



FIRAT ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ



ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

BİTİRME ÖDEVİ

DİNAMİK WEB TASARIMI

HAZIRLAYANLAR :

- . GONCA ÖCALAN
- . NAZLI ÇITAK
- . NESLİHAN ŞAHİN

KONTROL EDEN :

YRD.DOÇ.DR. HASAH H. BALIK

İÇİNDEKİLER

1. ASP(ACTİVE SERVER PAGE).....	3
2. ASP ile IIS İlişkisi.....	4
3. ASP Dosyalarının işletilmesi.....	6
4. IIS Kurulumu.....	6
5. IIS'e Erişim.....	8
6. Script Dilinin Belirlenmesi.....	9
7. ASP 3.0 'da Bulunan Yapı Nesneleri.....	10
8. SQL Komutları İle Veri sorgulama.....	13
9. Yayın Sistemi Otomasyonu.....	20

Teşekkür

Bilişim alanında yaşanan baş döndüren gelişmeler internet teknolojisinde de kendinin göstermekte ve kullanımı da aynı oranda gün geçtikçe artmaktadır.Bu hızlı gelişmeler internete yönelik programcılığı da teşvik etmekte ve günümüzde web sayfası tasarımı da önem kazanmaktadır.Yaptığımız ödevle dinamik web tasarımı konusunda bilgi vermeyi amaçlıyoruz.

Bu ödevin hazırlanmasında bize yol gösteren Değerli Hocamız Hasan H.BALIK'a ,hazırlık aşamasında bizden maddi ve manevi desteklerini esirgemeyen sevgili Ailelerimize teşekkür'ü bir borç biliyoruz.

ASP (ACTIVE SERVER PAGE)

World Wide Web (www) bir dizi deney sonucu olarak CERN laboratuvarlarında uzun süren çalışmalar sonucunda geliştirilmiştir. Tim Berners-Lee nin başkanlığını yaptığı geliştirme grubu internet üzerinden dataların uzak yerlere iletilmesi üzerine yaptıkları araştırmalar sonucunda HTTP denilen protokolü oluşturmuşlardır. Ve bu protokolle birlikte HTML denilen basit ve esnek olan programlama dili bulunmuştur. Bu dil web browserler üzerinde metin ve grafik görüntülemeye imkan veriyordu. Ve yine bu dille (HTML) sayfalar arasında geçişe olanak tanıyan “Link” deyimide doğmuş oldu.

Bu yeni başlayan akım hızla geliştirilerek şu anda kullanılmakta olan HTML 4.0 versiyonuna kadar ilerlemiştir. Günümüzde kullanılan HTML ilk bulunana göre bir çok ekstra özellik içermektedir. Bu özellikler içinde en önemli olanı esnek metin biçimlendirmedir. (elementi yardımı ile) ikinci olarak sayfa yapısının tamamen esnek şekilde yapılandırılmasıdır (Tablo ve Frame’ler kullanılarak).

HTML dilinin standartlaşması ile metinler, grafikler, resimler ve diğer tüm belgeler network üzerinden diğer kullanıcılar ile paylaşımaya başlandı. Fakat içerik yapısı tamamen sabitti ve herşey HTML içinde tanımlıydı değiştirilemezdi. Server’ın (sunucu) sunduğu dosyaları istemci (client) işletim sistemi farklılığı olmadan işleyebiliyordu.

HTML in gelişmesi ile birlikte kullanıcılara web sayfalarına bilgi girebilmelerine olanak tanıdı (<input> elementi yardımı ile). Bu şekilde bir çok uygulama geliştirildi çünkü artık kullanıcı da sunucuya bilgi gönderebiliyordu. Fakat çoğu uygulamada bu kullanıcıdan gelen bilgilerin anında işlenmesi ve yeniden bir text bazlı HTML dökümanı haline getirilmesi gerekiyordu. Bu ise hiç hızlı bir yöntem değildi.

Bu zorluğu aşmak isteyen geliştiriciler kolları sıvıyarak CGI (Common Gateway Interface) arabirimini geliştirmeye başladılar. Bu arabirimi standart haline getirdiler ve tamamen “C” dili üzerine kurdular. Cgi-bin dizinide bu şekilde doğmuştur (“bin” terimi derlenmiş “C” kodu olmasından dolayı “binary code” anlamında eklenmiştir). İlk uygulamalar derlenmiş ufan programcıklar halinde olmuştu. Fakat bu haliyle bile kullanışlı değildi çünkü dosya içinde yapılacak en ufak değişiklikte bile yeniden derlenmesi gerekmekteydi. Buda CGI in kullanımını olumsuz yönde etkiliyordu.

Bu kısıtlamaları kendine sınır olarak görmeyen geliştiriciler yeni bir script dili geliştirdiler. Bu dil “Practical Extraction and Reporting Language” yani PERL adını aldı. Bu dil sunucu ile iletişim halinde olan ilk dildi yani “C veya C++” dilleri ile yazılan scriptin her seferinde derlenmesi derdi ortandan kaldırılmış oldu.

Perl hala popüler bir dil olarak çoğu uygulamada özellikle de Unix ve Linux tabanlı sistemlerde kullanılmaktadır.

Server-Tabanlı Script Teknolojileri

CGI dilleri web sunucusu üzerine bir yama yapmadan yada ekstra bir program yüklemeyen çalışmamaktadır. Bu programlar kullanıcıdan gelen isteği algılar ve isteğe göre dosyayı okur daha sonra onu sunucu içinde işler ve bir çıkış dosyası oluşturarak kullanıcıya sunarlar.

Perl ilk popüler sunucu-tabanlı uygulama geliştirme dili olarak literatüre geçmiştir. Fakat bu dil çok fazla gelişim geçirmiştir. Özellikle de Unix ve Linux tabanlı sunucularda yeni nesil programlama dili olan PHP

(Personal Home Page) ye yerini bırakmıştır.

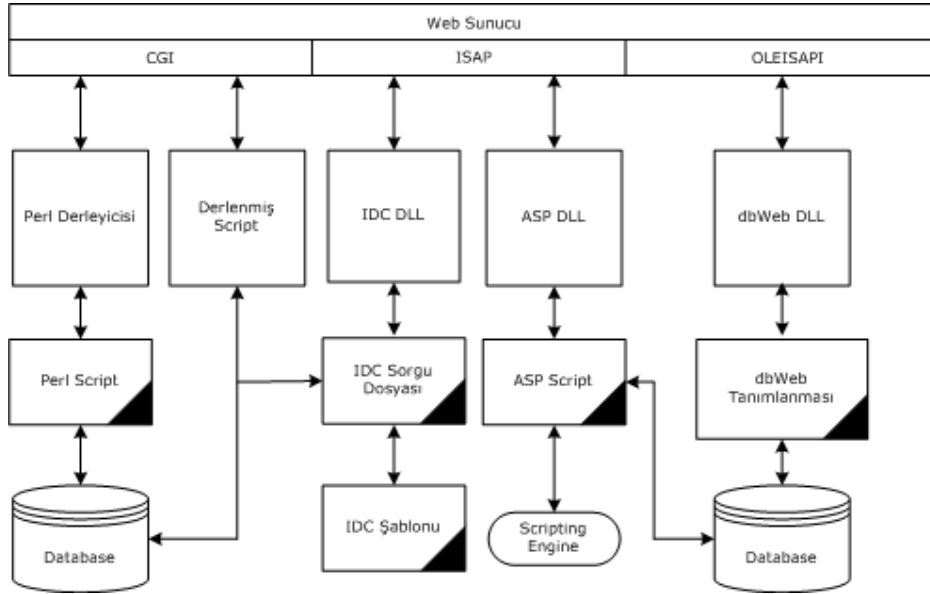
Microsoft firması web sunucu sektöründeki en önemli atılımını “Windows NT 3.51” ve bu işletim sistemine entegre halde olan “Internet Information Server 1.0” (IIS) sayesinde yapmıştır. Bu yazılım geçmişe dönük olarak CGI desteklemekle (her şekilde C ve C++ dili ile geliştirilmiş uygulamaları) birlikte yeni bir arabirim içeriyordu.

