu dizi içindeki verileri birleştirip tek bir değişken haline getirelim
```

```
    $veri= Serialize($dizi);
```

```
//Çereze yükleyelim
```

```
    $oldumu= setcookie("Bilgilerim", $veri, ($suan+100));
```

```
    if($oldumu === true)      echo "Çerez Oluşturuldu";
```

```
    else          echo "Çerez Oluşmadı";
```

PHP'de Cookie Yönetimi

Birinci yöntemin çerez üzerine kaydedilmesini önceki sayfada yaptık şimdi de okuma işlemini yapalım

```
if( isset($_COOKIE['Bilgilerim']))
{
    echo "Okunan Çerez içi (STRING HALİ) =      {$_COOKIE['Bilgilerim']} <br>";
    //tekrar diziye dönüştürelim
    $d= unserialize($_COOKIE['Bilgilerim']);
        echo "Dizi Hali = ";
        print_r($d);
} else
    echo "Çerez oluşmamış ki";
```

Aşağıda kodun çıktısı görülmektedir. Yukarıda gördünüz gibi önce `isset` fonksiyonu ile 'Bilgilerim' adında bir çerez oluşmuş mu baktık. Eğer oluşmuşsa Verileri alıp `unserialize` fonksiyonu ile diziye dönüştürdük. Diziye dönüştürülmüş ve ve önceki halini de ekrana yazdırdık. Çıktısı aşağıdaki gibidir.

```
Okunan Çerez içi (STRING HALİ) = a:3:{s:3:"adi";s:12:"Mutlu YAPICI";s:3:"yas";i:31;s:6:"meslek";s:11:"Akademisyen";}
Dizi Hali = Array ( [adi] => Mutlu YAPICI [yas] => 31 [meslek] => Akademisyen )
```


PHP'de Cookie Yönetimi

Şimdi aynı örneği ikinci yöntemle yapalım. Bu yöntemde dediğimiz gibi çerezin içerisine vereceğimiz değişkenin adını dizi tanımlar gibi vereceğiz. Aslında çerez fonksiyonu onu normal bir string değişken adımı gibi alacak ama aslında biz yazım şekli itibarıyla dizi tanımlamış olacağız. :D Biraz karışık gibi.

```
$suan = time();
```

```
$oldumu = setcookie("mutlu[adi]", "Mutlu YAPICI", ($suan+100));
```

```
$oldumu = setcookie("mutlu[yas]", 31,($suan+100));
```

```
$oldumu = setcookie("mutlu[meslek]", "Akademisyen", ($suan+100));
```

```
if($oldumu === true)
```

```
    echo "Çerez Oluşturuldu";
```

```
else
```

```
    echo "Çerez Oluşmadı";
```

PHP'de Cookie Yönetimi

Okuma kısmı;

```
if( isset($_COOKIE['Mutlu']))
{
    print_r($_COOKIE);
    echo"<br> Adı = {$_COOKIE['mutlu']['adi']}";
} else
    echo "Çerez oluşmamış ki";
```

Aşağıda kodun çıktısı görülmektedir.











```
Array ( [mutlu] => Array ( [adi] => Mutlu YAPICI [yas] => 31 [meslek] => Akademisyen ) [_jsuid] => 1034301492
[___utma] => 111872281.1300213960.1480193959.1480193959.1480193959.1 [__utmz] =>
111872281.1480193960.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none) [_ga] => GA1.1.1300213960.1480193959 )
Adı = Mutlu YAPICI
```

PHP'de Cookie Yönetimi

Önceki son örnekte gördüğümüz gibi eğer cookie nin zamanını ayarlar ve o zamandan daha sonra erişmeye kalkarsanız çerezin kendisi silinirse de içindeki bilgiler silinir. Bu yöntemi her hangi bir cookie yi silmek için de kullanabilirsiniz. Cookie zamanını olduğunuz zamandan daha önceye atarsanız çerez silinmiş olur. **Eğer bir zaman vermezseniz tarayıcı kapatılana kadar çerezler saklanır.**

Şimdi bir sepet uygulaması yapalım. Basit olarak bir veritabanımızda birkaç ürün olacak ve bu ürünlere ait bilgiler bulunacak. Ürünler sayfamızda bu ürünler listelenecek ve altlarındaki sepete ekle butonuna tıklayınca, sepetimize eklenecek. Sepet bilgilerini ise cookie içerisinde tutacağız. Veritabanınız aşağıdaki gibi olabilir.

+ Seçenekler

					id	adi	aciklama	fiyat	kdv_orani	kargo	stok	resim		
<input type="checkbox"/>		Düzenle		Kopyala		Sil	1	Kalem	Yumuşak Kurşun Kalem	2	8	3	100	resim/kursunkalem.jpg
<input type="checkbox"/>		Düzenle		Kopyala		Sil	2	Kalem	Tükenmez Kalem	3	8	3	50	resim/tukenmezkalem.jpg
<input type="checkbox"/>		Düzenle		Kopyala		Sil	3	Silgi	Kaliteli Yumuşak Silgi	2	5	3	200	resim/silgi.jpg
<input type="checkbox"/>		Düzenle		Kopyala		Sil	4	Kitap	TEOG Test Kitabı	25	18	3	150	resim/testkitabi.jpg
<input type="checkbox"/>		Düzenle		Kopyala		Sil	5	Defter	5 orta çizgili harita metod	10	18	3	200	resim/defter.jpg

PHP'de Cookie Yönetimi

Siparis.php dosyamızın içeriği;

```
<?php ///ürünlerimizi öncelikle veritabanından çekip listeleyelim
```

```
$vt=new mysqli("localhost","root","usbw","satis");
```

```
if(!$vt)
```

```
die("Veritabanı sunucu bağlantı hatası".$vt->error());
```

```
if(isset($_GET)) ///eğer veri geliyorsa
```

```
{
```

```
if(!empty($_GET['gorev']) && !empty($_GET['urunid']))
```

```
{
```

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

```
///önce daha önceden ürün varmı çerezleri alalım
```

```
if(isset($_COOKIE['sepet']))
```

```
{
```

```
$sepetDizisi=unserialize($_COOKIE['sepet']);
```

PHP'de Cookie Yönetimi

Siparis.php dosyamızın içeriği; (DEVAMI)

```
        }else
            $sepetDizisi="";

switch($gorev)
{
    case"urun_ekle"://ürünü eklemek için ürüne ait idyi alıp
//önce bu ürün daha önceden eklenmimi bakalım eklenmişse sadece adetini artıralım

                if(isset($sepetDizisi[$urunid]))
                    {
                        $sepetDizisi[$urunid]['adet']++;
                    }else{
//cereze yeni ürün ekleyelim

