

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

- Nesne tabanlı programlama ve tanımı
- Class, Method, Fonction, KeyWord kavramları
- Class ve Method oluşturma
- Access Modifiers (Erişim Düzenleyiciler)
- Kurucu ve yıkıcı metodlar

Kategori İşlemleri

PHP ile şimdiye kadar yaptığımız çalışmaları kapsayacak bir örnek yapalım. Kategorilendirme işlemleri web programlamada önemli bir yer tutmaktadır. İstediğiniz kadar dinamik kategori eklemek, istediğiniz kategorinin altına alt kategoriler eklemek yada bu kategoriler üzerinde işlemler yapabilmek için iyi bir algoritma tasarlamış olmanız gerekmektedir. Buradaki önemli nokta her hangi bir kategorinin hangi kategorinin altında olduğunu belirlemek ve yeni bir kategori eklendiğinde yada silindiğinde var olan diğer kategorileri etkilememesini sağlamaktır.

Bunu yapabilmek için her bir kategorinin bağlı olduğu üst kategori id sini veri tabanında tutmanız gerekir, böylece hangi kategorinin altında hangileri var anlayabilirsiniz. Ayrıca bir kategori altındaki alt kategoriler arasında sıralama işlemi de yapmak isterseniz o zaman birde sıra numarası alanı ekleyebilirsiniz.

Şimdiye kadar öğrendiğimiz PHP, veritabanı, AJAX ve CLASS işlemleri doğrultusunda dinamik bir kategori yönetim paneli geliştirelim ve bunu ajax ile yaparak daha etkili ve kullanışlı hale getirelim

Kategori İşlemleri

Öncelikle bu kategori yönetim sistemi için veritabanını tasarlamamız gerekir. Veritabanında kategori adını, üst kategori idsini, kategori ikonunu ve sıra numarasını saklayalım. Bu bilgilere ait oluşturulmuş veritabanı aşağıdaki gibidir. Veritabanı adım KategoriDB ve tablo adım Kategoriler olacak.

localhost ▶ KategoriDB ▶ kategoriler

Gözet Yapı SQL Ara Ekle Dışa Aktar İçe Aktar İşlemler

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
<input type="checkbox"/>	1	id			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼
<input type="checkbox"/>	2	adi	utf8_turkish_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
<input type="checkbox"/>	3	ustKat			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
<input type="checkbox"/>	4	ikon	utf8_turkish_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
<input type="checkbox"/>	5	siraNo			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼

↑ Tümünü Seç / Hiçbirini Seçme Seçilileri: Gözet Değiştir Kaldır Birincil Benzersiz İndeks

Kategori İşlemleri

Şimdi öncelikle bu kategorilerle ekleme, silme güncelleme, listeleme, sıra değiştirme gibi işlemleri yapabileceğimiz bir kategori sınıfını oluşturalım. Bütün kategori işlemlerini bu sınıf ile gerçekleştireceğiz.

İlk olarak bu sınıfa ait kategori ekleme fonksiyonunu tanımlayacağız ve bir kategori eklerken bu fonksiyondan faydalanacağız. Bu fonksiyona kategori adi, üst kategori id, ikon ve sıra no bilgilerini göndererek veritabanına ekleme işlemini gerçekleştireceğiz.

Tabi burada veritabanı bağlantı kodlarını da yazmamız gerekecek ama isterseniz bu kısmı ayrı bir sınıf olarak tasarlayıp çağıra bilirsiniz.

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Sınıfımız ve kategori ekleme fonksiyonunu

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x
1 <?php
2 class Kategori{
3
4     function KategoriEkle($adi,$ustKategori=0,$iconUrl="", $siralama=0) {
5         if(empty($adi))
6             return false;
7
8         $vt=new mysqli("localhost","root","usbw","KategoriDB");
9         if($vt->connect_error)
10            return "Veritabanı Bağlantı Hatası";
11
12        $sql="INSERT INTO `kategoriler` (`adi`,`ustKat`,`ikon`,`siraNo`)
13            VALUES      (?, ?, ?, ?)";
14
15        $vt->set_charset('UTF8');
16        $sorgu=$vt->prepare($sql);
17        $sorgu->bind_param('sssi',$adi,$ustKategori,$iconUrl,$siralama);
18        $sorgu->execute();
19
20        if($vt->error)
21            return "Sorgu Hatası ".$vt->error;
22
23        $sorgu->close();
24        $vt->close();
25
26        return "Kategori Başarıyla Eklendi";
27    }
28 }
29 ?>
```

Kategori İşlemleri

Şimdi bu kategori sınıfımızı nasıl kullanacağımıza bakalım. Öncelikle kategoriEkle.php dosyamızı hazırlayalım. Bu dosyamızın kodları aşağıdaki gibidir.

Görüldüğü üzere öncelikle Kategori sınıfımızı include ediyoruz, çünkü tüm işlemleri bu sınıftaki fonksiyonlar aracılığı ile gerçekleştireceğiz. Daha sonra bu sınıftan bir nesne yaratarak, KategoriEkle fonksiyonunu çağırıyoruz ve kategori ekle fonksiyonundan gelen sonucu ekrana yazdırıyoruz.

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x
1 <?php
2 include_once("class.php");
3
4 $kategoriSinifi = new Kategori();
5
6 $sonuc=$kategoriSinifi->KategoriEkle("Arabalar",0,"resimler/araba.gif",0);
7 echo "sonuc ".$sonuc;
8
9
10 ?>
```


Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

KategoriEkle.php sayfamızı çalıştırdığımızda, eğer başarılı bir şekilde kategori eklendiyse aşağıdaki gibi bir çıktı sayfamızda görünecektir ve veritabanımız aşağıdaki gibi olacaktır.

