**JavaScript Yazım Kuralları;**

Bu sayfamızda JavaScript'in yazım kuralları üzerinde duracağız. JavaScript'te hata yapmanın bedeli çok ağırdır. Nerede hata yaptığınızı bulmak uzun zaman alır. (Bunun için JavaScript editörleri vardır, siteden çekebilirsiniz.)

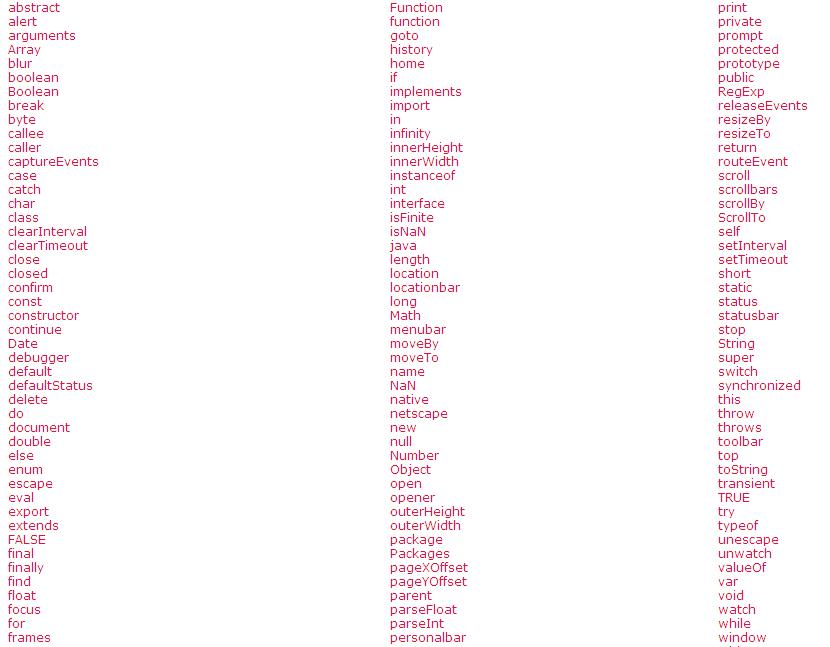
JavaScript tıpkı HTML gibi bırakılan boşlukları dikkate almaz. Yani **sayi=5** ile **sayi = 5** arasında fark gözetmez.

JavaScript'te komutlar ; (noktalı virgül) işareti konularak yanyana yazılabilir. Aynı satırda olmasa da her komuttan sonra ; işareti koymak yararlıdır, kodlar arasında ayıraç vazifesi görür. Gerçi ben pek yapmıyorum ama istiyorsanız siz yapın.

**JavaScript Nesnelerine İsim Koyma**

JavaScript nesnelerine isim koyarken aklımızdan çıkarmamamız gerekenlerin başında, JavaScript'in küçük harf, büyük harf ayrımı yaptığıdır. Mesela **AritmetikOrtalama()** isimli bir fonksiyonu daha sonra **aritmetikortalama()** ismiyle çağıramayız. JavaScript bu durumu iki ayrı fonksiyon gibi değerlendirir.

JavaScript nesnelerinin (değişken, fonksiyon vb.) ismi bir sayıyla başlamaz ama sonraki karakterler sayı olabilir; arada boşluk bırakılmaz, boşluk yerine \_ işareti kullanılır. Bunları değişkenler konusunda anlatmıştık. JavaScript nesneleri bu kurallara uygun her kelimeyi isim olarak alması da söz konusu değil, JavaScript komutlarını, nesne ismi olarak kullanamayız. Bazı isimlerde daha sonra komut olarak JavaScript'e girer diye ayrılmıştır ve onlarda nesne ismi olarak kullanılamaz. Nesne ismi olarak kullanılamayan bu kodları aşağıda listeledim:

****

**JavaScript'te Özel İşaret Gösterimi**

JavaScript'te HTML nesnesini ifade ederken bazı özel işaretlerin önlerine \ işareti koymalıyız.

