

08220

# Internet Programcılığı II

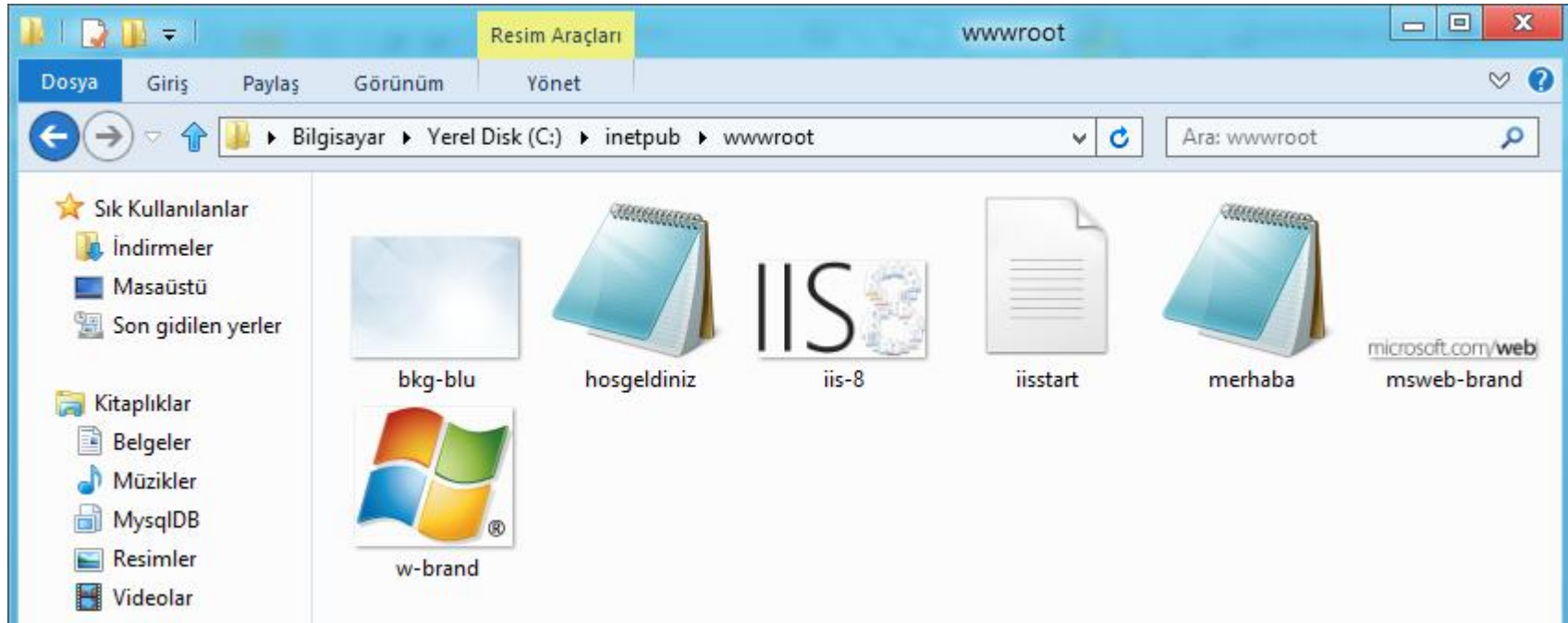
## How to ASP Language

Elbistan Meslek Yüksek Okulu  
2015 – 2016 Güz Yarıyılı



# Internet'te ASP

- ASP teknolojisi ile oluşturduğunuz sayfaları, yani içinde ASP teknolojisinin gerektirdiği kodlar bulunan HTML dosyalarını, .asp uzatmasıyla kaydederiz.