                        //önce ürüne ait bilgileri alalım
                        $sql="SELECT * FROM `urunler` WHERE
`id`=$urunid LIMIT 1";

                        $sorgu=$vt->prepare($sql);
                        $sorgu->execute();
                        $sonuc=$sorgu->get_result();
                        $v=$sonuc->fetch_assoc();
```

PHP'de Cookie Yönetimi

Siparis.php dosyamızın içeriği;(DEVAMI)

```
$sepetDizisi[$v['id']]=array('id'=>$v['id'],'adi'=>$v['adi'],'fiyat'=>$v['fiyat'],'adet'=>1);
                                $sorgu->free_result();

    $sorgu->close();
        }

        break;
    case"urun_sil":
        ///ürün silmek te eklemekle aynı mantıkla eğer ürün adedi 1 den fazla ise 1 azalt değilse sil
        if(isset($sepetDizisi[$urunid]))////////ürün varmı diye bakıyoruz
            {
                if($sepetDizisi[$urunid]['adet']>1)
                    $sepetDizisi[$urunid]['adet']--;
                else
                    unset($sepetDizisi[$urunid]);
            }

        break;
    }
}
```

PHP'de Cookie Yönetimi

Siparis.php dosyamızın içeriği;(DEVAMI)

```
        //İşimdi cerezin son halini kaydedelim
        echo (setcookie("sepet",serialize($sepetDizisi))?"Ürün Eklendi":"Ürün Eklenemedi" ;
    }
}
$sql="SELECT * FROM `urunler`";
$vt->set_charset("utf8");
$sorgu=$vt->prepare($sql);
if($sorgu)
{
    $sorgu->execute();
    $veri=$sorgu->get_result();
    echo "        <section> ";
    while($a=$veri->fetch_assoc())
    {
        echo " <style>
            table: hover{
background-color:RGBA(250,200,200,0.5);
transition:background 0.5s;
            }
    }
}
```

PHP'de Cookie Yönetimi

Siparis.php dosyamızın içeriği;(DEVAMI)

```
table{
    border:2px solid lightblue;
    border-radius:15px;
    margin-top:15px;
}
section{float:left; margin-bottom:15px;}
th{background-color:#BBBBBB;}
.sabit{position:fixed; width:40%;height:100%;right:20px;}
</style>
<table border='3' rules='none' cellpadding='10px' width='500px;'>
    <tr>
        <td rowspan='3' width='110px'>
            <img src='{$a['resim']}' width='100px' height='100px' />
        </td>
        <td>
            <h3>{$a['adi']}</h3>
            <p>{$a['aciklama']}</p>
        </td>
    </tr>
</table>
```


PHP'de Cookie Yönetimi

Siparis.php dosyamızın içeriği;(DEVAMI)

```
<td align='right'>
    <p align='right'>{$a['fiyat']} TL </p>
    <a
href='siparis.php?gorev=urun_ekle&urunid={$a['id']}' <button>Sepete Ekle</button></a>
    </td>
</tr>
</table> ";
}
echo" </section>";
    $sorgu->free_result();
    $sorgu->close();
}

$vt->close();

?>
<div class='sabit'>
<iframe src="sepet.php"scrolling="no" style="width:100%;height:100%;border:none;"/>
</div>
```

PHP'de Cookie Yönetimi

Şimdide sepet.php dosyamızın içeriğini yazalım ;

```
<style>
    table{
        border:2px solid lightblue;
        border-radius:15px;
        margin-top:15px;
        }
    th{background-color:#BBBBBB;}
</style>
<?php //sepetsayfamız
//şimdi sepet kısmını yapalım sepete ait verilerin tamamı cookie dosyasından gelecek
///önce daha önceden ürün varmı çerezleri alalım
    if(isset($_COOKIE['sepet']))
    {
        $sepetDizisi=unserialize($_COOKIE['sepet']);
    }else
        $sepetDizisi=array("Sepet Boş");
//print_r( $sepetDizisi);
$metin=" <section class='sabit'>
    <table border='3' rules='none' cellpadding='10px' width='500px;'>
```

PHP'de Cookie Yönetimi

Şimdide sepet.php dosyamızın içeriğini yazalım ;(DEVAMI)