http://localhost:8080/sil/kategorEkle.php

localhost:8080/sil/kategorEkle.php Arama

sonuc Kategori Başarıyla Eklendi

```
SELECT *
FROM `kategoriler`
LIMIT 0 , 30
```

Göster : 30 satır, başlayacağı satır numarası 0 yatay kipinde

+ Seçenekler

	id	adi	ustKat	ikon	siraNo
<input type="checkbox"/> Düzenle Sıralı Düzenleme Kopyala Sil	1	Arabalar	0	resimler/araba.gif	0

Tümünü Seç / Hiçbirini Seçme Seçilileri: Değiştir Sil Dışa Aktar

Kategori İşlemleri

KategoriEkle.php sayfamızı hazır gerçi şuanda statik veri ekliyoruz yani verileri bir form arayüzünden alıp eklemiyoruz. Formu tasarlamadan önce verilerimizi listeleme fonksiyonunu da sınıfımıza eklemek istiyorum daha sonra, ekleme işlemini form ve AJAX ile daha güzel ve verimli hale getireceğiz.

Listeleme işleminde fonksiyonumuz bize tüm kategorilerin hiyerarşik olarak gösterildiği bir tablo düzeni oluştursun. Bu sebeple veritabanından verileri çekerken öncelikle, hangi kategorinin önce hangisinin sonra geleceğini bilmemiz gerekir. Bu sebeple verileri sıra numaralarına göre artan sırada alacağız. Yani bir kategoride birden fazla alt kategori varsa bunlardan sıra numarası küçük olan en üstte daha büyükler altta gelecek, böylece kategoriler arası sıralama yapmış olacağız.

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Listeleme fonksiyonumuz Öz yinelemeli (recursive) yani kendi kendini çağırarak bir fonksiyon olacak, her çağırışında bir önceki kategorinin id verisini ustKategori verisi olarak gönderecek ki böylece bir sonraki işlemde bir önceki kategorinin alt kategorilerini oluşturabilelim. Evet biraz karışık görünüyor ama çalıştırdığımızda daha rahat anlaşılacaktır. KategoriEkle fonksiyonunun altına yeni bir fonksiyon ekeldik.

```
return "Kategori Başarıyla Eklendi";
```

```
function KategoriListele($veriDizisi, $ustKat=0){
    if(!is_array($veriDizisi))
        return "Veri Yok";

    $metin("<ul>");//yeni bir liste açıyoruz.
    ///dizideki tüm elemanları bu listeye ekleyelim.
    foreach($veriDizisi as $veri)
    {
        if($veri['ustKat']==$ustKat)//sadece üst kategori id si aynı olanları aynı listeye ekleyelim.
        {
            $metin.="<li>{$veri['adi']}";
            ///fonksiyonu tekrar çağıralım
            $metin.= $this->KategoriListele($veriDizisi, $veri['id']);
            $metin.="</li>";
        }
    }

    return ($metin."</ul>");    ///sonucu gönderelim
}
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Görüldüğü üzere bu fonksiyona kategoriye dönüştürülecek veriler VeriDizisi olarak geliyor ve ikinci parametre olarak ta ustKategori verisi gönderiliyor. Böylece üst kategoriye ait alt kategoriler bir tagine yerleştirilerek gönderiliyor. Bu işlemler öz yinelemeli olarak tekrarlandığında tüm kategoriler alt alta oluşacaktır.

```
return "Kategori Başarıyla Eklendi";
```

```
function KategoriListele($veriDizisi, $ustKat=0){
    if(!is_array($veriDizisi))
        return "Veri Yok";

    $metin="<ul>";//yeni bir liste açıyoruz.
    ///dizideki tüm elemanları bu listeye ekleyelim.
    foreach($veriDizisi as $veri)
    {
        if($veri['ustKat']==$ustKat)//sadece üst kategori id si aynı olanları aynı listeye ekleyelim.
        {
            $metin.="<li>{$veri['adi']}";
            ///fonksiyonu tekrar çağıralım
            $metin.= $this->KategoriListele($veriDizisi, $veri['id']);
            $metin.="</li>";
        }
    }