# Internet'te ASP

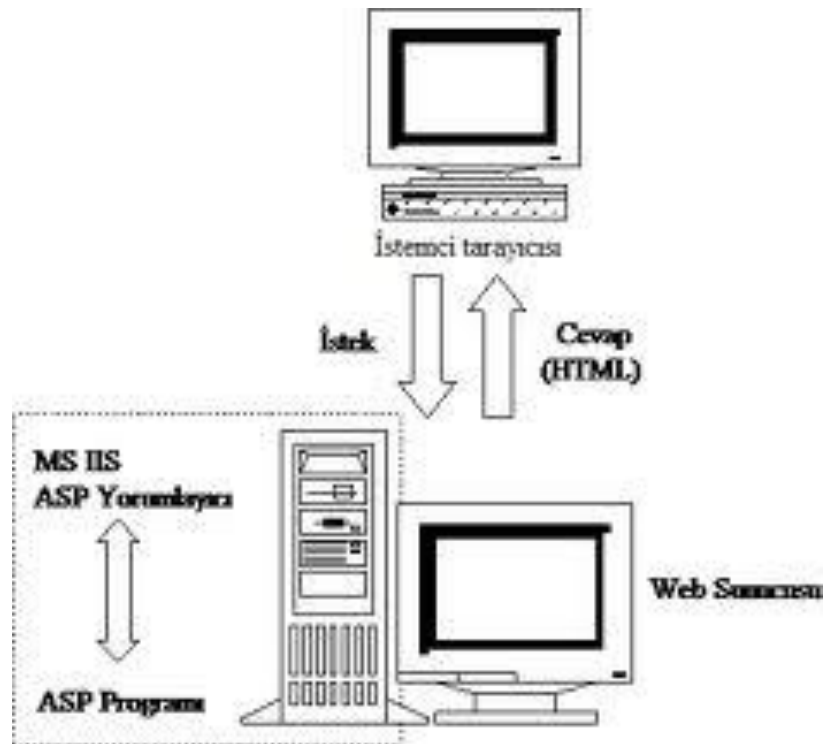
- Bu dosyalar talep edildiğinde, Web Server programı, eğer ASP teknolojisini tanıyorsa, talep edilenin ASP sayfası olduğunu dosya adının uzatmasından anlar; ve bu sayfayı ziyaretçiye yollamadan önce kendisi işleme tabi tutar.

# Internet'te ASP

- ASP sayfamızdaki kodlar bir takım dinamik işler yapacaktır.
- Örneğin hesaplamalar, araştırmalar, veri tabanından veri çekme, dosya sisteminden dosya isteme gibi görevlerini yerine getirecek, ve ortaya çıkacak olan HTML dosyasını ziyaretçiye gönderecektir.

# Internet'te ASP

- ASP internette nasıl çalışır.



# Internet'te ASP

- ASP.DLL, kendisine bir .asp dosyasının geldiğini görünce, hemen ASP'lerin Anayasası olan global.asp'nin çalışıp çalışmadığına bakar.

```
<script language="vbscript" runat="server">
sub Application_OnStart
getcustomers
end sub
sub getcustomers
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
set rs=conn.execute("select name from customers")
Application("customers")=rs.GetRows
rs.Close
conn.Close
end sub,br>
</script>
```

# Internet'te ASP

- global.asa, tıpkı diğer ASP dosyaları gibi bir düz yazı dosyasıdır ve ASP programlarının çalışma koşullarını düzenleyen kuralları içerir.
- Global.asa dosyasına yeri geldiğinde tekrar bakacağız.

# Internet'te ASP

- ASP dosyası hem HTML kodları içerir, hem de içinde bir Script diliyle yazılmış kodlar vardır.
- ASP'ye “program” özelliği kazandıran bu Script dili ile yazılmış kodlardır.



# Internet'te ASP

- ASP.DLL, önce gelen .asp dosyasında hangi Script dilinin kullanıldığına bakar ve bunun için gerekli ortamı oluşturur; yani bu Script dilini yorumlayacak programı çalıştırır.
- Bu program Script'i yorumlar ve icra edilecek komutları icra eder.

# Internet'te ASP

- ASP.DLL, icra edilen komutlar, işletim sisteminin yardımını istiyorsa;
  - veritabanından veri çekmek
  - dosya sistemine bir dosya açtırmak, yazdırmak, sildirmek gibi

bu yardımın edinilmesini de sağlar.

# Internet'te ASP

- Bütün bu işlerin sonunda sizin yazdığınız HTML kodlarına ek yapmak;
  - Bir tablonun içini, çekilen verilerle doldurmak
  - Dosya sisteminden edinilen bir dosyanın içeriğini sayfaya aktarmak gibi

gerekiyorsa bu ekleri ASP.DLL yapar.

# Internet'te ASP

- ASP.DLL, kendisine sevk edilen dosyayı satır satır okur ve önce dosyadaki ASP kodlarının gerektirdiği HTML değerlerini bulur; yani önce ASP icra edilir, gereği yerine getirilir.
- Sonra HTML bölümleri ile birleştirilip sonuçta sunucuya saf ve temiz bir HTML sayfası gönderilir.