```
<tr>
    <th>Ürün Adı</th><th>Fiyatı</th><th>Ürün Adedi</th><th>Sil</th> </tr>";
$stop=0;
foreach($sepetDizisi as $urn)
    {
        $stop+=$( $urn['adet']*$urn['fiyat'] );
        $metin.=" <tr>
            <td > <p>{$urn['adi']}</p> </td>
            <td> <p align='right'>{$urn['fiyat']} TL</p> </td>
            <td align='right'> <p>{$urn['adet']} ADET</p> </td>
            <td align='right'>
                <a href='siparis.php?gorev=urun_sil&urunid={$urn['id']}' target='_parent'> <button>Ürünü Sil</button></a>
            </td>
        </tr> ";
    }
    echo" $metin <tr>
    <td colspan='5' align='right' style='padding-right:40px; font-weight:bold'>Toplam Tutar : $stop TL </td>
    </tr>
    </table>
</section> ";
?>
```

PHP'de Cookie Yönetimi

Sayfamızın
son hali;


Ürün Eklendi



Kalem
Yumuşak Kurşun Kalem

2.00 TL


[Sepete Ekle](#)



Kalem
Tükenmez Kalem

3.00 TL


[Sepete Ekle](#)



Silgi
Kaliteli Yumuşak Silgi

2.00 TL


[Sepete Ekle](#)



Kitap
TEOG Test Kitabı

25.00 TL

[Sepete Ekle](#)



Defter
5 orta çizgili harita metod

10.00 TL

[Sepete Ekle](#)







Ürün Adı	Fiyatı	Ürün Adedi	Sil
Kalem	3.00 TL	7 ADET	Ürünü Sil
Kitap	25.00 TL	1 ADET	Ürünü Sil
Kalem	2.00 TL	1 ADET	Ürünü Sil
Toplam Tutar : 48 TL			

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.

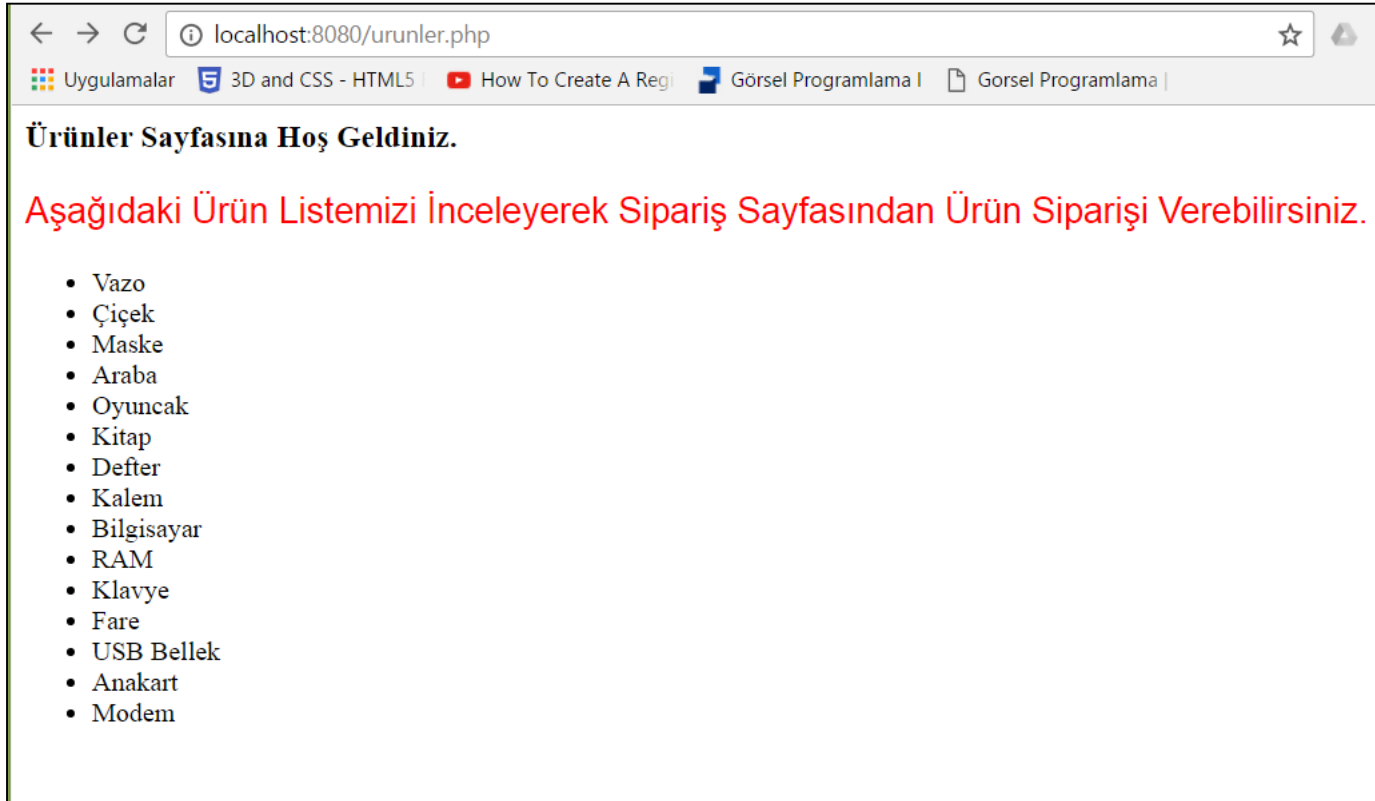
SESSION İşlemleri

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.



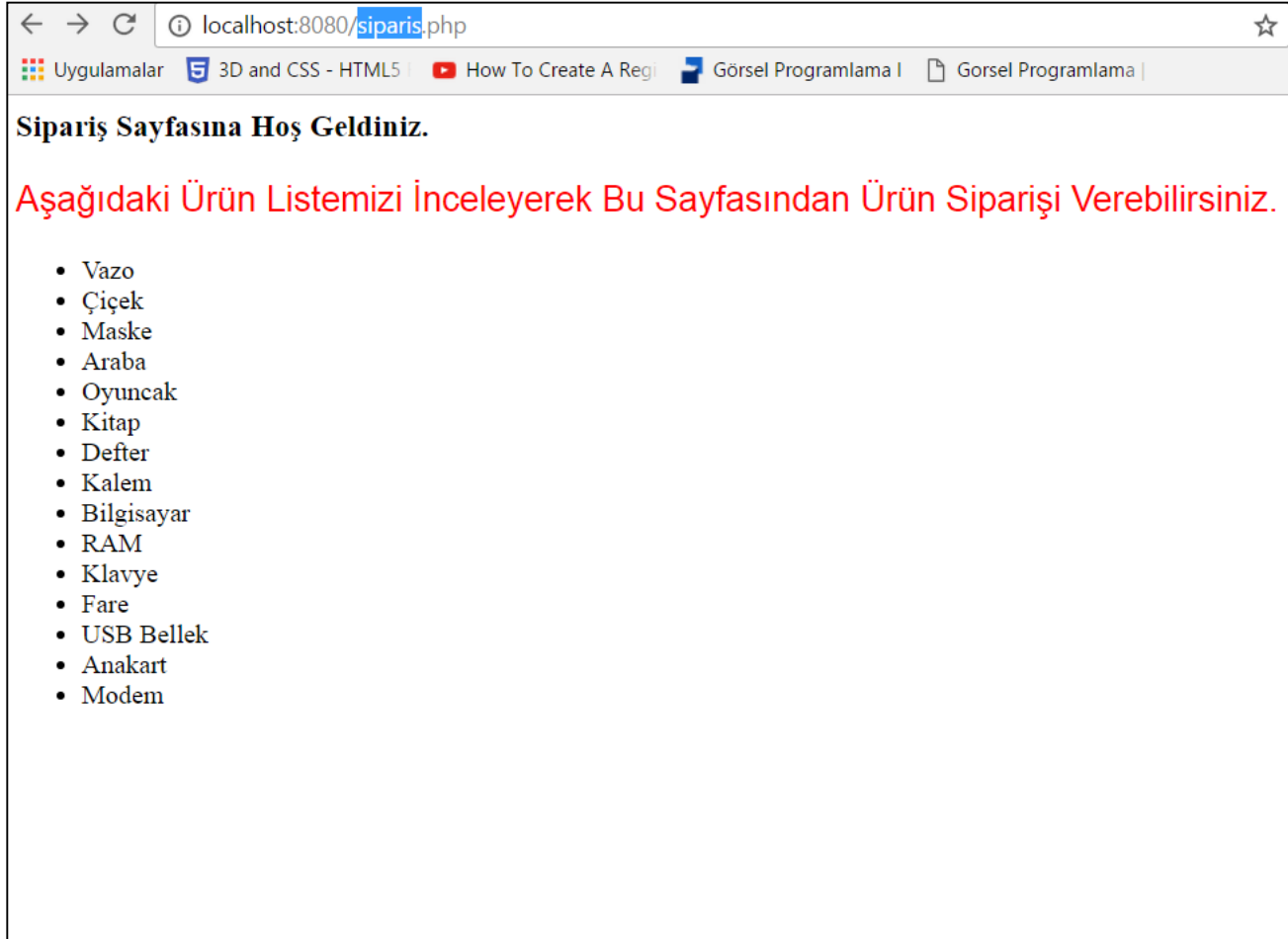
SESSION İşlemleri

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.



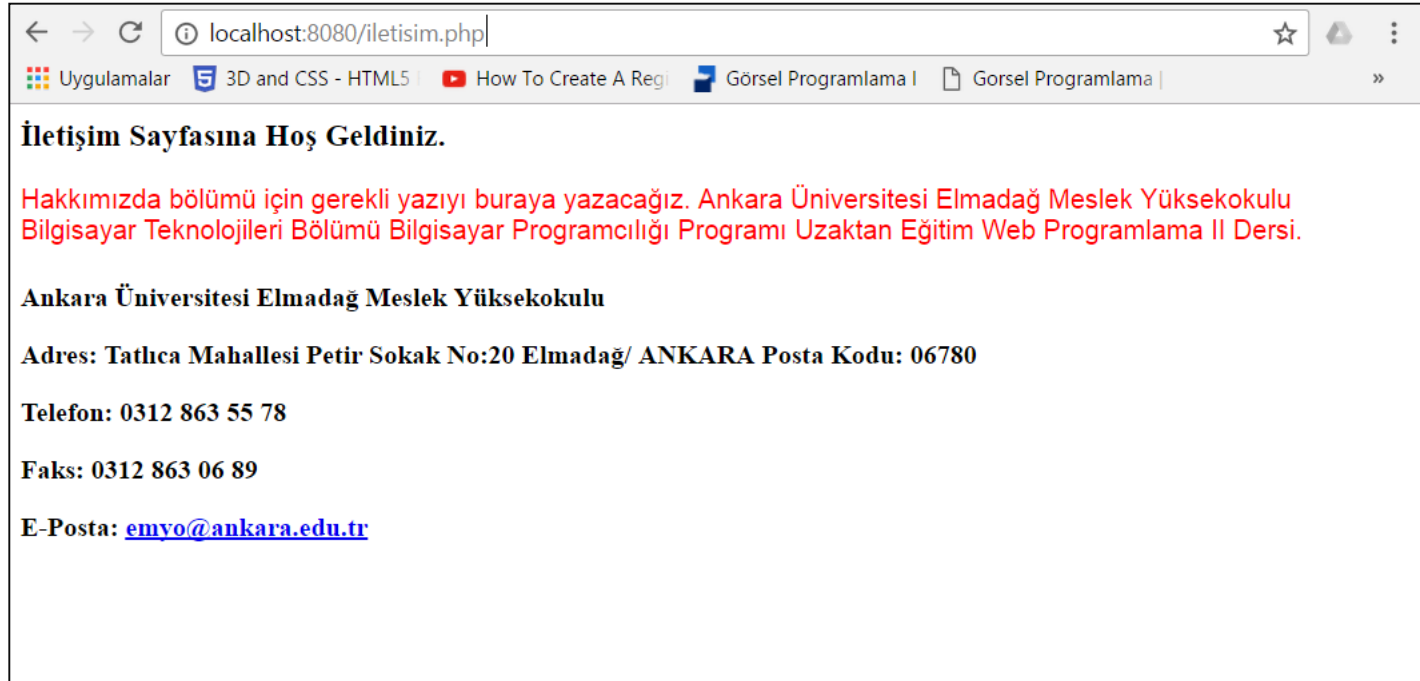
SESSION İşlemleri

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

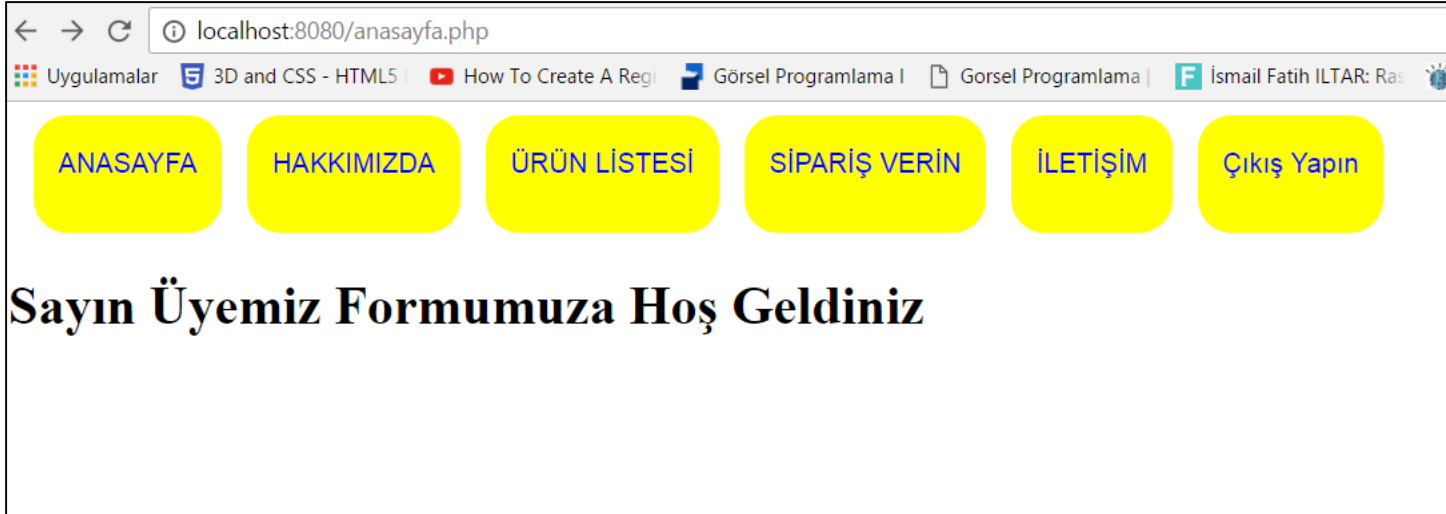
İletişim Sayfasına Ait Kodlar Aşağıdadır ve Sayfa Adı 'iletisim.php' dir.