    return ($metin."</ul>");    ///sonucu gönderelim
}
```

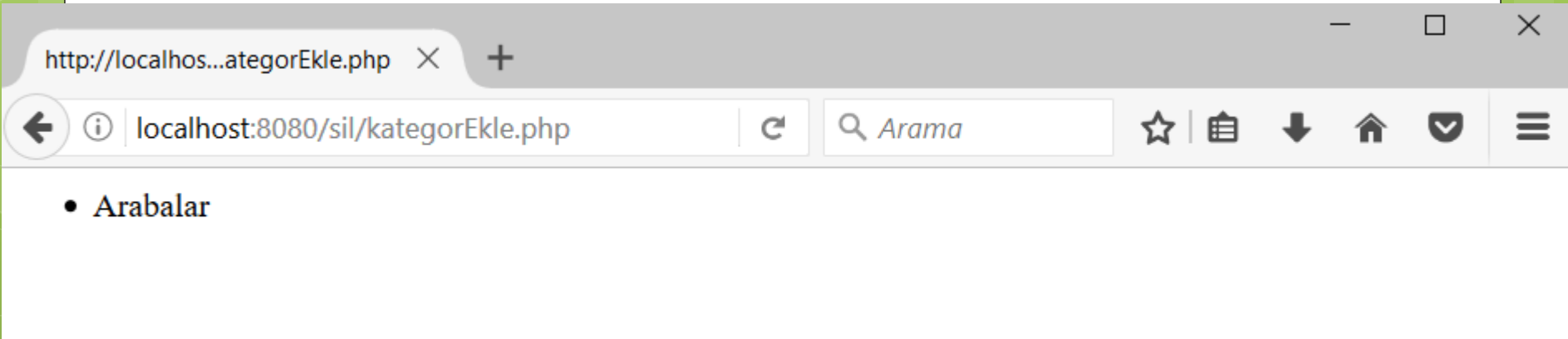
Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Fonksiyonumuzu yine KategoriEkle.php içerisinde çağıracağız. Tabii öncelikle veritabanındaki tüm verileri alıp diziye yüklüyoruz ve sonra KategoriListele fonksiyonunu çağırıp sonucu yazdırıyoruz.

Bir sonraki sayfada kodu incelediğinizde göreceksiniz ki en altta kategori ekleme kodlarını gizledim. Eğer gizlemezsek her Listeleme yaptığımızda aynı veriyi sürekli ekler. Ekleme işlemini formlar ve AJAX ile düzenlediğimizde bu sorun ortadan kalkmış olacak.

Listeleme fonksiyonumuzun çıktısı aşağıdaki gibidir. Sadece Arabalar eklendiği için tek bir kategori görüyoruz.



Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

KategoriEkle.php içerisinde çağırmaı aşağıdaki gibi gerçekleştiriyoruz. Öncelikle tüm verileri veritabanından alıp bir diziye yüklüyor ve sonra fonksiyonu çağırıyoruz.

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x
1 <?php
2 include_once("class.php");
3 $kategoriSinifi = new Kategori();
4
5 ///veritabanından tüm verileri alıp kategori listesini oluşturalım
6 $vt=new mysqli("localhost","root","usbw","KategoriDB");
7 if($vt->connect_error)
8     return "Veritabanı Bağlantı Hatası";
9
10 $sql="SELECT * FROM `kategoriler` ORDER BY `siraNo` ASC";
11
12 $vt->set_charset('UTF8');
13 $sorgu=$vt->prepare($sql);
14 $sorgu->execute();
15
16 if($vt->error)
17     return "Sorgu Hatası ".$vt->error;
18 $sonuc=$sorgu->get_result();
19 ///önce tüm kategori verilerini alıp bir diziye yükleyelim
20 $kategoriDizisi=$sonuc->fetch_all(MYSQLI_ASSOC);
21
22 $sorgu->close();
23 $vt->close();
24
25 $listeSonucu= $kategoriSinifi->KategoriListele($kategoriDizisi);
26 echo $listeSonucu;
27
28 // $sonuc=$kategoriSinifi->KategoriEkle("Arabalar",0,"resimler/araba.gif",0);
29 //echo "sonuc ".$sonuc;
30
31
32 ?>
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Şimdi daha fazla kategori ekleyelim. Ekleme işlemi daha düzgün yapabilmek için önce bir ekleme form sayfası oluşturacağız ve sonra da AJAX ile verileri göndereceğiz. KategoriIslemleri.php isimli dosyamızı oluşturalım.

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x kategoriislemleri.php x
1 <html>
2 <head>
3 <title>Kategori İşlemleri</title>
4 <script type="text/javascript">
5 </script>
6 </head>
7 <body>
8 <div id="AjaxSonuc"> Ajax Sonucu Burada Gösterilecek </div>
9 <form action="" method="GET">
10 <table border="0" cellpadding="10">
11 <tr>
12 <th>Kategori Adı </th>
13 <td><input type="text" name="kadi"/></td>
14 </tr>
15 <tr>
16 <th>Üst Kategori Id </th>
17 <td><input type="text" name="ustKId"/></td>
18 </tr>
19 <tr>
20 <th>İkon Seç </th>
21 <td><input type="file" name="icon"/></td>
22 </tr>
23 <tr>
24 <th>Sıra No </th>
25 <td><input type="text" name="sira"/></td>
26 </tr>
27 <tr>
28 <td colspan="2"><input type="button" id="btn" value="Kategori Ekle"/></td>
29 </tr>
30 </table>
31 </form>
32 </body>
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Formumuz hazır şimdide AJAX kodlarını Ekleyelim.

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x kategorislemeleri.php x
1 <html>
2 <head>
3 <title>Kategori İşlemleri</title>
4 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
5 <script type="text/javascript">
6     $(document).ready(function() {
7         $("#btn").click(function() {
8             $.ajax({
9                 type:"GET",
10                url:"kategorEkle.php",
11                data:$("form").serialize(),
12                dataType:"HTML",
13                beforeSend:function (cevap){$("#AjaxSonuc").html("İşlem Yapılıyor...");},
14                success:function (cevap){$("#AjaxSonuc").html(cevap);},
15                error:function () {alert("Hata Oluşturdu");}
16            });
17        });
18    });
19 </script>
20 </head>
21 <body>
22 <div id="AjaxSonuc"> Ajax Sonucu Burada Gösterilecek </div>
23 <form action="" method="GET">
24     <table border="0" cellpadding="10">
25     </table>
26 </form>
```


Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Şimdi ekleme işleminin doğru çalışabilmesi için formdan gelen verileri KategorEkle.php sayfasından alıp ekleme işlemini gerçekleştirelim. KategorEkle.php nin son hali aşağıdaki gibi dir.