# Internet'te ASP

- Gönderilen bu sayfanın içinde bir satır bile ASP kodu bulunmaz.
- Eğer sayfanıza ziyaretçinin Browser'ında çalışması amacıyla Javascript veya VBScript kodları koydu iseniz, elbette bu kodlar HTML'in parçası olarak ziyaretçiye gidecektir.
- Giden sayfada artık ASP'ye ilişkin hiç bir şey kalmamış olacaktır.

# Internet'te ASP

- ASP tekniğinin nasıl çalıştığını anladığımıza göre gerçekten ASP ile mesela Javascript arasındaki farkı anlamamız gerekir.
- ASP bir Server-Side Script teknolojisidir.
- Internet'te istemci ile sunucu arasındaki çizginin sunucu tarafına Server-Side (Sunucu Tarafı), istemci tarafına da Client-Side İstemci Tarafı) denir.

# Internet'te ASP

- Server-Side'da çalışan Script programları da “Server-Side Script” (Sunucu Tarafı Script'i) adını alır.
- Dolayısıyla şöyle bir sayfa yaparsanız (merhaba.htm), bu sayfada Client-Side Script teknolojisi kullanmış olursunuz:

# Server side script – Client side script

- İki örnekle konuyu pekiştirelim.

```
<HTML><HEAD>
<TITLE>JavaScript ile Tarih</TITLE>
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-9">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254">
</HEAD>
<BODY BGCOLOR=WHITE>
<H1>Merhaba Dünya</H1>
<H2>Bugün:</H2>
<H3><SCRIPT LANGUAGE=JAVASCRIPT>
<!--
tarih = new Date();
document.write(tarih);
//--></SCRIPT>
</H3>
</BODY>
</HTML>
```



# Internet'te ASP

- HTML sayfanıza `<SCRIPT>..</SCRIPT>` etiketleri arasına yerleştireceğiniz bu kodun çalışması için Server'ın hiçbir şey yapması gerekmiyor.
- Kodu Browser çalıştıracak ve günün tarihini size (Türkçeleştirmedeğiniz için İngilizce olarak) bildirecektir.

# Server side script – Client side script

- İki örnekle konuyu pekiştirelim.

```
<HTML>
<HEAD>
<TITLE>VBScript ile Tarih</TITLE>
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-9">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254">
</HEAD>

<BODY BGCOLOR=WHITE>
<H1>Merhaba Dünya</H1>
<H2>Bugün:</H2>
<H3>
<SCRIPT LANGUAGE=VBScript RUNAT=SERVER>
Response.write(Date)
</SCRIPT>.
</H3>
</BODY>
</HTML>
```

# Internet'te ASP

- Bu kodu Server çalıştıracak. Bunu <SCRIPT> etiketinin içindeki “RUNAT” özelliği ile belirtiyoruz.
- (“Run,” çalıştır; “at” ise “içinde, üzerinde” anlamına gelir..)
- “RUNAT” “sunucu ...de çalıştır” gibi bir anlama sahiptir.

# Internet'te ASP

- Burada RUNAT'ın karşısına yazdığımız SERVER ifadesi ile, Script'in Browser'a gönderilmeden önce Server'da çalıştırılmasını sağlıyoruz.
- Server'a “Bu Script'i Server'da Çalıştır!” demiş oluyoruz.

# VBScript'te İşlemciler (Operatörler)

- Bilgisayar Operatörü ile program Operatörü arasında fazla bir fark yoktur.
- Her ikisi de verdiğiniz değerleri ya karşılaştırır bir sonuç bulurlar; ya da bu değerlerle aritmetik işler yapar ve bir sonuç ortaya çıkarırlar.

# VBScript'te İşlemciler (Operatörler)

- Bilgisayar Operatörü daha sonra bu iş için sizden para ister; program operatörü istemez!
- VBScript'in operatörleri ve yaptıkları işler şöyle sıralanır:

# VBScript'te İşlemciler (Operatörler)

<u>Operatör</u>	<u>İşlev</u>	<u>Sınıfı</u>
+	Toplama	Aritmetik
-	Çıkartma	
*	Çarpma	
/	Bölme	
^	Üssünü alma	
\	Tamsayı bölme	

# VBScript'te İşlemciler (Operatörler)

<u>Operatör</u>	<u>İşlev</u>	<u>Sınıfı</u>
<u>Mod</u>	Modüler aritmetik	
=	Bir değişkenin değerine eşit olduğunu sınar Karşılaştırma	
<>	Bir değişkenin değerine eşit olmadığını sınar	
>and<	Bir değişkenin değerinden büyük veya küçük olduğunu sınar.	
>= and <=	Bir değişkenin değerinden büyük veya eşit, veya küçük veya eşit olduğunu sınar.	