Bu arabirime “Internet Server Application Programing Interface” yani ISAPI adı verilmiştir. Bu arabirim sayesinde web sunucu perl dilinin tüm esnekliği standart hale getirilmiş oldu. Bu atılımla

birlikte tüm yazılım geliştiriciler ISAPI ile uyumlu olan yazılımlar geliştirmeye başladılar.

Microsoft ISAPI ile beraber yeni teknoloji olan ASP’yi duyurdu. ASP teknolojisi IIS ile ISAPI sayesinde bağlanmış oldu. ASP den önce en çok “Internet Database Connector” (IDC) kullanılmaktaydı.

Aşağıdaki diyagramda microsoft server yapısı ayrıntılı olarak anlatılmıştır.

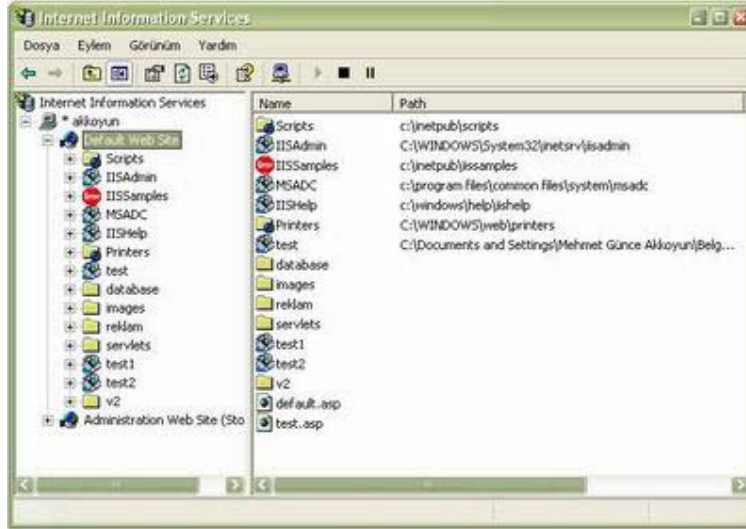


ASP ile IIS ilişkisi

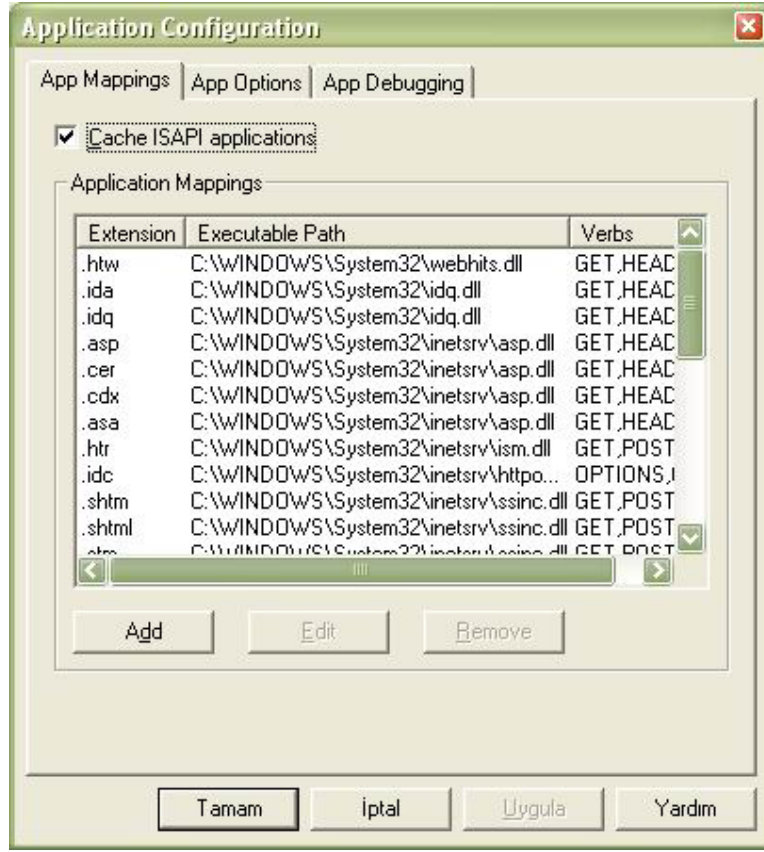
ASP sadece kendisi için yazılmış olan DLL i kullanır (asp.dll). Bu dosya standart olarak web sunucu da yer almaktadır (sadece IIS 1.0 sonrası) (Winnt\System32\inetsrv dizininde yer almaktadır). Bu DLL sadece ASP dosyalarını (genellikle .ASP uzantılıdır) okuyup içerisindeki script komutlarını işlemek ve sonuçlarını HTML ve metin içeriği ile birlikte Web gözeticisine yollamak görevini üstlenir.

IIS Uygulama Yapıları

IIS içerisindeki işlemleri daha iyi anlayabilmek için uygulama yapılarının windows içinde nasıl çalıştığını anlamalıyız. Web sunucudaki (IIS) her web sitesinin sunucu üzerinde yer alan bir kök dizini vardır. Varsayılan (Default) web sitesi otomatik olarak c:\inetpub\wwwroot dizinini kendine kök dizini atar (değiştirilebilir). Her yeni açılacak web sitesi için bir kök dizini belirlenmesi zorunludur. Sunucu üzerindeki web sitelerini görmek için IIS yönetim arabirimi olan “Internet Service Manager” programı kullanılır.



IIS üzerinden default web sitesine sağ tıklayıp özellikler seçeneğini seçtiğimiz zaman, açılacak olan “default web site özellikleri” ekranından “home directory” tabına tıklanır. Gelecek olan sekmeden “configuration” butonuna basarak uzantılar ile ilişkilendirilen arabirimler görülebilir.



Resimden de görülebileceği gibi asp uzantılı dosyalar asp.dll dosyası ile derlenmektedir. HTM ve HTML uzantılı html sayfaları ve XML uzantılı xml sayfaları direkt olarak diskten okunup (web sunucu tarafından) istemciye gönderilmektedir ama asp uzantılı dosyalar ISAPI yardımı ile asp.dll tarafından okunup derlenip sonuç çıktıları istemciye gönderilmektedir.

ASP dosyalarının işletilmesi

Asp uzantılı dosyaların asp.dll yardımı ile derlendiğini bilmekteyiz peki ya bu derlenme nasıl olmaktadır.

Birinci basamak olarak asp dosyası içerisinde server taraflı kod olup olmadığı denetlenir. Eğer dosya içerisinde sunucu taraflı işletilecek bir kod bulunmaz ise IIS tarafından direkt istemciye gönderilir. Bu windows 2000 de yeni bir özellik olarak eklenmiştir. Bu sayede .asp uzantılı dosyaları kullanmamıza olanak tanır (içerisinde sunucu taraflı çalışacak kod bulunmayan dosyalara bile asp uzantısı verilebilmektedir).