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x kategoriislemleri.php x
1 <?php
2 include_once("class.php");
3 $kategoriSinifi = new Kategori();
4
5 ////KATEGORİYE EKLEME İŞLEMLERİ
6 if(!empty($_GET['kadi']))
7 {
8     $kadi=$_GET['kadi'];
9     $ustKId=(isset($_GET['ustKId']))?$_GET['ustKId']:0;
10    $ikon=(isset($_GET['ikon']))?$_GET['ikon']:"";
11    $siraNo=(isset($_GET['sira']))?$_GET['sira']:0;
12    $sonuc=$kategoriSinifi->KategoriEkle($kadi,$ustKId,$ikon, $siraNo);
13    echo "Ekleme Sonucu <h3 style='color:red;'> $sonuc </h3>";
14 }
15
16 ////KATEGORİ LİSTELEME İŞLEMELRİ
17 ///veritabanından tüm verileri alıp kategori listesini oluşturalım
18 $vt=new mysqli("localhost","root","usbw","KategoriDB");
19 if($vt->connect_error)
20     return "Veritabanı Bağlantı Hatası";
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

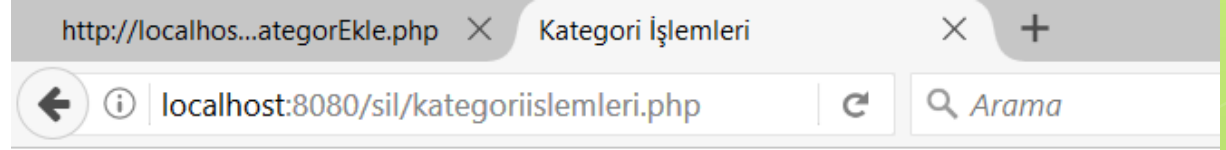
KategorEkle.php DEVAMI Listeleme kodlarını ekleme kodlarından sonra yaptım ki son eklenen de listelene bilsin.

```
16      ///KATEGORİ LİSTELEME İŞLEMELRİ
17      ///veritabanından tüm verileri alıp kategori listesini oluşturalım
18      $vt=new mysqli("localhost","root","usbw","KategoriDB");
19      if($vt->connect_error)
20          return "Veritabanı Bağlantı Hatası";
21
22      $sql="SELECT * FROM `kategoriler` ORDER BY `siraNo` ASC";
23      $vt->set_charset('UTF8');
24      $sorgu=$vt->prepare($sql);
25      $sorgu->execute();
26
27      if($vt->error)
28          return "Sorgu Hatası ".$vt->error;
29      $sonuc=$sorgu->get_result();
30      ///önce tüm kategori verilerini alıp bir diziye yükleyelim
31      $kategoriDizisi=$sonuc->fetch_all(MYSQLI_ASSOC);
32
33      $sorgu->close();
34      $vt->close();
35
36      $listeSonucu= $kategoriSinifi->KategoriListele($kategoriDizisi);
37      echo $listeSonucu;
38
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Kategoriislemleri.php dosyamızı çalıştırıp yeni bir eleman eklediğimizde sonuç aşağıdaki gibi olacaktır.



Kategori Başarıyla Eklendi

- Arabalar
- Çiçekler
- Evler
- Bilgisayar
 - Laptop
 - Masa Üstü

Kategori Adı

Üst Kategori Id

İkon Seç Hiçbir dosya seçilmedi.

Sıra No

Kategori İşlemleri

Burada sizde fark etmişsinizdir ki, her yeni elemanı eklerken üst kategori id sini girmemiz gerekir ancak hangi elemanın id si kaç bunu bilmemiz zordur. Hatta veri arttıkça imkansız olacaktır bunun yerine bir açılır kutu yapalım ve bu açılır kutu içerisinde hangi kategoriye ekleyeceğimizi seçelim. Böylece daha kullanışlı olacaktır.

Açılır kutu için önce veritabanındaki tüm kategorileri alıp select box a yükleyelim. Kategoriislemeleri.php sayfamızın Üst Kategori İd kısmı değişecektir. Sayfanın Görünümü;

http://localhost...ategorEkle.php x Kategori İşlemleri x +

localhost:8080/sil/kategoriislemeleri.php Arama

Ajax Sonucu Burada Gösterilecek

Kategori Adı

Üst Kategori Id

İkon Seç bir dosya seçilmedi.

Sıra No

Kategori Ekle

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Kategoriislemleri.php sayfamızın Üst Kategori İd kısmının son hali;

```
new 2 x class.php x kategorEkle.php x kategoriislemleri.php x
<tr>
  <th>Kategori Adı </th>
  <td><input type="text" name="kadi"/></td>
</tr>
<tr>
  <th>Üst Kategori Id </th>
  <td>
    <select name="ustKId">
      <?php
        //önce tüm verileri alıp diziye yükleyelim.
        $vt=new mysqli("localhost","root","usbw","KategoriDB");
        if($vt->connect_error)
          return "Veritabanı Bağlantı Hatası";

        $sql="SELECT * FROM `kategoriler` ORDER BY `adi` ASC";
        $vt->set_charset('UTF8');
        $sorgu=$vt->prepare($sql);
        $sorgu->execute();

        if($vt->error)
          return "Sorgu Hatası ".$vt->error;
        $sonuc=$sorgu->get_result();
        $kategoriDizisi=$sonuc->fetch_all(MYSQLI_ASSOC);

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

        ///şimdi de verileri açılır kutuya ekleyelim
        foreach($kategoriDizisi as $veri)
          echo "<option value='{ $veri['id'] }'>{ $veri['adi']}</option>";
      ?>
    </select>
  </td>
```

Kategori İşlemleri

Kategori Ekleme ve Listeleme işlemleri tamamlandı. Şimdi de güncelleme işlemini yapalım güncelleme işlemi için her bir kategorinin yanına bir güncelle butonu ekleyeceğiz. Listeleme Fonksiyonundaki ufak bir değişiklik ile bu güncelleme ikonunu ekleyebiliriz. Görüldüğü gibi fonksiyona sadece bir satır ekledik.