# VBScript'te İşlemciler (Operatörler)

<u>Operatör</u>	<u>İşlev</u>	<u>Sınıfı</u>
<u>And</u>	Bir veya daha fazla değişkeni test olarak karşılaştırır Mantıksal	
<u>Or</u>	Bir işlemin devamı için hangi koşulun oluşması gerektiğini sınar	
<u>Not</u>	Bir ifadeyi negatif hale getirir	
<u>XoR</u>	Sadece bir koşulun doğru olup olmadığını sınar	
<u>Eqv</u>	İki değişkenin eşitliğini sınar	
<u>Imp</u>	İki ifadede mantıksal implikasyon işlemi yapar.	

# VBScript'te İşlemciler (Operatörler)

<u>Operatör</u>	<u>İşlev</u>	<u>Sınıfı</u>
<u>And</u>	Bir veya daha fazla değişkeni test olarak karşılaştırır Mantıksal	
<u>Or</u>	Bir işlemin devamı için hangi koşulun oluşması gerektiğini sınar	
<u>Not</u>	Bir ifadeyi negatif hale getirir	
<u>XoR</u>	Sadece bir koşulun doğru olup olmadığını sınar	
<u>Eqv</u>	İki değişkenin eşitliğini sınar	
<u>Imp</u>	İki ifadede mantıksal implikasyon işlemi yapar.	

Vbasic scriptte işlem yaparken parantez kullanırız.

## VBScript'de Program Kontrolü

- İster Script diliyle, ister gerçek programlama diliyle yazılsın, bir bilgisayar programının varlık sebebi, çeşitli durumları değerlendirerek, belirli durumlarda belirli kararlar verebilmesidir.
- Bunu programın kontrol öğelerini kullanarak yaparız.

## VBScript'de Program Kontrolü

- Programlar, bu öğeler sayesinde karşılaştırma yaparlar; belirli durumların oluşup oluşmadığını sınırlarlar; veya belirli bir durumun oluşmasına veya sona ermesine bağlı olarak bir iş yaparlar veya yapmazlar.
- Bunu sınımlamalarla (koşullu ifadelerle) veya döngülerle sağlarız. Kimi zaman da, programa (programın mantığı çerçevesinde) istediğimiz anda yapmakta olduğu işi durdurarak, başka bir işi yapmasını bildirebiliriz.
- Bunlara da Süreçler (veya Prosedürler) denir.

# VBScript’de Program Kontrolü

## Mantıksal Sınamalar

- VBScript’te programın karar vermesini sağlayan ilk kontrol ögesini “eğer ... ise... yap!” şeklinde özetleyebiliriz.
- VBScript bu işlemi iki ayrı ifadeyle yaparız:

### If.. Else

- VBScript’in vereceğiniz bir durumun bulunup bulunmadığını sınamasını sağlar.

# VBScript'de Program Kontrolü

Genel yazım kuralı şöyledir:

```
If şart Then
    [şart doğru ise yapılacak işler]
Else
    [şart doğru değilse yapılacak işler]
End If
```

Bir uygulama yapalım. Ama önceki diğer iki uygulamayı da yapıp kendi clientımızda bir klasöre depolayalım.

# VBScript'de Program Kontrolü

```
<HTML><HEAD>
<TITLE>ASP İLE SAATE GORE SELAM</TITLE>
<META http-equiv="content-type" content="text/html; charset=ISO-8859-9">
<META http-equiv="Content-Type" content="text/html; charset=windows-1254">
</HEAD><BODY>
<H2><CENTER>
<%
If Hour (Now) <12 Then
    Response.Write "Günaydın! "
ElseIf Hour (Now) >= 18 Then
    Response.Write "İyi akşamlar! "
Else
    Response.Write "Tünaydın! "
End If
Response.Write "<BR>"
Response.Write "Site Onarım Sitesine Hoşgeldiniz"
%></CENTER>
</H2></BODY>
</HTML>
```

# Internet Programcılığı II

## Array Fonksiyonu

- VBScript'in kullanılmaya hazır bir çok fonksiyonu vardır; bunlardan biri olan Array ile dizi değişken oluşturabiliriz.
- Gunler(7) dizi-değişkenini gün adlarına yazarak oluşturmak istiyoruz.

```
<%  
Dim Gunler = Array ("Pazartesi", "Salı", "Çarşamba", "Perşembe", "Cuma",  
"Cumartesi", "Pazar")  
%>
```

- Bu şekilde hem dizi-değişkeni oluşturabiliriz; hem de değerlerini atayabiliriz.