Eğer DLL içinde server taraflı çalışacak bir kod olan dosya ile karşılaşırsa; satır satır bu dosyayı işleterek script blokları içerisindeki komutları işletir ve çıktısını yine aynı yere gelecek şekilde bir çıkış dosyasına kaydeder. Bu dosyada istemci bilgisayarına gönderilerek işlem tamamlanır.

Neden IIS?

1. PWS, hepimizin bildiği gibi Windows 98 üzerinde çalışmaktadır. Windows ME ile beraber PWS gelmemektedir. Birkaç makina üzerinde yapmış olduğum denemelerde Windows 98 CD'si içerisinde yeralan PWS'yi Windows ME üzerinde kurduğumda hiçbir problemle karşılaşmadım. Ancak düzgün çalıştıramayan ve problem yaşayanların varlığını biliyorum

2. Belki de çoğumuz ilk ASP kodlarını PWS üzerinde yazmıştır. Ancak Windows 98 ve PWS bize gerçek server ortamını sağlamadıkları için çoğu zaman yazdığımız kodları servera gönderdiğimiz zaman çalışmadığını, hata verdiğini görmüştür.

3. Windows 98 bir server işletim sistemi olmadığı için, kullandığımız bileşenler (component) ile ilgili hatalar ve problemler oluşabilmektedir.

4. Eğer ASP ile sadece hobi olarak uğraşmıyorsanız, çalıştığımız ortamlarda Windows 2000 Server ile meşgul olacaksınız demektir. Bu nedenle Windows 2000 Server üzerinde çalışmak yararlı olacaktır.

5. Server üzerinde çalışacak olan ASP kodlarının tüm geliştirme aşamasında benzer bir ortamda çalışmak size oluşabilecek hataları daha kolay görme, anında çözümler üretebilme gibi avantajlar sağlayacaktır. Yazdığınız kodun IIS üzerinde nasıl bir tepki vereceğini programlamayı tam olarak bitirmeden görme şansına sahip olabileceksiniz.

Yukarıdaki sebeplerden dolayı kodların, eğer imkanları müsaitse Windows 2000 Server üzerinde veya en azından Windows 2000 Professional Edition üzerine IIS kurarak (kurulum sırasında default olarak yüklenmez) yazmaları tavsiye edilir.

KURULUM

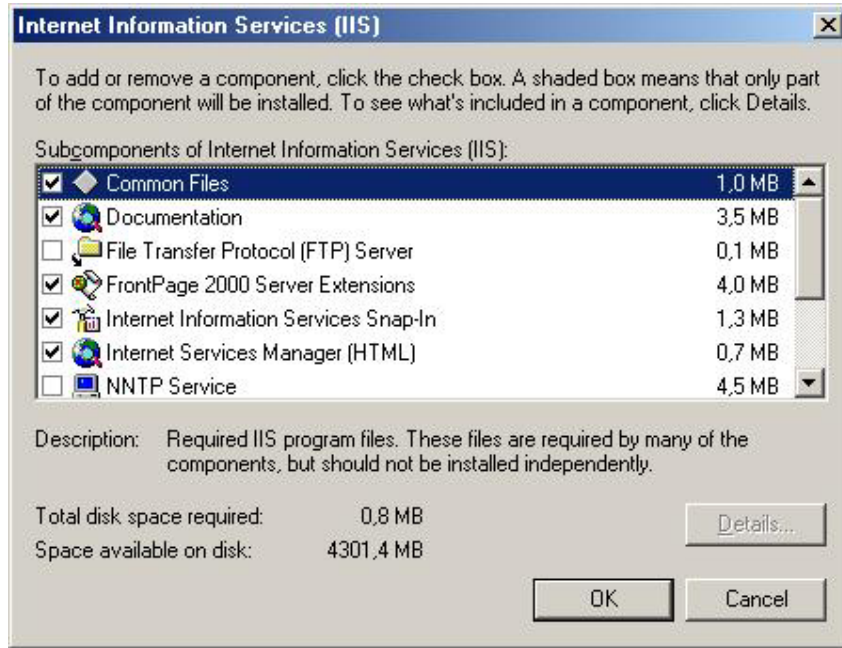
Windows 2000 Server üzerinde yüklenmemiş olma ihtimali ve Professional üzerine kurulması gerektiği için nasıl olduğunu bilmek amacıyla kısaca IIS'in kurulumundan bahsetmek istiyorum :

Start/Settings/Control Panel'e (Başlat/Ayarlar/Denetim Masası) tıklayarak **Control Panel'i** (Denetim Masası) açıyoruz. Control Panel'de **Add Remove Programs** (Program Ekle/Kaldır) linkine çift tıklayarak açılan pencerede "Add/Remove Windows Components"i (Windows Bileşenleri Ekle/Kaldır) tıklıyoruz. Karşınıza gelecek olan yeni pencerede windows bileşenlerini görebilirsiniz :



IIS linki üzerine çift tıklayarak veya "Details" linkine basarak IIS içerisinde yer alan bileşenleri (server ve servisler) görebilir, ihtiyacınız olanları işaretleyerek kurulumlarını sağlayabilirsiniz. Sıra ile bunlardan

bahsedelim :



IIS'E ERİŞİM

Web-FTP sitesi tanımlamak ve ayarlarını yapmak için IIS'e birkaç farklı yoldan ulaşabilirsiniz :

1. Start/Settings/Control Panel/Administrative Tools/Internet Services Manager yolunu izleyerek,
2. Start/Programs/Administrative Tools/Internet Services Manager yolunu kullanarak,
3. "My Computer"e sağ tıklayıp, "Manage" seçeneğine basarak açılacak olan "Computer Management" penceresinde "Services and Applications" bölümünden IIS'e ulaşabilirsiniz.

ASP ile IIS ilişkisi

Asp script motoru (asp.dll) dosya içerisinde sunucu taraflı çalışacak kod olup olmadığını iki şekilde anlar:

<% ve %> kullanarak

En çok kullanılan yöntem olup script bloğu başlangıcına "<%" ve sript bloğu bitişiine "%>" yazılarak arasında kalan kısma sunucu taraflı kod yazılır.

```
<HTML>
<Body>
Bu bir HTML metnidir
<%
Rem burasi sciprt bloğudur
```

```
%>
</Body>
</HTML>
```

<script> elementini kullanmak

Nadir olarak kullanılan bu yöntem ile sunucu taraflı kodun yer aldığı script bloğunu <script> elementi ile açıp </script> elementi ile bitirilmesi baz alınmıştır. Bu şekilde ki kullanımda element içerisine yazılacak olan “Runat” özelliği sayesinde istemci veya sunucu taraflı çalışma özelliği eklenmiştir.

```
<HTML>
<Body>
Bu bir HTML metnidir
<script runat="server">
Rem burasi script bloğudur
</script>
</Body>
</HTML>
```

Script Dilini Belirlemek

IIS standart olarak iki script motoru (scripting engine) ile beraber gelir. Bunlar “VBScript” ve “Jscript” dir. Bu motorlar birlikte bulunurlar. Bunlar dışında TCL ve PerlScript gibi diğer script motorları da mevcuttur fakat bunlar IIS ile beraber gelmez sadece sonradan eklenir.

