Web Teknolojileri

Hafta11

İçerik

- Javascript 2.Kısım

JavaScript Objeler

- Objeler(Nesneler) gerçek hayattaki varlıkları modelleyen değişkenlerdir. Örnek olarak bir arabayı obje olarak modellersek; Bir arabanın ağırlık, renk gibi özellikleri varken çalıştır ve stop et şeklinde metotları bulunmaktadır.

JavaScript Objeler

- Obje = araba

Özellik	Metod
araba.marka=Fiat	araba.calistir()
araba.model=500	araba.sur()
araba.agirlik=850kg	araba.rolanti()
araba.renk=beyaz	araba.durdur()

- Tanımlanan bir obje ile birden fazla araba oluşturabilir. Oluşturulan her araba farklı özelliklere sahip olabilir. Örneğin bir arabanın rengi siyahken diğer beyaz olabilir.

JavaScript Objeler

Aşağıdaki kodda car değişkenine basit bir Fiat string değeri aktarıldı.

```
var araba = "Fiat";
```

İkinci kodda car değişkenine birden fazla değer aktarılmaktadır.

```
var araba = {marka:"Fiat", model:"500", renk:"beyaz"};
```

isim:değer şeklinde değer aktarılır. Virgüllerle birden fazla değer birbirinden ayrılır.

```
var kisi = {ad:"Ayşe", soyad:"Yılmaz", yas:50, gozRengi:"mavi"};
```

Nesnelerde Özelliklere Erişim

Obje özelliklerine iki yolla erişilebilir.

nesneAdı.özellikAdı

İkinci yol

nesneAdı["özellikAdı"]

```
var kisi = {ad:"Ayşe", soyad:"Yılmaz", yas:50, gozRengi:"mavi"};
```

Örnek

1.Yöntem

```
kisi.soyad;
```

2.Yöntem

```
kisi["soyad"];
```

Nesnelerde Metotlara Erişim

Nesne metotlarına erişim

nesneAdı.metotAdı()

```
var kisi = {
    ad: "John",
    soyad : "Doe",
    id       : 5566,
    adSoyad : function() {
       return this.ad + " " + this.soyad;
    }
};
```

Örnek

Doğru Erişim

```
isim = kisi.adSoyad();
```

Hatalı Erişim

```
isim = kisi.adSoyad;
```

Nesne Özellik Örneği

```
<!DOCTYPE html>
<html>
<body>

<p id="demo">Sonucu Burada Göster.</p>

<script>
   var kisi = {ad:"Ayşe", soyad:"Yılmaz"};
   document.getElementById("demo").innerHTML = kisi.ad;
</script>

</body>
</html>
```

Nesne Metot Örneği

```
<!DOCTYPE html>
<html>
<body>
<p id="demo"> Sonucu Burada Göster.</p>
<script>
var kisi = {
    ad: "Ayşe",
    soyAd : "Yılmaz",
    id       : 5566,
    adSoyad : function() {
       return this.ad+ " " + this.soyAd;
    }
};
document.getElementById("demo").innerHTML = kisi.adSoyad();
</script>

</body>
</html>
```

Değişkenlerin Yaşam Alanları

Değişkenler yaşam alanlarına göre iki türdedir.

1-Global değişkenler

2-Lokal değişkenler

Global değişkenlere javascript kodunda her yerden ulaşılabilir ve yaşam alanları tüm javascript kodu kadardır.

Lokal değişkenler ise sadece tanımlandığı bloktan erişilebilir ve yaşam alanları blok içindedir. Bloğun dışına çıkılınca ölürler.

Global Değişkenler

Global Değişken örneği

```
var araba = " Volvo";

// araba değişkenine buradan erişilebilir.

function myFunction() {

    // araba değişkenine buradan da erişilebilir

}
```

Global Değişken Örneği

```
<!DOCTYPE html>
<html>
<body>
<p>Bir GLOBAL herhangi script veya fonksiyonda
kullanılabilir.</p>
<p id="demo"></p>
<script>
var araba = "Volvo";
myFunction();

function myFunction() {
    document.getElementById("demo").innerHTML =
    "Bu araba " + carName;
}
</script>
</body>
</html>
```

Lokal Değişkenler

Lokal Değişken örneği

```
// araba değişkenine buradan ulaşılamaz.
// Burada yaşamamaktadır.
function myFunction() {
    var araba = "Volvo";

    // araba değişkenine buradan erişilebilir.
    // Değişken bu blok için yaşar.

}
```

Lokal Değişken Örneği

```
<!DOCTYPE html>
<html>
<body>
<p>Bir LOCAL değişkene sadece tanımlandığı fonksiyon
içerisinden erişilebilmektedir. </p>
<p id="demo"></p>
<script>
myFunction();
document.getElementById("demo").innerHTML =
"Bu araba " + typeof araba;

function myFunction() {
    var araba = "Volvo";
}
</script>

</body>
</html>
```

String Özellikleri

Stringler metinsel bilgileri saklayan değişken tipleridir.