# Internet Programcılığı II

## Array Fonksiyonu

- Bu suretle oluşturduğumuz dizi değişkenin üyelerine daha sonra sıra numaraları görüntülemek istediğimizde;

```
<%=Gunler(6)%>
```

- Tarayıcı ekranımızda “Pazar” yazacaktır. (Günler 1’den değil 0’dan başlar.)

# Internet Programcılığı II

## Neden Dizi (Array) Değişken

- Dizi değişken oluşturacağımıza, neden “ogrenci1,” “ogrenci2,” “ogrenci3” diye değişkenler oluşturmuyoruz ve bunlara değerler atamıyoruz?

Kötü bir soru!

- Dizi değişkenlerde bir değişkene ancak numarasıyla atıfta bulunabilirsiniz. Veya numara yerine bir başka değişkeni kullanabilirsiniz.

# Internet Programcılığı II

## Neden Dizi (Array) Değişken

```
<%  
....  
If Ogrenciler(OgrenciNo) = "Selami"  
...  
%>
```

- Böyle bir döngü ile yaptığımız “Ogrenciler” dizisinde “ogrencino” dizisinden değer almaktır.
- Aradığınız öğrencinin Selami olup olmadığını sınavabiliriz.

# Internet Programcılığı II

## Numeric – Alfanumeric

- VBScript'te değişkenin metin mi, sayı mı olduğunu ayırt etmiyor muyuz?
- Başka programlama dillerinde bir değişkenin değeri harf ve rakamlardan oluşuyorsa, yani matematik işlem yapmaya elverişli değilse bunlara String (Alfanümerik, karakter değerler) denir.

# Internet Programcılığı II

## Numeric – Alfanumeric

- Programlama dillerinde bir de matematik işlem yapmaya elverişli değişken türü vardır: Sayı (Number).
- VBScript, bir değişkene alfanümerik (karakter, metin) olarak atadığınız değeri çift tırnak içine almanızı ister.

# Internet Programcılığı II

## Numeric – Alfanumeric

- Örneğin: Ogrenci(12) değişkeni için Necip değerini atamak istiyorsunuz: Necip kelimesini çift tırnak içine almak zorundasınız.
- Sayı olarak kullanacağınız değerleri ise tırnak içine almazsınız. (Sayı olarak kullanacağınız değerlerin sadece rakam olabileceğini söylemek gerekmez!)
- Fakat, işleri karıştıran nokta, VBScript açısından değişken türü diye bir şey olmamasıdır.

# Internet Programcılığı II

## Numeric – Alfanumeric

- Visual Basic, küçültülüp Script dili haline getirilirken, değişken türü özelliği alınmamıştır. Bu durumda ortaya işleri karıştıran (veya kolaylaştıran) böyle bir durum çıkmış bulunuyor.
- Karışıklık, çift tırnak içinde verdiğiniz bir değeri matematik işlemde kullanmaya kalktığınızda karşınıza çıkabilir.
- Rakam olmayan bir karakter-dizisini bir değişkene tırnaksız olarak atadığınızda VBScript “tanımsız değişken” vermeye kalktığınızı söyleyerek, duracaktır.

# Internet Programcılığı II

## Numeric – Alfanumeric

- VBScript'in bu eksikliğini gidermek için bazı ASP programcıları değişken adlarının önüne karakter-dizileri için “str” harflerini koyarlar: strAy, strOgrenciler, gibi.
- VBScript'in karakter-dizisi ve sayı değişkenleri nasıl ayırt ettiğini veya etmediğini bir kaç örnekle görelim.
- Şu kodu yazıp farklı varyasyonlarını çalıştıralım.