ASP ye biz hangi script motorunu kullanması gerektiğini söyleyebiliriz. Bu genelde standart olarak IIS de ayarlanmış haldedir. Bu tanımlı yapmanın en kolay yolu ASP sayfamızın ilk satırında özel içerik tanımlı yapmaktır. Aşağıdaki şekilde yapılır.

```
<%@Language = "dil"%>
```

Bu şekilde sayfanın VBScript mi yoksa Jscript mi kullanacağı tanımlanmış olur. VBScript için

```
<%@Language = "VBScript"%>
```

yazabiliriz. Aynı Mantıkta Jscript için

```
<%@Language = "JScript"%>
```

Bu tanımlama yapılsın yada yapılmıy script bloğumuzu <script> elementi ile tanımlamışsak istediğimiz dili oradada tanımlayabiliriz.

```
<HTML>
<Body>

Bu bir HTML metnidir

<script Runat = "server" Language = "VBScript">
Rem burasi scriprt bloğudur ve dil olarak VBScript Kullanılmıştır
</script>

<script Runat = "server"
Language = "JScript">
Rem burasi scriprt bloğudur ve dil olarak JScript Kullanılmıştır
</script>

</Body>
</HTML>
```

ASP temelde VBScript'i kullanır ve VBScriptte geçen tüm fonksiyonlar, Operatörler, Metodlar ve diğer işlemler aynı şekilde ASP içinde geçerlidir.

Burada aklınıza "O zaman neden VBScript kullanmıyoruzda ASP kullanıyoruz ?" diye bir soru gelebilir. Bunu ASP ile VBScript farkları olarak şöyle açıklayabiliriz.

- ASP Server tabanlı bir script dilidir ve Webserver olmadan çalışmaz. VBScript ise normal HTML sayfalarında da kullanılabilir.
- ASP hem VBScript in getirdiği özellikleri kullanabilir hem de objelerle ilişkilidir. VBScript ise ASP kadar obje yönelimli değildir.
- ASP nin tabanında yapı objeleri (Built-In Objects) vardır fakat VBScriptte bu şekilde objeler yoktur.
- ASP , VBScript ' e göre veritabanları ile daha kolay anlaşılabilir ve daha çarpıcı işlemler yapabilir.
- ASP de yazdığınız kodlar kullanıcının bilgisayarına gittiğinde HTML tagları olarak görünür fakat VBScript kodları kullanıcının bilgisayarına gittiğinde yazıldığı gibi görünür.

Bu farkları gördüğümüzde neden ASP kullandığımızı daha iyi anlıyoruz.

ASP 2.0 da 5 adet yapı objesi bulunurken ASP 3.0 da bu sayı 7 ye çıkarılmıştır.

ASP 2.0 da bulunan yapı objeleri :

- Response
- Request
- Session
- Application
- Server

ASP 3.0 da bulunan yapı objeleri :

- Response
- Request
- Session
- Application
- Server
- Errors Object
- ObjectContext Object

ASP 3.0 ile birlikte hata kontrolleri için **Errors objesi** ve Transaction işlemleri için **ObjectContext** objesi

gelmiştir.

RESPONSE :

Bu obje ASP’de en çok kullanılan objelerden biridir. Webserver’ın kullanıcıya (Client) cevap vermesi anlamına gelir.

Yani kullanıcı Webserver dan bir istekte bulunur ve Webserver da kullanıcıya Response objesi ile cevap verir.

Örneğin, kullanıcı herhangi bir ASP sayfasını çağırdığında ekrana gelen değişken değerleri veya kişinin bilgisayarına çerezlerin (Cookie) gönderilmesi gibi işlemler bu obje sayesinde olur.

Response.Write :

Bu metod ASP sayfalarında çok fazla kullanılır. Bir değişkenin değerini yada herhangi bir yazıyı ekranda görüntülemek için kullanılır.

Örnek kullanımı : Response.write (“ bu bir deneme yazısıdır...”)

şeklindedir. Bu örneği kullandığınızda ekranda “ bu bir deneme yazısıdır...” yazısını görürsünüz.

Bunun dışında tek satırlık işlemler yapıyorsanız “ = “ işareti kullanabilirsiniz.Bu işlem Response.Write metodunun yaptığı işi yapar. Örnek kullanımı : <%=data(“kisiadi”)%> Şeklindedir.

Response.Redirect :

Bu metod bulunduğunuz sayfadan başka bir sayfaya yada başka bir internet adresine yönlendirilmek istediğinizde size yardımcı olacaktır.Örneğin bir kişiyi sitenize kayıt ettikten sonra otomatikman o kişiyi kayıt olmuş kişiler sayfasına yönlendirmek isterseniz redirect metodunu kullanabilirsiniz.

Örnek kullanımı : Response.Redirect (“default.asp”)

şeklindedir. Bu örneği kullandığınızda bulunduğunuz sayfadan default.asp sayfasına yönlendirilirsiniz.

Response.Expires :

Bu özellik sayfanın kaç saniye sonra iptal (Expire) olacağını belirler. Örneğin kullanıcı hiçbir işlem yapmadan 20 saniye beklerse sayfa expire olur.tabi burada 20 saniye yada diğer zaman birimlerini belirlemek size kalmış.

Örnek kullanımı : Response.Expires=50

Şeklindedir. Burada 50 rakamı sayfanın 50 saniye sonra expire edileceğini belirtmektedir.

Response.Cookies :

Bu koleksiyon Ziyaretçinin bilgisayarına Cookie (Çerez) denilen, genelde içinde kişi ve site hakkında bilgi barındıran küçük bilgi dosyacıkları gönderir.Cookieler siteler tarafından fazlaca kullanılır ve kullanıcıyı tekrar tekrar sitede bir takım olayları gerçekleştirmekten kurtarır. Örneğin kişi bilgilerini içeren bir Cookie niz varsa ziyaretçi sizin sitenize girdiğinde

O kişiye ismi ile hitap edebilirsiniz ve bunun o kişiyi memnun edeceğinden emin olabilirsiniz.

Örnek kullanımı : Response.Cookies(“isim”)=”ismail AKTAŞ”

Şeklindedir. Bu Örneği kullandığınızda ziyaretçinin bilgisayarına isim adında ismail AKTAŞ değerini içeren bir cookie göndermiş olursunuz .

Response.IsClientConnected :

Bu özellik sitenizde herhangi ziyaretçinin olup olmadığı hakkında size bilgi vermektedir. Eğer sitenizde ziyaretçi var ise geriye değer olarak true döndürmektedir.

Örnek kullanımı : Response.IsClientConnected

Şeklindedir. Burada size döndürecek True , False değerlerinden birine göre işlemler yapabilirsiniz.

Response.Buffer :

Bu özellik sayfanın tam olarak yüklenmeden gösterilmemesini sağlar. Yani bu özelliği kullandığınızda sayfanızdaki objeler,elemanlar ve diğer tüm işlemlerin oluşması tamamlanmadan sayfanız ekranda görünmez.

Örnek kullanımı : Response.Buffer=True

Şeklindedir. Burada True değerini girmezseniz yada buffer özelliğini hiç kullanmazsanız mevcut değeri olan False aktif olacaktır.

Response.Charset :

Bu özellik bölgesel ayarlarda kullanılır ve sayfada kullanılan karakterlerin set edilmesi işlemini içerir.

Örnek kullanımı : Response.Charset=1254

Şeklindedir.Buradaki 1254 değeri türkçe karakterlerin yazılacağını belirtir.