İki şekilde tanımlanabilir; tek tırnak veya çift tırnak içinde.

```
var araba = "Volvo XC60";
var  araba = 'Volvo XC60';
```

String uzunluğu length özelliği ile bulunur.

```
var metin = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
var sln = metin.length;
```

String özel karekterleri escape karetkeri ile yazdırabilirsiniz.

```
var x = 'It\'s alright';
var y = "\"Sakarya Üniversitesi\"ne Hoşgeldiniz."
```

String Özel Karekterler

Code	Outputs
\'	Tek tırnak
\"	Çift tırnak
\\	backslash
\n	Yeni satır
\r	Satır başı
\t	tab
\b	backspace
\f	Sayfa başı

String Metodlar

Method	Tanımlama
charAt()	İndeksi verilen karakteri döndürür(pozisyon)
indexOf()	Stringte verilen değerin ilk bulunduğu indisi döndürür
concat()	İki veya daha fazla stringi birleştirir ve birleşmiş stringi döndürür
lastIndexOf()	Stringte verilen değerin son bulunduğu indisi döndürür
replace()	Stringte yer bir metni bulup değiştirmek için kullanılır
search()	String içinde bir metnin pozisyonunu döndürür
slice()	Metinde Başlangıç ve bitiş değeri verilen aralıktaki metni döndürür
split()	verilen karaktere göre metni bölerek diziye dönüştürür
substr()	Metinde Başlangıç ve uzunluğu verilen aralıktaki metni döndürür
substring()	Metinde Başlangıç ve bitiş değeri verilen aralıktaki metni döndürür

String Metodlar

toLocaleLowerCase()	Sunucunun bölge ayarlarına referans alarak metni küçük harfe çevirir
toLocaleUpperCase()	Sunucunun bölge ayarlarına referans alarak metni büyük harfe çevirir
toLowerCase()	Metni küçük harfe çevirir
toString()	Nesneyi string ifadeye dönüştürür
toUpperCase()	Metni büyük harfe çevirir
trim()	Metnin başındaki ve sonundaki boşlukları siler
valueOf()	Nesnenin string değerini döndürür

String Örnekler (indexOf)

```
<!DOCTYPE html>
<html>
<body>
<p id="p1">Bu köşe kış köşesi, bu köşe yaz köşesi…</p>
<button onclick="myFunction()">DENE</button>
<p id="demo"></p>
<script>
   function myFunction() {
      var str = document.getElementById("p1").innerHTML;
      var pos = str.indexOf("köşe");
    document.getElementById("demo").innerHTML = pos;
}
</script>

</body>
</html>
```

String Örnekler(substring)

```
<!DOCTYPE html>
<html>
<body>

<p>substr() metodu, bir karakter katarının bir parçasını çıkartır ve
çıkartılan parçayı yeni bir karakter katarında döndürür:</p>

<p id="demo"></p>

<script>
var str = "Elma, Muz, Kivi";
document.getElementById("demo").innerHTML = str.substring(6,10);
</script>

</body>
</html>
```

Number Metodları

parseInt tamsayı tipine dönüştürür.

```
parseInt("10");          // 10 döndürür
parseInt("10.33");       // 10 döndürür
parseInt("10 20 30");    // 10 döndürür
parseInt("10 yıl");      // 10 döndürür
parseInt("yıl 10");      // NaN döndürür
```

parseFloat() ondalık sayı tipine dönüştürür.

```
parseFloat("10");          // 10 döndürür
parseFloat("10.33");       // 10.33 döndürür
parseFloat("10 20 30");    // 10 döndürür
parseFloat("10 yıl");      // 10 döndürür
parseFloat("yıl 10");      // NaN döndürür
```

Number Metodları

valueOf() sayısal değerini geri gönderir.

```
var x = 123;
x.valueOf();            // returns 123 from variable x
(123).valueOf();        // returns 123 from literal 123
(100 + 23).valueOf();   // returns 123 from expression 100 + 23
```

Tarih Saat Metodları

```
new Date()
new Date(milliseconds)  //86400000
new Date(dateString)  //"October 13, 2014 11:13:00"
new Date(year, month, day, hours, minutes, seconds, milliseconds)
//99, 5, 24, 11, 33, 30, 0
```

```
<script>
  var d = new Date();
  document.getElementById("demo").innerHTML  = d;
</script>
```

Tue Mar 14 2017 14:23:44 GMT+0300

Dizileri Kullanma

Söz Dizimi (Syntax):

```
var dizi-adi = [item1, item2, ...];
```

Örnek:

```
var arabalar = ["Ford", "Volvo", "BMW"];
```

```
<p id="demo"></p>

<script>
  var arabalar = ["Ford", "Volvo", "BMW"];
  document.getElementById("demo").innerHTML  = arabalar[0];
</script>
```

Dizilerde Farklı Tipleri Barındırma

Dizilerde farklı tipler tek bir dizide barındırılabilir.