# Internet Programcılığı II

```
<% Option Explicit %>  
<HTML>  
<%  
Dim Degisken(2), Toplam  
Degisken(1) = "Selami"  
Degisken(2) = "KELLE"  
Toplam = degisken(1) + Degisken(2)  
%>  
<% =Toplam %>  
</HTML>
```

Tarayıcıdaki çıktımız : “SelamiKELLE” değil mi?

# Internet Programcılığı II

## Numeric – Alfanumeric

- Şimdi Degisken(1) değeri olarak tırnak içine alarak 5, Degisken(2) değeri olarak yine tırnak içinde 10 yazın.
- Programı tekrar çalıştırın. Sonuç? 510!?
- Tamam şimdi tırnakları kaldırın 5 ve 10 rakamlarını tırnaksız yazın. Sonuç? 15.

# Internet Programcılığı II

## Numeric – Alfnumeric

- VBScript ayırım yapmıyor bile olsa, karakter-dizisi (String) iken toplama işleminde ard arda eklenmesinden (concatenate) kaynaklanıyor.
- Tırnakları kaldırdığınızda, VBScript bu değerlerin sayı olduğunu ve doğru işlemi, yani toplama işlemini yapıyor.
- Şimdi VBScript'in değişkenleri ele alış tarzını daha iyi kavramak için Degisken(1) değeri olarak tırnak işareti koymadan kendi adınızı, Degisken(2) değeri olarak da soyadınızı yazın ve programı çalıştırın. Ne sonuç aldınız?

# Internet Programcılığı II

## Numeric – Alfanumeric

- VBScript'in kullandığı tek tür değişkene variant denir.
- Variant, karakter-dizini (String) de olabilir, sayı (Number) da.
- Fakat bir variant'ın içindeki değer, veri olarak nitelendiğinde şu veri türlerinden birisidir.

# Internet Programcılığı II

## Veri olarak değişken türleri

- Boolean Değişkenin değeri ya True (doğru) ya da False (yanlış) olur; True'nun değeri  $-1$ , False'un değeri ise  $0$ 'dır.
- Byte  $0$  ile  $255$  arasında bir sayısal değer olabilir.
- Double Yüzen noktalı değer denilen veri türüdür. Pozitif sayılar için  $4.9E^{-324}$  ile  $1.8E^{308}$  arasında, negatif sayılarda  $-4.9E^{-324}$  ile  $-1.8E^{308}$  arasında bir değer alabilir.

# Internet Programcılığı II

## Veri olarak değişken türleri

- Date/Time Önceden belirlenmiş biçimlerde tarih ve zaman bilgisi içerir.
- Empty Tanımlanmış ancak henüz değer atanmamış (boş) değişken türüdür.
- Error Programın hata mesajlarını ve hata değerlerini tutar.
- Integer Noktalık bölüm içermeyen tamsayı verilerdir; değeri -32.768 ile +32.767 arasında olabilir.

# Internet Programcılığı II

## Veri olarak değişken türleri

- Long Noktalık bölüm içermeyen tamsayı verilerdir; değeri -2.147.483.648 ile 2.147.483.648 arasında olabilir.
- Null İçinde veri bulunmamak üzere tanımlanmış değişkenlerdir.
- Object Windows OLE Nesnelerini tutmak üzere tanımlanmış değişkenlerdir.
- Single Yüzen noktalı değer denilen veri türüdür. Pozitif sayılar için  $1.4E^{-45}$  ile  $3.4E^{38}$  arasında, negatif sayılarda  $-1.4E^{-45}$  ile  $-3.4E^{38}$  arasında bir değer alabilir.
- String Alfanumerik veri tutan değişkenlerin değeridir

# Internet Programcılığı II

## Sabit Değerler

- VBScript'te sabit değer atabilirmiyiz. Elbette
- Sabit değer, bütün ASP sayfası boyunca (hatta isterseniz, bütün site, yani Uygulama boyunca) değişmeden kalır.
- Bu değerleri Const (constant, sabit kelimesinden türetilme) komutuyla belirtiriz



# Internet Programcılığı II

## Sabit Değerler

```
Const DolarDeger = 1760  
Const SirketinAdi = "Web Tasarım ve Site Onarım A.Ş."  
Const Slogan = "Siteleriniz itina ile onarılır"
```

- Her yazdırma işlemi yaptırdığımızda const ile atanmış değeri tarayıcımıza yazdırmış oluruz.

```
<%Response.Write(Slogan)%>
```

08220

# Internet Programcılığı II

Devamı haftaya...

Not Haftaya çok çalışacağız