```
<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 eklemezsek, 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 yansiyacak 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 ürünler.php siparis.php iletisim.php menu.php anasayfa.php
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?



Sayfalama Ve Arama İşlemleri

Bu bölümde PHP de veritabanından çektiğimiz veri listesini sayfalar halinde listelemeyi öğreneceğiz. Ayrıca sayfalama işlemimiz bittikten sonra da bu liste içerisinden arama işlemlerini yapacağız. Sayfalama işlemlerini yapmadan önce sayfalama yapacağımız veritabanını oluşturalım.

Proje senaryomuz online satış firmasına ait ürünlerin listelenmesi ve liste içerisinden ürün arama yapmak olsun. 'EmyoPazar' adında bir veritabanı ve ürünlerimizi tuttuğuz 'urunler' adında da tablomuz aşağıdaki gibi olsun. Şimdilik iki ürün var ama ileride ekleyeceğiz

Anahtara göre sırala:

+ Seçenekler

		id	uadi	aciklama	stok	fiyat	resim		
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	1	Defter	Beş ortalı harita metod kareli ders defteri MOPAK	158	16.50	
<input type="checkbox"/>	 Düzenle	 Kopyala	 Sil	2	YGS Soru Bankası	Her konuya ait 30 sr doruluk 20 testten ve bölüm s...	20	60.00	

Tümünü Seç Seçilileri:  Değiştir  Sil  Dışa Aktar

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

Sayfalama Ve Arama İşlemleri

Veritabanımızı oluşturduk ama veritabanına bağlanıp direk veritabanından çektiğimiz verileri listelemek yerine, öncelikle bir dizi halinde gelen verileri otomatik olarak tablo yapıp liste haline dönüştürecek bir PHP fonksiyonu hazırlamak istiyorum. Böylece bundan sonra verileri tablo yapmam gerekirse sürekli olarak aynı kodları yazmak yerine bu oluşturduğumuz fonksiyonu kullanmak daha mantıklı olacak. Bu arada zaten böyle bir fonksiyonu bir önceki dönem Web Programlama I dersinde yapmıştık. Dolayısı ile zaten hepiniz biliyorsunuz. :D

Evet Bu Fonksiyona Ait Kodları Bir Sonraki Sayfada Bulabilirsiniz.

Sayfalama Ve Arama İşlemleri

<?PHP

```
/// Bu fonksiyon kensdisine 1. parametrede, iki boyutlu bir dizi halinde gelen verileri
/// HTML deki table tagleri arasına yerleştirerek tablolama yapar. Fonksiyonun 2. parametresi
/// Bu tablonun sütunlarının başlıklarını tutan bir dizi olacak
function TabloYapBakalim($veriDizisi="", $baslikDizisi="")
{
    $metin="";
    ///Eğer Verilerimiz veya Başlıklarımız Boşsa yada Dizi Değilse Uyarı Verelim
    if(trim($veriDizisi)==" || trim($baslikDizisi)==" || !is_array($veriDizisi) || !is_array($baslikDizisi))
    {
        $metin="Veriler ve Başlıklar Dizi Olarak Gönderilmeli. Boş Olamazlar ";
    }else{
        $metin="<table border='1' cellpadding='10px' ><tr>";

        ///önce Başlıkları oluşturalım
        foreach($baslikDizisi as $ver)
            $metin.="<th>{$ver}</th>";
        //başlıkların satırını kapatalım
        $metin.="</tr>";

        //yukarıdaki üç satırlık kodla aşağıdaki satır aynı işlemi yapar
        ///$metin.="<tr><th>".implode("</th><th>",$baslikDizisi)."</th></tr>";

        ///şimdide Verileimizi Tabloya yükleyelim
        foreach($veriDizisi as $veriD)
        {
            $metin.="<tr>";
            foreach($veriD as $v)
                $metin.="<td>{$v}</td>";
        }

        //ensondaki tabloyu kapatıp tablomuzu gönderelim
        return($metin."</table>");
    }
}
```

?>

Sayfalama Ve Arama İşlemleri

Peki Oluşturduğumuz Bu Fonksiyonu Nasıl Kullanacağız. Örnek Bir Kullanım Şekli Aşağıdaki Gibidir.

Sizde istediğiniz bir sayfaya bu fonksiyonun bulunduğu sayfayı yükleyip fonksiyonu çağırarak kolayca tablo oluşturabilirsiniz. Dikkat etmeniz gereken nokta fonksiyon parametrelerini uygun şekilde düzenlemeniz gerektiğidir.

```
<?PHP

require_once("fonksiyonlar.php");

$veridiz=array( "1"=>array("adi"=>"Mutlu YAPICI", "yas"=>31, "meslek"=>"Akademisyen" ),
               "2"=>array("adi"=>"Ayse ÇALIŞKAN", "yas"=>24, "meslek"=>"Öğrenci" ),
               "3"=>array("adi"=>"Mine EMİNE", "yas"=>26, "meslek"=>"Ev Hanımı" )
               );

$baslikDiz= array("Adı Soyadı", "Yaşı", "Mesleği" );

echo TabloYapBakalim($veridiz, $baslikDiz);

?>
```


Sayfalama Ve Arama İşlemleri

Tabi listeleme yapabilmemiz için veritabanımızda onlarca veri olması gerekir. Bunun için önce bir veri ekleme sayfası oluşturalım ve veritabanımıza veri ekleyelim. Aslında bu sayfayı oluşturmadan da direkt PHPMYADMIN den veri ekleyebiliriz. Ancak bir önceki dönem yaptığımız derslerden de biliyorsunuz ki, veriyi PHPMYADMIN den eklersek tablo yaparken Türkçe karakter uyumsuzluğu ile karşılaşacağız. Bu nedenle bir veri ekleme arayüzü oluşturalım, hem bir önceki konuları da hatırlamış oluruz.

Sayfalama Ve Arama İşlemleri

Arayüze Ait Kodlar;

```
<h1 style="color:red;">Ürün Ekleme Sayfası</h1>
<form action="urunekle.php" method="POST">
  <table border="1" cellpadding="10px">
    <tr>
      <th> Ürünün Adı : </th><td><input type="text" value="" name="uadi"/></td>
    </tr>
    <tr>
      <th> Ürünün Açıklaması : </th><td><input type="text" value="" name="uack"/></td>
    </tr>
    <tr>
      <th> Ürünün Adeti : </th><td><input type="text" name="uadet" value="" /></td>
    </tr>
    <tr>
      <th> Ürünün Fiyatı : </th><td><input type="text" value="" name="ufyt"/></td>
    </tr>
    <tr>
      <th> Ürünün Resmi : </th><td><input type="text" value="" name="ursm"/></td>
    </tr>
    <tr>
      <td colspan="2" align="right"><input type="submit" value="Ürün Ekle" name="btn"/></td>
    </tr>
  </table>
</form>
```

Sayfalama Ve Arama İşlemleri

Ürün Ekleme Arayüzü

... Gözetim Programında... - DENEY Tasarımı 250KB ...