```
function KategoriListele($veriDizisi, $ustKat=0){
    if(!is_array($veriDizisi))
        return "Veri Yok";

    $metin="<ul>"; //yeni bir liste açıyoruz.
    //dizideki tüm elemanları bu listeye ekleyelim.
    foreach($veriDizisi as $veri)
    {
        if($veri['ustKat']==$ustKat) //sadece üst kategori id si aynı olanları aynı listeye ekleyelim.
        {
            $metin.="<li>{$veri['adi']}";
            $metin.="<div class='guncelleBtn' title='Güncelleme İşlemi'> </div>"; //Güncelle Butonunu Ekledik.
            //fonksiyonu tekrar çağıralım
            $metin.= $this->KategoriListele($veriDizisi, $veri['id']);
            $metin.="</li>";
        }
    }

    return ($metin."</ul>"); //sonucu gönderelim
}
```

Kategori İşlemleri

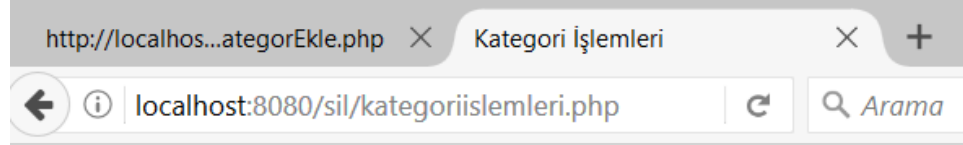
Aynı zamanda Kategoriislemeleri.php sayfasında ise güncelleme ikonlarını gösterecek style kodlarını ekledik aşağıda görülmektedir. Sayfamızda her bir kategorinin yanında çıkan güncelleme resimleri, ikonlar klasörünün içinde bulunmaktadır.










```
20 //
21 </script>
22
23 <style>
24     .guncelleBtn{
25         width:20px;
26         height:20px;
27         padding:0px;
28         background-position:0px 148px;
29         background-size: 20px;
30         background-image:url("../ikonlar/sil_duzenle.png");
31         display:inline-block;
32         margin-left: 10px;
33         cursor: pointer;
34     }
35 </style>
36 </head>
37 <body>
38     <div id="AjaxSonuc"> Ajax Sonucu Burada Gösterilecek </div>
39     <form action="" method="GET">
40         <table border="0" cellpadding="10">
41             <tr>
42                 <th>Kategori Adı </th>
43                 <td><input type="text" name="kadi"/></td>
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Kategoriislemleri.php sayfamızın son hali aşağıdaki gibidir. Güncelleme ikonları görünüyor ancak tıkladığımızda henüz çalışmıyorlar. Şimdi işlevsellik kazandıralım.



- Arabalar 
 - Ferrari 
 - Ferrari  Güncelleme İşlemi
 - Porche 
- Çiçekler 
- Evler 
- Bilgisayar 
 - Laptop 
 - Masa Üstü 

Kategori Adı

Üst Kategori Id

İkon Seç Hiçbir dosya seçilmedi.

Sıra No

Kategori İşlemleri

Kategoriislemleri.php sayfamızın son hali aşağıdaki gibidir. Güncelleme ikonları görünüyor ancak tıkladığımızda henüz çalışmıyorlar. Şimdi işlevsellik kazandıralım.

İşlevsellik için güncelleme butonuna tıkladığımızda güncellenecek olan elemanın verilerini form sayfasına yüklememiz gerekir bunun için her bir ikona link ekleyerek tıkladığımızda güncelleme modunda açılmasını saplayacağız. Listeleme fonksiyonunun son hali;

```
function KategoriListele($veriDizisi, $ustKat=0){
    if(!is_array($veriDizisi))
        return "Veri Yok";

    $metin="<ul>"; //yeni bir liste açıyoruz.
    //dizideki tüm elemanları bu listeye ekleyelim.
    foreach($veriDizisi as $veri)
    {
        if($veri['ustKat']==$ustKat) //sadece üst kategori id si aynı olanları aynı listeye ekleyelim.
        {
            $metin.="<li>{$veri['adi']}";
            $metin.="<a href='kategoriislemleri.php?gid={$veri['id']}'>
                <div class='guncelleBtn' title='Güncelleme İşlemi'> </div>
                </a>"; //Güncelle Butonunu Ekledik.
            //fonksiyonu tekrar çağıralım
            $metin.= $this->KategoriListele($veriDizisi, $veri['id']);
            $metin.="</li>";
        }
    }

    return ($metin."</ul>"); //sonucu gönderelim
}
```

Kategori İşlemleri

Kategoriislemleri.php sayfamızından eğer gid yani güncelleme id si geliyorsa formumuzu bu id ye sahip elemanın bilgileri ile dolduralım. Bu işlemlerden sonra Kategoriislemleri.php sayfamızın içeriği aşağıdaki gibi olmuştur. Değiştirilen ve güncellenen yerler kırmızı çerçeve içindedir.

```
change.log x new1 x new2 x class.php x kategorEkle.php x kategoriislemleri.php x
1 <?php
2 //Eğer Güncelleme Modundaydık yani gid ile bir id geliyorsa
3 //0 kategoriye ait Tüm verileri alıp diziyeye yükleyelim ve Formu
4 //verilerie göre dolduralım.
5
6 if(isset($_GET['gid']))
7 {
8     $guncellenecekId=$_GET['gid']+0;
9     $vt=new mysqli("localhost","root","usbw","KategoriDB");
10    if($vt->connect_error)
11        echo "Veritabanı Bağlantı Hatası";
12
13    $sql="SELECT * FROM `kategoriler` WHERE `id`=?";
14    $vt->set_charset('UTF8');
15    $sorgu= $vt->prepare($sql);
16    $sorgu->bind_param('i',$guncellenecekId);
17    $sorgu->execute();
18
19    if($vt->error)
20        echo "Sorgu Hatası ".$vt->error;
21    $sonuc=$sorgu->get_result();
22    $guncellenecekDizi=$sonuc->fetch_all(MYSQLI_ASSOC);
23        print_r($guncellenecekDizi);
24    $sorgu->close();
25    $vt->close();
26 }
27
28 ?>
```