Response.End :

Bu metodu sayfanın neresine koyarsanız kodlar oraya kadar normal bir şekilde çalışır ve response.end satırını gördüğü anda durur. Bu metod daha çok kodlama sırasında kontrol işlemlerinde kullanılır fakat bazı sayfalarda da işlemleri sonlandırmak için kullanıldığı görülmüştür.

Örnek kullanımı : Response.End şeklindedir.

Response.ContentType :

Kullanacağınız ASP sayfasının içeriğini tanımlamakta kullanılır.Örneğin Browser' ınızda bir Excel,Word yada bir PowerPoint sayfasını ContentType özelliğini kullanarak açabilirsiniz.

Örnek kullanımı : Response.ContentType="application/msword" yazdığınızda browser bir Word sayfası açar.

Eğer Excel sayfası açmak isterseniz ; Response.ContentType = "application/vnd.ms-excel" yazarak bu işlemi de yapabilirsiniz.

REQUEST :

Request objesi Webserver ' ın ziyaretçiden bilgi alması anlamına gelmektedir.

Request.QueryString :

Formlarda GET metodu ile bilgi gönderilmek istediğinde bu bilgi request objesinin QueryString koleksiyonu ile alınır.Ayrıca linkler aracılığı ile bilgi gönderilmek istendiğinde de bu bilgi GET metodu ile gönderilmiş sayılıp yine QueryString koleksiyonu ile alınır.

Örnek kullanımı : Response.QueryString("formdan gelen bilgi") Şeklindedir.

Request.Form :

Bu koleksiyon ise QueryString in yaptığı işlemlerinin bir benzerini yapmaktadır. Tek farkı formlarda GET metodu yerine POST metodu aktif olduğunda kullanılır ve linkler aracılığıyla gönderilen veriler bu koleksiyon ile alınmaz.

Örnek kullanımı : Response.Form("formdan gelen bilgi") Şeklindedir.

Request.ServerVariables :

Bu koleksiyon server bilgilerini almak için kullanılır. Webserver'ların tuttuğu bilgilere ihtiyacımız olduğunda bu koleksiyonu kullanırız.

Örnek kullanımı : Response.Servervariables("REMOTE_ADDR")

Şeklindedir. Bu örneği uyguladığınızda sitenize gelen ziyaretçinin IP (Internet Protokol) numarasını almış olursunuz.

Request.Cookies :

Bu koleksiyon ise Response objesi ile gönderilen cookie'leri alma da kullanılır.

Örnek kullanımı : Request.Cookies("isim")="İsmail AKTAŞ"

Şeklindedir. Bu örneği uyguladığınızda ziyaretçinin sisteminde bulunan içeriği İsmail AKTAŞ olan cookie ' yi okumuş olursunuz.

Sql Komutları ile Veritabanı Sorgulama

Bu bölümde temel Sql komutları ile veritabanını sorgulayarak veri alma, sıralama, çift kayıtları elema, sonuçların formatlarını değiştirme işlemlerinin nasıl yapıldığını inceleyeceğiz. Select ifadesini kullanmadan önce hangi veritabanı ve tablolar ile işlem yapacağımızı bilmeliyiz. Verilerle işlem yapmadan önce bazı komutlarla bu verilerin çekilmesi gerekmektedir. Bu komutlara sql ifadesi diyoruz. Select ifadesi ile seçmek istediğiniz kolon ve satırları belirtebilirsiniz.

Select ifadesinin kullanımı :

Select ifadesi veritabanından bilgi çekmek için kullanılır.

Select ifadesi ile istediğiniz kolan ve satırları belirtebilirsiniz.

- Select ifadesi istenen satırları bir liste şeklinde verir.
- WHERE ifadesini kullanarak sorgunuzu filtreleyebilir ve sadece sizin kriterlerinize uygun satırların sorgu sonucu olarak gelmesini sağlayabilirsiniz. Bu kriterler lojik ifadeler, karşılaştırmalar olabilir.
- FROM ifadesi bu verilerin alınacağı tablo isimlerini belirtmemizi sağlar.

Kolonların belirlenmesi :

Sorgu sanucunda tablodaki tüm kolonlar değilde sadece işinize yarayacak kolonların gelmesini isteyebilirsiniz. Bu durumda istediğiniz kolanların herbirinin adını belirlemek zorundasınız. Select'den sonra kolon isimleri, local değişkenler, hesaplanmış ifadeler belirtebilirsiniz. Yani veritabanından çekilen kolon üzerinde herhangi bir işlem yaptırarak listelettirebilirsiniz. Örneğin veritabanımızda ürün fiyatları olsun. Fakat bizim ürün fiyatları yanında KDV'sinede ihtiyacımız oldu fakat bu bilgi veritabanında yok. Bu

durumda Select ifadesi ile veriyi çekme aşamasında iken burada hesaplanmış ek bir kolon olarak çekebiliriz. Select ifadesinden sonra sadece kolon isimleri bulunmak zorunda değil.

Kolon isimlerini virgül (,) ile ayırmak zorundayız. Veriler listelenirken kolon bilgileri verdiğimiz sırada gelir. Performan kazanmak için kolon kullanımına alışmalıyız. * kullandığımız takdirde tablodaki tüm alanlar ihtiyacımız olsun veya olmasın gelecektir. Bu durum ise bize performans kaybı sağlayacaktır.

Aşağıdaki örnekte employees tablosunda belirttiğimiz kolonları çekiyoruz. Bu tablo Northwind veritabanında bulunmaktadır.

ÖRN: SELECT employeeid, lastname, firstname, title

FROM employees

SELECT	EmployeeID, LastName, FirstName, Title			
FROM	Employees			
				
	EmployeeID	LastName	FirstName	Title
1		Davolio	Nancy	Sales Representative
2		Fuller	Andrew	Vice President, Sales
3		Leverling	Janet	Sales Representative
4		Peacock	Margaret	Sales Representative
5		Buchanan	Steven	Sales Manager
6		Suyama	Michael	Sales Representative
7		King	Robert	Sales Representative
8		Callahan	Laura	Inside Sales Coordinator
9		Dodsworth	Anne	Sales Representative
				

SNÇ:

Where ifadesinin kullanımı :

Where ifadesi ile istediğimiz satırları alabiliriz. Where ifadesinden sonra belirttiğimiz kriterler sınırsız sayıda olabilir.

Where ifadesinden sonra kriter olarak kullandığımız verinin tipi **char, nchar, varchar, nvarchar, text, datetime** ve **smalldatetime** ise tek tırnak (') kullanmalıyız.

Where ifadesi ile normalde fazla sayıda gelen satır sayısını azaltmış oluruz. Where ifademiz tutarsız ise tüm satırlar yine gelecektir.

ÖRN : SELECT employeeid, lastname, firstname, title

FROM employees

WHERE employeeid=5

```

SELECT  EmployeeID, LastName, FirstName, Title
FROM    Employees
WHERE   (EmployeeID = 5)

```

	EmployeeID	LastName	FirstName	Title
▶	5	Buchanan	Steven	Sales Manager
*				

SNÇ :

Sonuç olarak tüm kayıtlarımız arasından sadece employeeid'si 5 olan veriler geldi.