Ürün Ekleme Sayfası

Ürünün Adı :	<input type="text"/>
Ürünün Açıklaması :	<input type="text"/>
Ürünün Adeti :	<input type="text"/>
Ürünün Fiyatı :	<input type="text"/>
Ürünün Resmi :	<input type="text"/>
<input type="button" value="Ürün Ekle"/>	

Sayfalama Ve Arama İşlemleri

Ürün Ekleme Kodları. Bu kodları ayrı bir sayfada yada aynı sayfa içerisinde de kullanabilirsiniz.

```
<?php
if( !empty($_POST['uadi']) &&
    !empty($_POST['uadet']) &&
    !empty($_POST['ufyt'])
) {
    print_r($_POST);
    extract($_POST);
    $vt = @new mysqli("127.0.0.1","root","usbw","emyopazar");

    if($vt->connect_error)
        die("Veritabanı Bağlantı Hatası ".$vt->connect_error);

    $sql="INSERT INTO `urunler`(`uadi`, `aciklama`, `stok`, `fiyat`, `resim`)
        VALUES (?, ?, ?, ?, ?)";
    $vt->set_charset("utf8"); ///Türkçe Karakterleri Tanıması İçin
    $sorgu= $vt->prepare($sql);
    $sorgu->bind_param("ssis",$_POST['uadi'],$_POST['uack'],$uadet,$ufyt,$ursm);
    $sorgu->execute();

    if($sorgu ===false)
        die("<h3 style='color:red;'>Veritabanı Ekleme Hatası ".$vt->error."</h3>");
    else
        echo "<h3 style='color:#00FF00;'>".$vt->affected_rows." Satır Veri Eklendi</h3>";

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

Sayfalama Ve Arama İşlemleri

Veritabanımız ve Tablo şeklinde liste oluşturabileceğimiz fonksiyonumuz hazır. Ürünlerimizi de ekledik. Şimdi sayfalama işlemlerine geçebiliriz.

Öncelikle sayfalama işlemlerinin mantığından bahsetmek istiyorum. Sayfalama işlemlerinin temel mantığı veritabanındaki tüm verileri çekip aynı anda yüzlerce veriyi ekrana bastırmak yerine onarlı veya yirmişerli gibi gruplar halinde ekranda listelemektir. Böylece yüzlerce verinin tarayıcıdaki yükü azaltılacak daha rahat bir analiz işlemi yapılabilecektir. Sayfanızda da kitlenmeler ve takılmalar azalacaktır.

Peki ama bu işlemi nasıl yapacağız. Aslında sayfalama işlemi küçük bir algoritma ve SQL sorgusundan ibarettir. Verileri veritabanından çerkerken kaçınıcı sıradaki veriden itibaren kaç adet çekmek istiyorsak SQL sorgusuyla ayarlayacağız. Böylece istediğimiz yerden itibaren istediğimiz veri grubunu ekranda göstermiş olacağız.

Sayfalama Ve Arama İşlemleri

Bunu örneklemek gerekirse, veritabanında 100 satır veri olsun biz yirmişerli gruplar halinde ekrana yazdırmak ve her yirmi gruptan bir sayfa oluşturmak isteyelim. Bunun için SQL sorgusuna hangi satırdan başlayacağını ve kaç tane veri çekeceğini söylememiz gerekir. Bu noktada OFFSET ve LIMIT kavramı devreye giriyor SQL cümleciğinde veri çekme işleminin kaçınıcı satırdan başlayacağını OFFSET, ve kaç satır çekeceğini de LIMIT anahtar kelimeleri belirlemektedir. Bizde sayfalama işlemi için tam olarak bunları kullanacağız. Tabi sayfa numaralarına bağlı ufak bir algoritmayla birlikte :D

Öncelikle basitten zora doğru gitmek için veritabanımızda ki tüm verileri çekip oluşturduğumuz tablolama fonksiyonu ile listeleyelim daha sonra sayfalara böleceğiz.

Sayfalama Ve Arama İşlemleri

Listeleme Dosyasının İçeriği;

zoku.php | cerezsil.php | cerezolustur.php | anasayfa.php | new 2 | urunle.php | sayfalama.php | fonksiyonlar.php

```
<?php
require_once("fonksiyonlar.php");
//öncelikle verilerimizi veritabanından çekelim
$vt=new mysqli( "localhost","root","usbw","emyopazar");
//eğer bağlantı başarısızsa hata verdirelim
if($vt->connect_error)
    die("Veritabanı Bağlantı Hatası. ".$vt->connect_error);

//bağlantı başarılı ise
$SQL="SELECT * FROM `urunler`";
$vt->set_charset("UTF8");
$sorgu=$vt->prepare($SQL);
$sorgu->execute();
//eğer Sorgu başarısızsa hata verdirelim
if($vt->error)
    die("SQL Sorgu Hatası. ".$vt->error);

$sonuc=$sorgu->get_result();
```

**Fonksiyonlar Dosyasını
Yüklüyoruz**

Sayfalama Ve Arama İşlemleri

Listeleme Dosyasının İçeriği (DEVAMI);

```
8 $sonuc=$sorgu->get_result();
9
0 $veriDizi=Array();
1 $i=0;
2 while($veri=$sonuc->fetch_assoc())
3 {
4     $veriDizi[$i]=$veri;
5     $i++;
6 }
7
8 $sorgu->close();
9 $vt->close();
0
1 ///başlıkları oluşturalım ve Tablo yapma fonksiyonumuza gönderelim
2 $basliklar=Array("Ürün İd","Ürün Adı","Ürün Açıklaması","Stok","Fiyatı","Resim")
3
4 echo TabloYapBakalim($veriDizi, $basliklar);
```

?>

Sayfalama Ve Arama İşlemleri

Dosyanın Çıktısı

localhost:8080/sayfalama.php

Arama

.. Görsel Programlama... Bellek tasarımı 256KB ... Multiplexer example ... Implementation of ... UDEMY KURS Çevrimi... 8X1 Multiplexer - Y... Logic Circuit Design fo... Logic Circuit

Ürün İd	Ürün Adı	Ürün Açıklaması	Stok	Fiyatı	Resim
1	Defter	Beş ortalı harita metod kareli ders defteri MOPAK	158	16.50	
2	YGS Soru Bankası	Her konuya ait 30 sr doruluk 20 testten ve bölüm sonu deneme sınavından oluşan soru bankası	20	60.00	
3	Kitap	DGS Soru Bankası	10	150.00	../resim/dolmakalem.jpg
4	Kalem	Mavi Renkli Dolma Kalem	15	22.00	../resim/dolmakalem.jpg
5	Şeker	Düdüklü Şeker	10	2.00	../resim/teogtestkitabi.png

Sayfalama Ve Arama İşlemleri

Tüm verileri listeleme işlemi yaptık peki ama şimdi istediğimiz veriden itibaren istediğimiz sayıda veriyi ekrana nasıl yazdıracağız. Bunun için daha öncede söylediğimiz OFFSET ve LIMIT SQL anahtar kelimelerini kullanacağız.

Öncelikler bu liste üzerinde, bu anahtar kelimelerin etkisini görelim. Sayfalama.php dosyasının içerisindeki aşağıda gösterilen SQL cümleliğini yeni haliyle değiştirip sonucu görelim.