Kategori İşlemleri

Kategoriİslemleri.php DEVAMI

Öğr. Gör. M. Mutlu YAPICI

```
<?>
<html>
  <head>
    <title>Kategori İşlemleri</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
    <script type="text/javascript">
      $(document).ready(function() {
        $("#btn").click(function() {
          $.ajax({
            type:"GET",
            url:"kategorEkle.php",
            data:$("form").serialize(),
            dataType:"HTML",
            beforeSend:function (cevap){$("#AjaxSonuc").html("İşlem Yapılıyor...");},
            success:function (cevap){$("#AjaxSonuc").html(cevap);},
            error:function () {alert("Hata Oluşturdu");}
          });
        });
      });
    </script>
    <style>
      .guncelleBtn{
        width:20px;
        height:20px;
        padding:0px;
        background-position:0px 148px;
        background-size: 20px;
        background-image:url("../ikonlar/sil_duzenle.png");
        display:inline-block;
        margin-left: 10px;
        cursor: pointer;
      }
    </style>
  </head>
  <body>
```

Kategori İşlemleri

Kategoriislemeleri.php DEVAMI

Öğr. Gör. M. Mutlu YAPICI

```
<body>
<div id="AjaxSonuc"> Ajax Sonucu Burada Gösterilecek </div>
<form action="" method="GET">
<table border="0" cellpadding="10">
<tr>
<th>Kategori Adı </th>
<td><input type="text" name="kadi" value="<?php if(isset($guncelleneckDizi[0]['adi']))echo $guncelleneckDizi[0]['adi'];?>"/></td>
</tr>
<tr>
<th>Üst Kategori Id </th>
<td>
<select name="ustKIid">
<?php
$seciliId="";
if(isset( $guncelleneckDizi[0]['ustKat']))
$seciliId= $guncelleneckDizi[0]['ustKat'];

//önce tüm verileri alıp diziye yükleyelim.
$vt=new mysqli("localhost","root","usbw","KategoriDB");
if($vt->connect_error)
echo "Veritabanı Bağlantı Hatası";

$sql="SELECT * FROM `kategoriler` ORDER BY `adi` ASC";
$vt->set_charset('UTF8');
$sorgu=$vt->prepare($sql);
$sorgu->execute();

if($vt->error)
echo "Sorgu Hatası ".$vt->error;
$sonuc=$sorgu->get_result();
$kategoriDizisi=$sonuc->fetch_all(MYSQLI_ASSOC);

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

Kategori İşlemleri

Kategoriislemeleri.php DEVAMI

Öğr. Gör. M. Mutlu YAPICI

```
if($vt->error)
    echo "Sorgu Hatası ". $vt->error;
$sonuc=$sorgu->get_result();
$kategoriDizisi=$sonuc->fetch_all(MYSQLI_ASSOC);

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

```
///şimdi de verileri açılır kutuya ekleyelim
foreach($kategoriDizisi as $veri)
{
    echo "<option value='{ $veri['id'] }'";
    echo ($seciliId==$veri['id'])? "selected":"";
    echo ">{ $veri['adi'] }</option>";
}
```

```
</select>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<th>İkon Seç </th>
```

```
<td><input type="file" name="ikon" value="<?php if(isset( $guncelleneckDizi[0]['ikon']))echo $guncelleneckDizi[0]['ikon'];?>" /></td>
```

```
</tr>
```

```
<tr>
```

```
<th>Sıra No </th>
```

```
<td><input type="text" name="sira" value="<?php if(isset( $guncelleneckDizi[0]['siraNo']))echo $guncelleneckDizi[0]['siraNo'];?>" />
```

```
</tr>
```

```
<tr>
```

```
<td colspan="2">
```

```
<input type="hidden" id="guncel" name="guncelleId" value="<?php echo (isset($guncelleneckDizi[0]['id']))?$guncelleneckDizi[0]['id']:"
```

```
<input type="button" id="btn" value="<?php echo (isset( $guncelleneckDizi[0]['adi']))?"Kategori Güncelle":"Kategori Ekle";?>" /></td>
```

```
</tr>
```

```
</table>
```

```
</form>
```

Kategori İşlemleri

Şimdi eğer güncelleme butonuna tıklarsak güncelleme işlemi yapsın. Güncelleme butonuna tıklayıp tıklamadığımızı, hidden olarak ayarladığımız guncelleld isimli textbox verisinden anlayacağız eğer guncelleld ile get verisi geliyorsa güncelleme yapıyoruz demektir. Şimdi KategorEkle.php içerisinde güncelleme işlemlerimizi yapalım. Tabiki öncelikle güncelleme fonksiyonumuzu sınıfımızın içerisine ekleyeceğiz. Güncelleme Fonksiyonu aşağıdaki gibidir;

```
function KategoriGuncelle($guncellenecekId,$adi,$ustKategori,$iconUrl,$siralama){
    if(empty($guncellenecekId) || empty($adi))
        return false;

    $vt=new mysqli("localhost","root","usbw","KategoriDB");
    if($vt->connect_error)
        return "Veritabanı Bağlantı Hatası";
    $sql="UPDATE `kategoriler` SET `adi`=?,`ustKat`=?,`ikon`=?,`siraNo`=? WHERE `id`=?";

    $vt->set_charset('UTF8');
    $sorgu=$vt->prepare($sql);
    $sorgu->bind_param('sssii',$adi,$ustKategori,$iconUrl,$siralama,$guncellenecekId);
    $sorgu->execute();

    if($vt->error)
        return "Sorgu Hatası ".$vt->error;

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

    return "Kategori Başarıyla Güncellendi";
}
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

kategoriEkle.php dosyamızın içerisinde de değişikli yaptık son hali;