Aşağıdaki örnekte string ve integer tipdeki değerler aynı dizide barınmaktadır.

```
var kisi = ["Ayşe", "Yılmaz", 46];
```

length() özelliği ile dizinin uzunluğu bulunabilir.

```
var meyveler = ["Muz", "Portakal", "Elma", "Mango"];
meyveler.length;    // meyveler dizisinin uzunluğu:4
```

Dizi Elemanlarında Dolaşma

Dizi elemanlarında döngü yardımıyla dolaşılabilir.

```
var index;
var meyveler = ["Muz", "Portakal", "Elma", "Mango"];
for (index = 0; index < meyveler.length; index++) {
    metin += meyveler[index];
}
```

Diziye Eleman Ekleme

Diziye eleman iki yolla eklenebilir.

Son eleman olarak ekleme

```
var meyveler = ["Muz", "Portakal", "Elma", "Mango"];
meyveler[meyveler.length] = "Limon"; //meyveler dizisine (Limon)
                                     //ekleme
                          meyveler
```

Push metoduyla ekleme

```
var meyveler = ["Muz", "Portakal", "Elma", "Mango"];
fruits.push("Limon");   //meyveler dizisine (Limon) ekleme
```

Boolean Değerler

İki değere sahip olan programlamada sıklıkla kullanılan değişken tipidir.

YES / NO
ON / OFF
TRUE / FALSE

Boolean() fonksiyonu karşılaştırmanın sonucunu verir.

```
Boolean(10 > 9)          // true döndürür
```

0 değeri false olarak değerlendirilir.

```
var x = 0;
Boolean(x);        // false döndürür
```

Karşılaştırma Operatörleri

Operator	Açıklama	Karşılaştırma	Dönen Değer
==	eşit	x == 8	false
		x == 5	true
		x == "5"	true
===	Değer ve tipi eşit	x === 5	true
		x === "5"	false
!=	Eşit değil	x != 8	true
!==	değeri veya tipi eşit değil	x !== 5	false
		x !== "5"	true
		x !== 8	true
>	büyük	x > 8	false
<	küçük	x < 8	true
>=	Büyük veya eşit	x >= 8	false
<=	Küçük veya eşit	x <= 8	true

Lojic Operatörler

Operator	Açıklama	Example
&&	and	(x < 10 && y > 1) is true
||	or	(x == 5 || y == 5) is false
!	not	!(x == y) is true

Karşılaştırma

Syntax

```
if (koşul) {
    koşul doğruysa yapılacaklar
}
```

```
if (saat < 18) {
    mesaj = "İyi Günler…";
}
```

Karşılaştırma

Syntax

```
if (koşul) {
    koşul doğruysa yapılacaklar
} else {
    koşul yanlışsa yapılacaklar
}
```

```
if (saat < 18) {
    mesaj = "İyi Günler…";
} else {
    mesaj = "İyi Akşamlar…";
}
```

Karşılaştırma

Syntax

```
if (koşul1) {
    koşul1 doğruysa yapılacaklar
} else if (koşul2) {
    koşul1 yanlış ve koşul2 doğruysa yapılacaklar
} else {
    koşul1 ve koşul2 yanlışsa yapılacaklar
}
```

```
if (saat < 10) {
    mesaj = "Günaydın…";
} else if (saat < 20) {
    mesaj = "İyi Günler…";
} else {
    mesaj = "İyi Akşamlar…";
}
```

Karşılaştırma Örneği

```
<!DOCTYPE html>
<html>
<body>
<p id="demo">Sonucu Burada Göster.</p>
<script>
   var mesaj;
   var saat = new Date().getHours();
   if (saat < 18) {
     mesaj = "İyi Günler…";
   }
   else {
      mesaj = "İyi Akşamlar…";
   }
   document.getElementById("demo").innerHTML = mesaj;
</script>

</body>
</html>
```

Switch

```
switch(ifade) {
    case n:
        kod bloğu
        break;
    case n:
        kod bloğu
        break;
    default:
        varsayılan kod bloğu
}
```

Switch

```
switch (new Date().getDay()) {
    case 0: gun = "Pazar";          break;
    case 1: gun = "Pazartesi";      break;
    case 2: gun = "Salı";           break;
    case 3: gun = "Çarşamba";       break;
    case 4: gun = "Perşembe";       break;
    case 5: gun = "Cuma";           break;
    case 6: gun = "Cumartesi";      break;
    default: alert("!!!");          break;
}
```