Veri Filtreleme :

Zaman zaman ihtiyacınız olan verileri filtrelemek isteyebilirsiniz. Verilerinizi Where ifadesinden sonra kullanacağınız bazı koşullar ile filtreleyebilirsiniz. Bu koşullar şunlardır;

- * Karşılaştırma operatörleri =, >, <, >=, <=, <>
- * String karşılaştırmalar LIKE ve NOT LIKE
- * Lojik operatörler AND, OR
- * Lojik operatörler (Negatif) NOT
- * Değer aralıkları BETWEEN ve NOT BETWEEN
- * Değer listeleri IN ve NOT IN
- * Bilinmeyen değerler IS NULL ve IS NOT NULL

Karşılaştırma operatörlerinin kullanımı :

Karşılaştırma operatörleri ile tablodaki bir veri ile belirlenmiş bir veri veya hesaplanmış veri karşılaştırılabilir. Fakat karşılaştırmak istediğiniz alan ve verinin tipleri aynı olmalıdır. Sayısal bir ifadeyi tarih ifadesi ile karşılaştıramazsınız. Burada kullanılan operatörler;

= Eşittir

> Büyüktür

< Küçüktür

>= Büyük veya eşittir

<= Küçük veya eşittir

<> Eşit değildir

NOT ifadesinin kullanımı tavsiye edilemez. Çünkü kullanıldığı takdirde tüm tablo değerlendirilir ve veri akışının yavaş olması problemi yaşanır.

ÖRN: SELECT lastname, city

FROM employees

WHERE country='USA'

SELECT	LastName, City
FROM	Employees
WHERE	(Country = 'USA')

LastName	City
Davolio	Seattle
Fuller	Tacoma
Leverling	Kirkland
Peacock	Redmond
Callahan	Seattle
*	

SNÇ :

Country alanı USA olanların listesi alındı. Dikkat edilirse country alanı string bir ifade olduğu için USA ifadesi tek tırnak içine alındı. Bu dikkat edilmesi gereken bir noktadır.

ÖRN : SELECT orderid, customerid

FROM orders

WHERE orderdate < '8/1/96'

SELECT	OrderID, CustomerID
FROM	Orders
WHERE	(OrderDate < '8/1/96')

OrderID	CustomerID
10248	VINET
10249	TOMSP
10250	HANAR
10251	VICTE
10252	SUPRD
10253	HANAR
10254	CHOPS
10255	RICSU
10256	WELLI
10257	HILAA
10258	ERNSH
10259	CENTC
10260	OTTIK
10261	QUEDE
10262	RATTC

SNÇ :

String karşılaştırmaların kullanımı :

Satırlarımızdaki verileri LIKE ifadesi ile bazı wildcard'ları kullanarak filtreleyebiliriz. LIKE ifadesini sadece char, nchar, varchar, nvarchar, binary, varbinary, smalldatetime ve datetime veri tiplerinde kullanabiliriz. Bu wildcardlar aşağıdaki gibidir;

% 0 veya daha fazla karakterli string

_ tek bir karakter

[] belirtilenler arasından tek bir karakter

[^] belirtilenler arasında olmayan tek bir karakter

Bu wildcarların LIKE ile kullanım örnekleri aşağıdaki gibidir;

LIKE 'BR%' BR ile başlayan tüm isimler

LIKE 'Br%' Br ile başlayan tüm isimler. Büyük küçük harf farkı

LIKE '%een' een ile biten tüm isimler

LIKE '%en%' en karakterlerini içeren tüm isimler

LIKE '_en' en ile biten tüm 3 karakterli isimler

LIKE '[CK]%' C veya K ile başlayan tüm isimler

LIKE '[S-V]ing' Tüm 4 karakterli ve ilk karakteri S ile V arasında olan ve ing ile biten tüm isimler

LIKE 'M[^c]%' M ile başlayan ve ikinci karakteri c olmayan tüm isimler

ÖRN : SELECT companyname

FROM customers

WHERE companyname LIKE '%Restaurant%'



The screenshot shows a SQL query window with the following text:

```
SELECT CompanyName
FROM Customers
WHERE (CompanyName LIKE '%Restaurant%')
```

Below the query window, a list of results is displayed:

CompanyName
GROSELLA-Restaurante
Lonesome Pine Restaurant
Tortuga Restaurante
*

Lojik Operatörler

Birden fazla kriterleri bir arada kullanmamıza yardımcı olurlar. AND operatörü kullandığımız takdirde verdiğimiz tüm koşullara uyan kayıtlar gelir. OR operatörü kullandığımız takdirde verdiğimiz koşullardan birine uyanlar gelir. NOT kullandığımızda ise NOT dan sonraki koşulu sağlamayan kayıtlar gelir. Birden fazla koşul kullandığımız durumlarda parantezi koşulları gruplamak için kullanabiliriz. Böylece daha okunaklı olur ve gruplama işlemini yapabiliriz.

SQL Server 2000 ilk önce NOT operatörüne arkasından AND ve OR operatörlerine bakar. Eğer tüm operatörler aynı grup içerisinde ise soldan sağa doğru işlenir.

ÖRN : SELECT productid, productname, supplierid, unitprice

FROM products

WHERE (productname LIKE 'T%' OR productid=46)

AND (unitprice >16.00)

SELECT	ProductID, ProductName, SupplierID, UnitPrice
FROM	Products
WHERE	(ProductName LIKE 'T%' OR ProductID = 46)
	AND (UnitPrice > 16.00)

	ProductID	ProductName	SupplierID	UnitPrice
▶	14	Tofu	6	23,25
	29	Thüringer Rostbrat	12	123,79
	62	Tarte au sucre	29	49,3
*				

SNÇ :

Birden fazla koşulu parantezler ile gruplandırdık.

Değer aralıkları :

BETWEEN ifadesini Where ile beraber kullanarak belli bir değer aralığındaki kayıtları getirebiliriz. Between ifadesi ile belirlenen aralık dışında kalan verileri almak için NOT BETWEEN kullanabiliriz. Between ifadesinde verilen aralığın bir bitiş noktası değildir. Between ile verilen aralık (10 and 20) aslında AND ifadesi kullanımı ve büyük eşittir, küçük eşittirden oluşan bir koşuldur. Bunu kısaca between ifadesi ile yapmaktayız. Örneğin (KOD BETWEEN 10 AND 20) aslında ((KOD>=10) AND (KOD<=20)) demektir.

ÖRN : SELECT productname, unitprice

FROM products

WHERE unitprice BETWEEN 10 AND 20

```

SELECT ProductName, UnitPrice
FROM Products
WHERE (UnitPrice BETWEEN 10 AND 20)

```

	ProductName	UnitPrice
►	Chai	18
	Chang	19
	Aniseed Syrup	10
	Genen Shouyu	15,5
	Pavlova	17,45
	Sir Rodney's Scones	10
	NuNuCa Nuß-Nougat-Creme	14
	Gorgonzola Telino	12,5
	Sasquatch Ale	14
	Steeleye Stout	18
	Inlagd Sill	19
	Chartreuse verte	18
	Boston Crab Meat	18,4
	Singaporean Hokkien Fried Mee	14

SNÇ :

Aslında aynı ifadeyi aşağıdaki şekilde yazabilirdik.

SELECT productname, unitprice

FROM products

WHERE (unitprice>=10)

AND (UnitPrice<=20)

Değer Listeleri :

Verilen bir liste içerisinde yer alan veya almayan değerleri içeren kayıtları getirir. Listede yer alanları aramak için IN, yer almayanları aramak için NOT IN ifadelerini kullanırız. Birden fazla koşulu OR ilede ayırabiliriz veya IN ile bir liste içerisinde arayabiliriz. İkisininide SQL aynı şekilde çözümler. Yani KOD IN [1,2] diyebildiğimiz gibi KOD=1 OR KOD=2 de diyebiliriz. NOT IN ifadesi daha öncede belirtildiği gibi veri akışını yavaşlatabilir.