```
1 <!--PHP
2     include_once("class.php");
3     $kategoriSinifi = new Kategori();
4
5     $guncellemeMi=false;
6     ////KATEGORİYE GÜNCELLEME İŞLEMLERİ
7     if(!empty($_GET['guncelleId']))
8         if($_GET['guncelleId']!="bos")
9             $guncellemeMi=true;
10
11     if($guncellemeMi)
12     {
13         print_r($_GET);
14         $kadi=$_GET['kadi'];
15         $gid=(isset($_GET['guncelleId']))?$_GET['guncelleId']:null;
16         $ustKId=(isset($_GET['ustKId']))?$_GET['ustKId']:0;
17         $ikon=(isset($_GET['ikon']))?$_GET['ikon']:"";
18         $siraNo=(isset($_GET['sira']))?$_GET['sira']:0;
19         $sonuc=$kategoriSinifi->KategoriGuncelle($gid,$kadi,$ustKId,$ikon, $siraNo);
20         echo "<h3 style='color:red;'> $sonuc </h3>";
21
22     }else if(!empty($_GET['kadi'])) ////KATEGORİYE EKLEME İŞLEMLERİ
23     {
24         $kadi=$_GET['kadi'];
25         $ustKId=(isset($_GET['ustKId']))?$_GET['ustKId']:0;
26         $ikon=(isset($_GET['ikon']))?$_GET['ikon']:"";
27         $siraNo=(isset($_GET['sira']))?$_GET['sira']:0;
28         $sonuc=$kategoriSinifi->KategoriEkle($kadi,$ustKId,$ikon, $siraNo);
29         echo "<h3 style='color:red;'> $sonuc </h3>";
30     }
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Evet Artık Güncelleme işlemlerimiz de çalışıyor










http://localhost:8080/sil/kategoriislemleri.php?gid

← localhost:8080/sil/kategoriislemleri.php?gid

Kategori İşlemleri

Arama

Kategori Başarıyla Güncellendi

- Arabalar_ 
 - Ferrari_ 
 - Porche_ 
 - Meseratti_ 
- Çiçekler_ 
- Evler_ 
- Bilgisayar_ 
 - Laptop_ 
 - Masa Üstü_ 

Kategori Adı

Üst Kategori Id

İkon Seç

Sıra No

Kategori İşlemleri

Şimdi de silme ikonumuzu ekleyip silme işlemleri yapalım. Güncelleme de olduğu gibi listeleme fonksiyonumuzun içerisinde değişiklik yapıp silme ikonumuzu ekleyelim.

Silme ikonumuzla güncelle ikonumuz aslında sitil bakımından aynı bu sebeple aynı sitil kodlarını tekrar yazmaya gerek yok sadece resim farklı, aynı olanları ortak bir sitilde birleştirelim.

```
function KategoriListele($veriDizisi, $ustKat=0){
    if(!is_array($veriDizisi))
        return "Veri Yok";

    $metin="<ul>"; //yeni bir liste açıyoruz.
    //dizideki tüm elemanları bu listeye ekleyelim.
    foreach($veriDizisi as $veri)
    {
        if($veri['ustKat']==$ustKat) //sadece üst kategori id si aynı olanları aynı listeye ekleyelim.
        {
            $metin.="<li>{$veri['adi']}";
            $metin.="<a href='kategoriislemleri.php?gid={$veri['id']}'>
                <div class='butonortaksitiller guncelleBtn' title='Güncelleme İşlemi'> </div>
            </a>"; //Güncelle Butonunu Ekledik.

            $metin.=" <div class='butonortaksitiller silBtn' title='Silme İşlemi'> </div>"; //Silme Butonunu Ekledik.
            //fonksiyonu tekrar çağıralım
            $metin.= $this->KategoriListele($veriDizisi, $veri['id']);
            $metin.="</li>";
        }
    }

    return ($metin."</ul>"); //sonucu gönderelim
}
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Kategoriislemeleri.php deki sitil kodlarını da biraz değiştirdik.



















```
46         // ;
47
48     });
49 </script>
50 <style>
51     .butonortaksitiller{
52         width:20px;
53         height:20px;
54         background-image:url("../ikonlar/sil_duzenle.png");
55         padding:0px;
56         background-size: 20px;
57         display:inline-block;
58         margin-left: 10px;
59         cursor: pointer;
60     }
61     .guncelleBtn{
62         background-position:0px 148px;
63     }
64     .silBtn{
65         background-position:0px 0px;
66     }
67 </style>
68 </head>
69 <body>
70 <div id="AjaxSonuc"> Ajax Sonucu Burada Gösterilecek </div>
71 <form action="" method="GET">
72 <table border="0" cellpadding="10">
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Sayfamızın son hali görünüm olarak tamam ancak sil butonuna tıklayınca silmiyor. Son olarak silme işlemini de gerçekleştirelim.