Döngüler for

```
for (ifade 1; ifade 2; ifade 3) {
    gerçekleştirilecek kod bloğu
}
```

```
for (i = 0; i < 5; i++) {
    text += "Sayı " + i + "<br>";
}
```

```
for (i = 0, len = arabalar.length, metin = ""; i < len; i++)
{
    metin += arabalar[i] + "<br>";
}
```

Döngüler for

```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>
<script>
   var metin = "";
   var i;
   for (i = 1; i < 10; i = i + 2) {
       metin += i + "<br>";
   }
   document.getElementById("demo").innerHTML = metin;
</script>

</body>
</html>
```

1
3
5
7
9

Döngüler while

```
while (koşul) {
    gerçekleştirilecek kod bloğu
}
```

```
while (i < 10) {
    text += "The number is " + i;
    i++;
}
```

Döngüler do while

```
do {
    gerçekleştirilecek kod bloğu
}
while (koşul);
```

```
do {
    text += "Sayı " + i;
    i++;
}
while (i < 10);
```

Break

Break komutu döngüyü kullanıldığı yerde kırarak sonlandırır.

```
for (i = 0; i < 10; i++) {
    if (i === 3) { break; }
    text += "Sayı:  " + i + "<br>";
}
```

Continue

Continue komutu döngüyü kullanıldığı yerde işlem yaptırmadan bir sonraki iterasyona yönlendirir.

```
for (i = 0; i < 10; i++) {
    if (i === 3) { continue; }
    text += "Sayı: " + i + "<br>";
}
```

JavaScript JSON

JSON veri saklamak ve taşımak için bir biçimdir.

- JSON temsil **J** ava **S** cript **O** bject **N** otasyon
- JSON hafif veri değişim formatıdır
- JSON bağımsız dildir *
- JSON "kendini açıklayan" ve anlaşılması kolaydır

- SON, programlama dilinden bağımsız olan Xml'e alternatif olarak kullanılan javascript tabanlı veri değişim formatıdır. JSON'un amacı veri alış verişi yaparken daha küçük boyutlarda veri alıp göndermektir.Bu özellikleri sayesinde JSON ile çok hızlı web uygulamaları oluşturabilir.

```
{
'{"calisan":[' +
   '{"ad":"Ayşe",  "soyad":"Yılmaz" },' +
   '{"ad":"Mehmet","soyad":"Öztürk" },' +
   '{"ad":"Ömer",  "soyad":"Çetin" }
  ]
}
```

JavaScript JSON

JSON sözdizimi kuralları
•Veri ad / değer çiftleri içinde yazılır
•Veri virgül ile ayrılır
•Köşeli parantezler diziler tutar

JSON nesneleri küme parantezi içine yazılır.

Sadece JavaScript gibi nesneler birden fazla ad / değer çiftlerini içerebilir:

```
{"ad":"Ayşe", "soyad":"Yılmaz"}
```

JavaScript JSON

JSON Diziler

JSON diziler köşeli parantez içinde yazılır.

Sadece JavaScript gibi, bir dizi nesneleri içerebilir:

```
"calisan":[' +
   '{"ad":"Ayşe",  "soyad":"Yılmaz" },' +
   '{"ad":"Mehmet","soyad":"Öztürk" },' +
   '{"ad":"Ömer",  "soyad":"Çetin" }
]
```

JavaScript JSON

İlk olarak, JSON sözdizimi içeren bir JavaScript string oluşturun:

```
var metin = '{"calisan":[' +
    '{"ad":"Ayşe",  "soyad":"Yılmaz" },' +
    '{"ad":"Mehmet","soyad":"Öztürk" },' +
    '{"ad":"Ömer",  "soyad":"Çetin" }]
     }';
```

Sonra, bir JavaScript nesnesine dönüştürmek için yerleşik fonksiyon JSON.parse() kullanın:

```
var obj = JSON.parse(metin);
```

JavaScript JSON

```
<!DOCTYPE html>
<html>
<body>
<h2>JSON String ile Nesne Oluştur</h2>
<p id="demo"></p>
<script>
   var metin = '{"calisan":[' +
   '{"ad":"Ayşe","soyad":"Yılmaz" },' +
   '{"ad":"Mehmet","soyad":"Öztürk" },' +
   '{"ad":"Ömer","soyad":"Çetin" }]}';
   obj = JSON.parse(metin);
   document.getElementById("demo").innerHTML =
   obj.calisan[1].ad + " " + obj.calisan[1].soyad;
</script>
</body>
</html>
```