ÖRN : SELECT companyname, country

FROM suppliers

WHERE country IN ('Japan','Italy')

SELECT	CompanyName, Country
FROM	Suppliers
WHERE	(Country IN ('Japan', 'Italy'))

CompanyName	Country
Tokyo Traders	Japan
Mayumi's	Japan
Formaggi Fortini s.r.l.	Italy
Pasta Buttini s.r.l.	Italy
*	

SNÇ :

Bu ifadeyi şu şekilde de yazabilirdik.

SELECT companyname, country

FROM suppliers

WHERE country='Japan' OR country='Italy'

Bilinmeyen değerler :

Veri girişi sırasında tablodaki kolona veri girilmemiş olabilir ve default veri de atanmamış olabilir. Bu durumda bu kolondaki değer NULL'dur. Null değer 0 (nümerik) veya boşluk (string) ile aynı şey değildir. Bir alana değer girilmemiş kayıtları listelemek için IS NULL ifadesini kullanabiliriz. Null ifadesini diğer koşullar ile kullanamayız. Yani >Null diye birşey diyemeyiz. Kolonlar Null değer içerebilir tanımını tablo yaratırken yapabiliriz.

ÖRN : SELECT companyname, fax

FROM suppliers

WHERE fax IS NULL

SELECT	CompanyName, Fax
FROM	Suppliers
WHERE	(Fax IS NULL)

CompanyName	Fax
Exotic Liquids	<NULL>
New Orleans Cajun Delights	<NULL>
Tokyo Traders	<NULL>
Cooperativa de Quesos 'Las Cabras'	<NULL>
Mayumi's	<NULL>
Specialty Biscuits, Ltd.	<NULL>
Refrescos Americanas LTDA	<NULL>
Heli Süßwaren GmbH & Co. KG	<NULL>
Plutzer Lebensmittelgroßmärkte AG	<NULL>
Norske Meierier	<NULL>
Bigfoot Breweries	<NULL>
Svensk Sjöföda AB	<NULL>

YAYIN SİSTEMİ OTOMASYONU

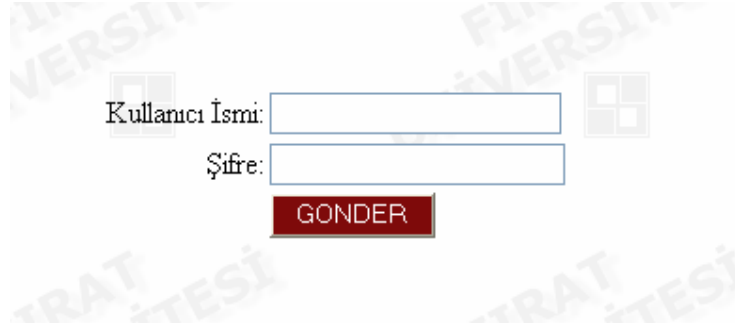
Üniversite bünyesinde görev yapan öğretim elemanlarının gerek yurt içi gerekse yurt dışında sundukları yayınların merkezi bir şekilde dinamik hızlı ve güvenli bir şekilde gerçekleştirilmesine imkan veren programların yapılması amaçlanmaktadır.

Yöneticilerimiz net' e ayrılan her noktadan otomasyona ulaşabilmekte kendilerine tanınan haklar doğrultusunda web' de yayınlanacak olan yayınlarını ekleme güncelleme ve silme işlemlerini gerçekleştirebilmektedir.

Otomasyonda web programlama dili olarak ASP(Active Server Page) ,Database olarak MS SQL Server 2000 kullanılmıştır.otomasyonda yönetici girişleri kullanıcı adı ve şifre ile kontrol edilmektedir.Yönetici bilgileri database server' da bulunup sorgulamalar yapılmaktadır

Yönetici Ara Yüzü :

Yöneticilerimiz net' te açılan her noktadan otomasyona ulaşabilmekte kendilerine açılan hesapla sisteme giriş yapabilmektedir.Kullanıcı bilgilerinin girişi için kullanılan asp sayfası [uye_giris.asp](#) , veritabanı işlemlerinin yapıldığı asp sayfası [uye_giris_islem.asp](#))



Kullanılan veri tabanı tablosu aşağıdaki gibidir:

SQL Server Enterprise Manager - [Data in Table 'sifre' in 'gonca' on '(local)']					
Dosya Pencere Yardım					
SQL					
k_id	kullanici_ad	sifre	adi	soyadi	
1	gonca	nazlı	gonca	öcalan	
*					

Yayın işlemleri modülü

Bu bölümde yayın ekleme silme ve güncelleme işlemleri yapılmaktadır.

YAYINLARIM

Yayın Adı:	Yayın Türü:	Yazarlar	Yayınlandığı yer	Yayın Tarihi:	YAYIN EKLE		
bitirme ödevi	Ulusal Dergi Yayınları	Nazlı Çıtak, Gonca Öcalan	Elazığ	17.4.2001	DOSYA EKLE	GÜNCELLE	SİL
haberleşme	Projeler	Neslihan Şahin	Malatya	16.04.2001	DOSYA EKLE	GÜNCELLE	SİL
GSM Sistemleri	Ulusal Dergi Yayınları	Nazlı Çıtak	Elazığ	24.1.2003	DOSYA EKLE	GÜNCELLE	SİL

Yayın işlemlerinde kullanılan tablo aşağıdadır:

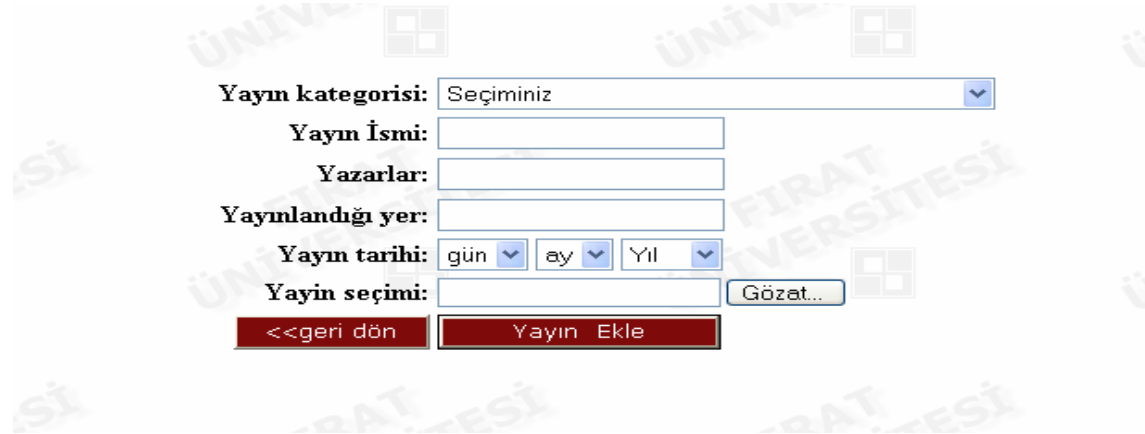
k_id	y_adi	y_turu	y_tarihi	yazarlar	y_yer	dosya	y_id
1	bitirme ödevi	Ulusal Dergi Yayınları	17.4.2001	Nazlı Çıtak, Gonca Öcalan	Elazığ	1_60_ufak asp proi	60
1	haberleşme	Ulusal Seminerler	16.04.2001	Neslihan Şahin	Malatya		61
1	GSM Sistemleri	Ulusal Dergi Yayınları	24.1.2003	Nazlı Çıtak	Elazığ		62