```
8 //bağlantı başarılı ise  
9 $SQL="SELECT * FROM `urunler`";  
10 $vt->set_charset("UTF8");  
11 $sorgu=$vt->prepare($SQL);  
12
```

\$SQL="SELECT * FROM `urunler` LIMIT 2 OFFSET 1"

Sayfalama Ve Arama İşlemleri

Ürün İd	Ürün Adı	Ürün Açıklaması	Stok	Fiyatı	Resim
1	Defter	Beş ortalı harita metod kareli ders defteri MOPAK	158	16.50	
2	YGS Soru Bankası	Her konuya ait 30 sr doruluk 20 testten ve bölüm sonu deneme sınavından oluşan soru bankası	20	60.00	
3	Kitap	DGS Soru Bankası	10	150.00	../resim/dolmakalem.jpg
4	Kalem	Mavi Renkli Dolma Kalem	15	22.00	../resim/dolmakalem.jpg
5	Şeker	Düdüklü Şeker	10	2.00	../resim/teogtestkitabi.png

Yukarıda Listenin İlk hali aşağıda ise son hali var. Farkları görebiliyoruz.

\$\$SQL="SELECT * FROM `urunler` LIMIT 2 OFFSET 1"

Ürün İd	Ürün Adı	Ürün Açıklaması	Stok	Fiyatı	Resim
2	YGS Soru Bankası	Her konuya ait 30 sr doruluk 20 testten ve bölüm sonu deneme sınavından oluşan soru bankası	20	60.00	
3	Kitap	DGS Soru Bankası	10	150.00	../resim/dolmakalem.jpg

Görüldüğü gibi **OFFSET 1** kelimesi ile 1. ofset olan 2. elemandan itibaren veri çekmeye başladı. **LIMIT 2** kelimesi ile de kaç tane veri çekeceğimizi belirlemiş olduk.

Sayfalama Ve Arama İşlemleri

Şimdi sayfalama işlemi için tablonun altına sayfalarda dolaşabileceğimiz. 1-2-3-4-gibi linkler ekleyelim ve bu linklere tıkladıkça veriler değişsin. Öncelikle linkleri ekleyerek başlayalım.

Daha önceki sayfalama kodumuzun en alttaki fonksiyonun altına aşağıdaki kodları ekleyebilirsiniz.

```
echo TabloYapBakalim($veriDizi, $basliklar);  
///tablo altına sayfa numaraları oluşturalım  
for($i=1;$i<5;$i++)  
{  
    echo "<a href='#' >$i</a> <b>|</b> ";  
}
```

Ürün İd	Ürün Adı	Ürün Açıklaması	Stok	Fiyatı	
2	YGS Soru Bankası	Her konuya ait 30 sr doruluk 20 testten ve bölüm sonu deneme sınavından oluşan soru bankası	20	60.00	
3	Kitap	DGS Soru Bankası	10	150.00	../r

Sayfalama Ve Arama İşlemleri

Şimdi sayfalama işlemi için tablonun altına sayfalarda dolaşabileceğimiz. 1-2-3-4-gibi linkler ekleyelim ve bu linklere tıkladıkça veriler değişsin. Öncelikle linkleri ekleyerek başlayalım.

Daha önceki sayfalama kodumuzun en alttaki fonksiyonun altına aşağıdaki kodları ekleyebilirsiniz.

```
echo TabloYapBakalim($veriDizi, $basliklar);  
///tablo altına sayfa numaraları oluşturalım  
for($i=1;$i<5;$i++)  
{  
    echo "<a href='#' >$i</a> <b>|</b> ";  
}
```

Ürün İd	Ürün Adı	Ürün Açıklaması	Stok	Fiyatı	
2	YGS Soru Bankası	Her konuya ait 30 sr doruluk 20 testten ve bölüm sonu deneme sınavından oluşan soru bankası	20	60.00	
3	Kitap	DGS Soru Bankası	10	150.00	../r

Sayfalama Ve Arama İşlemleri

Evet sayfalarımızı oluşturduk ancak tıkladığımızda verilerin değişmediğini görüyorsunuz. Şimdi işlevsellik kazandırmak için her sayfaya tıkladığımızda GET ile bir ofset numarası göndermeli ve ona göre SQL cümlesi düzenlenmelidir.

Sayfalama.php sayfamızın son halini tekrara yazalım değişiklik yaptığımız yerler kırmızı çerçeve içerisindedir.

Sayfalama Ve Arama İşlemleri

Sayfalama.php içeriği;

```
<?php
require_once("fonksiyonlar.php");
//öncelikle verilerimizi veritabanından çekelim
$vt=new mysqli( "localhost","root","usbw","emyopazar");
//eğer bağlantı başarısızsa hata verdirelim
if($vt->connect_error)
    die("Veritabanı Bağlantı Hatası. ".$vt->connect_error);

$offsetim=0;
if(!empty($_GET['offsetno']))
    $offsetim=$_GET['offsetno']+0;

//bağlantı başarılı ise
//$$SQL="SELECT * FROM `urunler`";
$$SQL="SELECT * FROM `urunler` LIMIT 2 OFFSET ?";
$vt->set_charset("UTF8");
$sorgu=$vt->prepare($$SQL);
$sorgu->bind_param('i',$offsetim);
$sorgu->execute();
```

Sayfalama Ve Arama İşlemleri

Sayfalama.php içeriği (DEVAMI);