← ⓘ localhost:8080/sil/kategoriislemleri.php?gid ↻ 🔍 Arama

- Arabalar_  
 - Ferrari_  
 - Porche_  
 - Meseratti_  
- Çiçekler_  
- Evler_  
- Bilgisayar_  
 - Laptop_  
 - Masa Üstü_  

Silme İşlemi

Kategori Adı

Üst Kategori Id

İkon Seç

Sıra No

Kategori Güncelle

Kategori İşlemleri

Silme ikonuna tıkladığımızda ajax fonksiyonunu çağırırız ve bu fonksiyonda idsi belli olan elemanımızı silsin. Tabii her kategori için silinecek olan kategori idsini butonumuza vermemiz gerekir, böylece hangi butona tıklarsak o butonun kategori idsini, ajax ile sunucuya gönderilir ve silinir. Ben butonların alt isimli attribute (özelligi) de kategori idlerini saklayacağım. Şu halde kategoriislemeleri.php sayfamızdaki silme ajax fonksiyonu aşağıdaki gibi oldu;

```
g x new 1 x new 2 x class.php x kategorEkle.php x kategoriislemeleri.php x
type:"GET",
url:"kategorEkle.php",
data:$("form").serialize(),
dataType:"HTML",
beforeSend:function (cevap){$("#AjaxSonuc").html("İşlem Yapılıyor...");},
success:function (cevap){$("#AjaxSonuc").html(cevap);},
error:function (){alert("Hata Oluşturdu");}
});
});

$("div").on("click", ".silBtn", function(){
    var idim=$(this).attr('alt');
    $.ajax({
        type:"GET",
        url:"kategorEkle.php",
        data:{sil:"katsil",sid:idim},
        dataType:"HTML",
        beforeSend:function (cevap){$("#AjaxSonuc").html("İşlem Yapılıyor...");},
        success:function (cevap){$("#AjaxSonuc").html(cevap);},
        error:function (){alert("Hata Oluşturdu");}
    });
});
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Sınıfımızın içerisine de bir kategori silme fonksiyonu ekleyelim.

```
75         return "Kategori Başarıyla Güncellendi";
76     }
77
78
79     function KategoriSil($silinecekId) {
80         if(empty($silinecekId))
81             return false;
82
83         $vt=new mysqli("localhost","root","usbw","KategoriDB");
84         if($vt->connect_error)
85             return "Veritabanı Bağlantı Hatası";
86         $sql="DELETE FROM `kategoriler` WHERE `id`=?";
87
88         $sorgu=$vt->prepare($sql);
89         $sorgu->bind_param('i',$silinecekId);
90         $sorgu->execute();
91
92         if($vt->error)
93             return "Sorgu Hatası ".$vt->error;
94
95         $sorgu->close();
96         $vt->close();
97
98         return "Kategori Başarıyla Silindi";
99     }
100
101
102 }
103 ?>
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

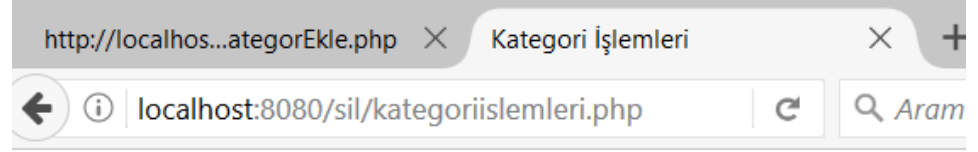
Son olarak silme işlemini yapacağımız fonksiyonumuz çağırdığımız kategoriEkle.php sayfamızın son hali:

```
change.log x new 1 x new 2 x class.php x kategorEkle.php x kategoriislemleri.php x
1 <?php
2     include_once("class.php");
3     $kategoriSinifi = new Kategori();
4
5     $silmeMi=false;
6     ///KATEGORİ SİLME İŞLEMLERİ
7     if(isset($_GET['sil']))
8         if(isset($_GET['sid']) && $_GET['sil']=="katsil")
9             $silmeMi=true;
10
11 $guncellemeMi=false;
12 ///KATEGORİYE GÜNCELLEME İŞLEMLERİ
13 if(!empty($_GET['guncelleId']))
14     if($_GET['guncelleId']!="bos")
15         $guncellemeMi=true;
16
17 if($silmeMi)
18 {
19     $silinecekId=$_GET['sid'];
20     $sonuc=$kategoriSinifi->KategoriSil($gid,$kadi,$ustKId,$ikon, $siraNo);
21     echo "<h3 style='color:red;'> $sonuc </h3>";
22
23 }else if($guncellemeMi)
24 {
25     $kadi=$_GET['kadi'];
26     $gid=(isset($_GET['guncelleId']))?$_GET['guncelleId']:null;
27     $ustKId=(isset($_GET['ustKId']))?$_GET['ustKId']:0;
28     $ikon=(isset($_GET['ikon']))?$_GET['ikon']:"";
29     $siraNo=(isset($_GET['sira']))?$_GET['sira']:0;
30     $sonuc=$kategoriSinifi->KategoriGuncelle($gid,$kadi,$ustKId,$ikon, $siraNo);
31     echo "<h3 style='color:red;'> $sonuc </h3>";
32
```

Kategori İşlemleri

Öğr. Gör. M. Mutlu YAPICI

Son olarak silme işlemini yapacağımız fonksiyonumuz çağırdığımız kategoriEkle.php sayfamızın son hali;



Kategori Başarıyla Silindi

- Arabalar_  
 - Ferrari_  
 - Meseratti_  
- Çiçekler_  
- Evler_  
- Bilgisayar_  
 - Laptop_  
 - Masa Üstü_  

Kategori Adı

Üst Kategori Id

İkon Seç

Sıra No

Kategori Ekle

KAYNAKLAR

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