Yayın kategorisine ait tablo aşağıdaki gibidir:

kategori_id	kategori_ismi
0	Projeler
1	Kitaplar
2	Uluslararası Semine
3	Ulusal Dergi Yayınları
4	Ulusal Konferanslar
5	Ulusal Seminerler
6	Uluslararası Hakemli
7	Uluslararası Konfer.
8	Uluslararası Patent
9	Ulusal Patent Ve Öc

Yayın Ekleme Modülü:

Yöneticiler bu bölümde yayın ekleme işlemlerini gerçekleştirebilirler.(Yayına ait veri girişlerinin yapıldığı dosya [yayin_ekle.asp](#) , Yayın ekleme işlemlerini veri tabanına kaydeden asp dosyası [yayin.asp](#))



Yayın Güncelleme Modülü

Yöneticiler web’de sundukları yayınları güncelleme modülü ile düzenleyebilirler.(Veri girişinin yapıldığı asp dosyası [yayin_duzenle.asp](#), Yayın düzenleme işlemlerini veri tabanına kaydeden asp dosyası [yayin_duzenle_islem.asp](#))



Yayın Silme modülü

Yöneticilerin Web’de yayınladıkları yayınlara bu modül ile silme işlemleri yapılır.Yayın silme işlemlerini gerçekleştiren asp dosyası [yayin_duzenle_islem.asp](#))

Dosya Ekle Modülü

Yöneticilerin web’de sundukları yayınlarının dosya güncelleme işlemleri yapılır.(Dosya girişinin yapıldığı asp dosyası [dosya_ekle.asp](#), veri tabanı işlemlerini gerçekleştiren asp programı [dosya_islem.asp](#))



Akademik Modül:

Bu bölümde öğretim üyelerinin yayınlarının sunulduğu bölümdür.Yayına ilişkin dosya bulunuyorsa “tıklayınız...” linkine tıklanarak yayına ulaşılabilir.(Bu sayfayı getiren asp dosyası [akademik.asp](#))

Ulusal Dergi Yayınları

- 1) Nazlı Çıtak, Gonca Öcalan , bitirme ödevi , Elazığ , 17.4.2001 [tıklayınız...](#)
- 2) Nazlı Çıtak , GSM Sistemleri , Elazığ , 24.1.2003

Ulusal Seminerler

- 1) Neslihan Şahin , haberleşme , Malatya , 16.04.2001

[<<geri dön](#)

ARAMA TEKNİKLERİ :

Bu bölümde anlatılacak olan arama teknikleri web sitesine yönelik olup,internetteki arama motorlarıyla ilgili değildir.

Dinamik bir sitede ,sayfaların içeriği veri tabanında oluşturulur.Dinamik olarak hazırlanmış bir sitede bu özellik bize esneklik dinamizm ve arama işlemlerinin detaylı bir şekilde yapılabilmesini sağlar.Oluşturduğumuz veri tabanının alanlarına göre,birçok arama kriteri oluşturabiliriz.Ödevimiz(akademik web) için yapılan arama kriterleri;

- Girilen bir yazıya göre(text bazında) arama ,
- Seçilen kategoriye göre arama,
- Tarihe göre arama

İstenirse bu 3 arama kriteri seçilerek buna göre bir arama istenirse tek birine göre arama ve farklı kombinasyonlarda aramalar yapılabilir.Ve arama sonuçları istenen şartlara göre veritabanında aranıp bulunan sonuçlar görüntülenecektir.

* Arama sayfası([ara.asp](http://naz.ara.asp));

Arama sayfasında yazarak arama kısmında yayın sahibinin adını,soyadının,yardımcı yazarların adını,yayın adını yada yayın tarihinden birini yazarak arama yapılabilir.

Kategori seçiminde ise yayın türleri yer almakta olup bunlara görede arama yapılabilir.

Tarih seçiminde ise yıl olarak seçim yapılarak o yıldaki yayınlar aranabilir.

* Arama sonuçları ([arama_islem.asp](http://naz/arama_islem.asp))(Eğer arama sonucunda aranan özellikte bir yayın bulunamamış ise);



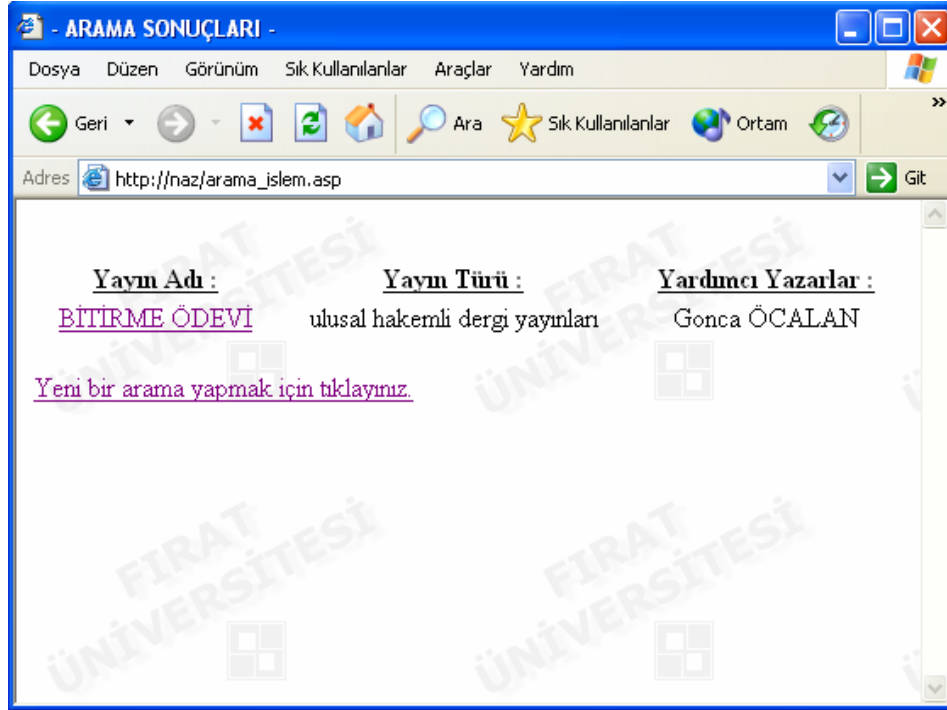
Örneğin;

Yazar adı olarak nazlı,yayın kategorisi olarak ulusal hakemli dergi yayını ve tarih olarak ta 1999 seçilerek yapılan bir arama aşağıdaki gibi görünecektir.

Yapılan aramada ismi yazılarak aranan kişinin hem kendi yayınları hem de yardımcı yazar olarak katıldığı yayınlar görüntülenecektir.Diğer arama kriterleri(kategori ve tarih)aynı türde yayınlardan aranana daha kolay ulaşılması için kullanılabilir.İstenirse bunlar yanlarındaki checkbox'lar işaretlenerek seçilebilir, istenirse seçilmez ve bu kriterler aramaya dahil edilmez.



Yapılan arama sonuçlarının gösterildiği sayfa ;



Bu sayfada yayın adına tıklanarak o yayın görülebilir.