**<html>**

**<head>**

**<title>JavaScript Uygulaması</title>**

**</head>**

**<body>**

**<script language="JavaScript">**

**document.write ("Ali'nin köpeği "Hav hav!" dedi.");**

**</script>**

**</body>**

**</html>**

Bu uygulama çalışmaz ve JavaScript hatası verir. Bu kodda JavaScript içiçe tırnakları, ayrı ayrı tırnak işareti olarak görür. Doğrusu şöyle yazılır:

**<html>**

**<head>**

**<title>JavaScript Uygulaması</title>**

**</head>**

**<body>**

**<script language="JavaScript">**

**document.write ("Ali'nin köpeği \"Hav hav!\" dedi.");**

**</script>**

**</body>**

**</html>**

JavaScript'te ' (tek tırnak) işareti ile " (çift tırnak) işareti arasında işlevsel bakımdan bir fark yoktur. Ancak içiçe tırnak kullanılan durumlarda içtekini başka, dıştakini başka tırnak işaretiyle göstermek yararlıdır. Örneğin:

**<input type="button" value="Bas" onclick="alert('Düğmeye Bastınız!');">**

Burada " (çift tırnak) işaretiyle JavaScript ifadesi HTML parametresine tanıtılmış oluyor. ' (tek tırnak) ifadesi ile JavaScript kodunun yazdırdığı metin belirtilmiş oluyor. Böylece hangi tırnağın nerede bittiği konusunda herhangi bir karışıklık olmuyor.

**Yorum Kodları**

JavaScript ile yapacağımız programlar bazen o kadar büyük ve karmaşık olur ki, neler yaptığımızı biz bile unuturuz. Gerek sonradan hatırlamak için, gerekse bizden başka kodları okuyacak olanlara yardımcı olmak için, kodların arasına bazı notlar düşeriz. Bu tür notları doğrudan yazarsanız JavaScript hatasına sebep olur veya programınızın işlevini değiştirirsiniz. Bu notları düşmek için not satırı önüne // işaretini koyarız. Bu işaret bir satırlık not bırakmamızı sağlar ve JavaScript kodlarını etkilemez:

**<html>**

**<head>**

**<title>JavaScript Uygulaması</title>**

**</head>**

**<body>**

**<script language="JavaScript">**

**//Bu uygulama JavaScript yazım kuralları üzerinedir.**

**document.write ("Ali'nin köpeği \"Hav hav!\" dedi.");**

**</script>**

**</body>**

**</html>**

Bazen düşeceğimiz not birkaç satırdan oluşabilir. Bu durumda not **/\*...\*/** işaretleri arasına yazılır.

**<html>**

**<head>**

**<title>JavaScript Uygulaması</title>**

**</head>**

**<body>**

**<script language="JavaScript">**

/\*Bu uygulama JavaScript yazım kuralları üzerinedir.

İtiraf edeyim ki pek iyi bir uygulama olmadı,

Ali'ymiş, yok efendim köpeği hav hav dermiş.

İlkokul uygulamalarına benzedi.\*/

**document.write ("Ali'nin köpeği \"Hav hav!\" dedi.");**

**</script>**

**</body>**

**</html>**

JavaScript'in gözatıcı (browser) tarafından derlendiğini daha önce söylemiştim. Ancak bazı kişisel gözatıcılar ve eski sürüm gözatıcılar JavaScript'i desteklemiyor ve sayfada hatalar veriyor. Bu tip gözatıcıların JavaScript kodlarını tanıması için elimizden gelen birşey yok. Ama hiç olmazsa sayfada hataya neden olmamak için HTML'in <!-- yorum kodu kullanılır. Ayrıca yeni sürümlerin kodu görebilmesi için // yorum kodları da gereklidir. Eski sürüm gözatıcılar veya JavaScript'i desteklemeyen kişisel gözatıcılar JavaScript kodlarını HTML yorumu olarak görür ve sayfada bozulmaya neden olmaz; kullanıcı JavaScript'in faydalarından yararlanamasa da, sayfayı düzgün bir şekilde görür. Aşağıda bunun nasıl yapıldığını görüyorsunuz.

**<html>**

**<head>**

**<title>JavaScript Uygulaması</title>**

**</head>**

**<body>**

**<script language="JavaScript">**

**<!-- Eski sürüm gözatıcılardan JS kodlarını saklayalım.**

**//Şimdi normal kodları yazabiliriz.**

**document.write ("Ali'nin köpeği \"Hav hav!\" dedi.");**

**//-->**

**</script>**

**</body>**

**</html>**