```
0 //eğer Sorgu başarısızsa hata verdirelim
1 if($vt->error)
2     die("SQL Sorgu Hatası. ".$vt->error);
3
4 $sonuc=$sorgu->get_result();
5
6 $veriDizi=Array();
7 $i=0;
8
9 while($veri=$sonuc->fetch_assoc())
10 {
11     $veriDizi[$i]=$veri;
12     $i++;
13 }
```


Sayfalama Ve Arama İşlemleri

Sayfalama.php içeriği (DEVAMI);

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

///başlıkları oluşturalım ve Tablo yapma fonksiyonumuza gönderelim
$basliklar=Array("Ürün İd","Ürün Adı","Ürün Açıklaması","Stok","Fiyatı","Resim");

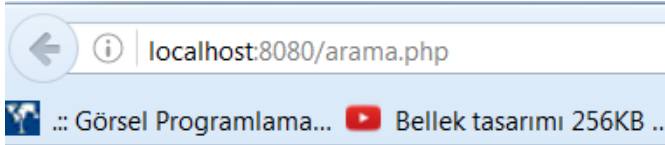
echo TabloYapBakalim($veriDizi, $basliklar);
///tablo altına sayfa numaraları oluşturalım
$listelenecekSatirSayisi=2;
for($i=1;$i<5;$i++)
{
    echo"<a href='sayfalama.php?ofsetno=".((($i-1)*$listelenecekSatirSayisi)." '>$i</a> <b>|</b> "
}
```

Sayfalama Ve Arama İşlemleri

Sayfalama içeriğini yaptık şimdi sıra geldi arama yapmaya veritabanındaki veriler içerisinde arama yapmak için aslında şimdiye kadar yaptığımız listeleme işlemlerinden farklı bir işlem yapmayacağız sadece SQL sorgusu değişecek.

Aslında bu arama işlemini biraz önce yaptığımız sayfalama tablosunun üstüne bir arama kutusu ekleyerek yapmak istiyorum ama kodlar daha fazla karmaşık olmasın diye öncelikle ayrı bir arama sayfası yapacağım.

Öncelikle arayüzü aşağıdaki gibi tasarlayalım. Sayfamızın adı 'arama.php' olacak.



Ürün Arama Sayfası

Ürünün Adı :	<input type="text"/>
<input type="button" value="Ürün Ara"/>	

Sayfalama Ve Arama İşlemleri

'arama.php' arayüz kodları;

```
<h1 style="color:red;">Ürün Arama Sayfası</h1>
<form action="arama.php" method="POST">
  <table border="1" cellpadding="10px">
    <tr>
      <th> Ürünün Adı : </th><td><input type="text" value="" name="uadi"/></td>
    </tr>
    <tr>
      <td colspan="2" align="right"><input type="submit" value="Ürün Ara" name="btn"/></td>
    </tr>
  </table>
</form>
```



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/arama.php'. The page content includes the title 'Ürün Arama Sayfası' in red text. Below the title is a search form with a label 'Ürünün Adı :', an input field, and a 'Ürün Ara' button.

Sayfalama Ve Arama İşlemleri

'arama.php' sayfasında Ürün Ara butonuna tıkladığımızda veri tabanından ürünü bulup aynı sayfada tablo halinde listeleyelim. Öncelikle birebir arama yapalım yani ürün adını tam olarak arasın. Daha sonrada yazdığımız kelime herhangi bir ürünün adında veya bir parçasında geçiyorsa da listeleyebilelim.

Kodlarımızı çalıştırdığımızda eğer ürün varsa aşağıdaki gibi listemizi oluşturuyor.



localhost:8080/arama.php

.. Görsel Programlama... Bellek tasarımı 256KB ... Multiplexer example ... Implementation of ... U UDEMY KURS

Ürün Arama Sayfası

Ürünün Adı :

Ürün İd	Ürün Adı	Ürün Açıklaması	Stok	Fiyatı	Resim
1	Defter	Be? ortal? harita metod kareli ders defteri MOPAK	158	16.50	

Sayfalama Ve Arama İşlemleri

'arama.php' sayfasına ait kodlar; Bu kodları aynı sayfanın en altına ekliyoruz.

```
3 <?php
4 //eğer aranacak kelime geldiyse arayıp listeyi oluşturalım
5 if(!empty($_POST['uadi']))
6 {
7     require_once("fonksiyonlar.php");
8     $aranacakKelime=$_POST['uadi'];//aranacak kelimeyi aldık
9     $vt=new mysqli("localhost","root","usbw","emyopazar");
10
11     if($vt->connect_error)
12         die("Veritabanı Bağlantı Hatası. ".$vt->connect_error);
13
14     $sql="SELECT * FROM `urunler` WHERE `uadi`=?";
15     $sorgu=$vt->prepare($sql);
16     $sorgu->bind_param('s',$aranacakKelime);
17     $sorgu->execute();
```

Sayfalama Ve Arama İşlemleri

'arama.php' sayfasına ait kodlar (DEVAMI);

```
if($sorgu->error)
    die("SQL Sorgu Hatası. ".$svt->error);

$sonuc=$sorgu->get_result();

$veriDizi=Array();
$i=0;
while($veri=$sonuc->fetch_assoc())
{
    $veriDizi[$i]=$veri;
    $i++;
}

$sorgu->close();
$svt->close();

///başlıkları oluşturalım ve Tablo yapma fonksiyonumuza gönderelim
$basliklar=Array("Ürün İd","Ürün Adı","Ürün Açıklaması","Stok","Fiyatı","Resim");

echo TabloYapBakalim($veriDizi, $basliklar);
}
```

Sayfalama Ve Arama İşlemleri

Arama sayfamızda hazır ve çalışıyor ancak fark ettiyseniz sadece aradığımız kelime bire bir tutuyorsa liste oluşuyor. D ile başlayan tüm verileri listele diyemiyoruz. Aynı şekilde 'a' ile bitenleri listele diyemiyoruz peki bunu nasıl yapacağız.

Aslında bu işlem oldukça basit sadece biraz önceki proje üzerinden SQL sorgumuzu değiştireceğiz. Şimdi bunu yapalım.

SQL cümlesi içerisinde LIKE anahtar kelimesi aranacak değerler benzer değerlerinin de bulunmasına olanak sağlar.

Sayfalama Ve Arama İşlemleri

Aşağıda önceki sorgu ve değiştirilmiş hali görülmektedir.

```
$sql="SELECT * FROM `urunler` WHERE `uadi`=?";  
$sorgu=$vt->prepare($sql);
```

Son hali ;

```
$aranacakKelime=("%" . $aranacakKelime . "%");  
$sql="SELECT * FROM `urunler` WHERE `uadi` LIKE ?";  
$sorgu=$vt->prepare($sql);
```

Sonuç olarak SQL cümleciğini

```
$sql="SELECT * FROM `urunler` WHERE `uadi` LIKE '%$aranacakKelime%'";
```

Haline dönüştürmüş olduk. Böylece istediğimiz gibi esnek bir arama yapabiliyoruz.

KAYNAKLAR

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

