**JavaScript Fonksiyonlar;**

En iyi yazılımcı, az kodla çok şey yapabilendir. Acemi yazılımcıların sayfalarca kod yazarak yaptığı programları, usta yazılımcılar yarım sayfada yapabilir. Bunun inceliklerini hem bu sayfada, hem de ileride yapacağımız ileri uygulamalarda öğreneceksiniz.

Diyelim ki programda bir sürü işlem yapacağız ama öyle bir işlem var ki 50 defa yapacağız. (Mesela "Ali gel." yazdıracağız.) Programa bu işlemi 50 defa yaptırmak için o işlem kodunu 50 defa yazdırmak zorunda kalırız ve programımız sayfalarca sürebilir. Üstelik bu işlem kodlarının birkaç satır değil de yarım sayfa sürmesi söz konusu da olabilir. Bu durumda 50 defa o yarım sayfalık kodu yazmak zorunda mı kalacağız?

Hayır. O yarım sayfalık işlem kodunu bir fonksiyon haline getiririz ve 50 defa o fonksiyonu tek kelime olan fonksiyon ismi ile çağırırız. Bu durumda 30-40 sayfalık bir kodu 3-4 sayfaya indirmiş oluruz.

Peki bir işlem nasıl fonksiyon haline getirilir? İşte sayfamızın konusu bu.

**Fonksiyon Yapısı**

Fonksiyonlar **<head>...</head>** bölümünde yer alır. Bir fonksiyonun genel yapısı aşağıdaki gibidir:

**function** *fonksiyonun adı* **(***argüman***1***, argüman***2***,....,argüman****N*)**

**{**

*İŞLEMLER*

**}**

Fonksiyonlara koyduğumuz isim, fonksiyonu çağırmak için gereklidir. Her fonksiyonun adı olmalıdır, zira fonksiyonlar bir olay (durum) karşısında çağrılabilsin diye yapılır. Bu olay bir düğmenin tıklanması, bir radyo veya kontrol kutusunun seçilmesi, bir HTML nesnesinin üzerine fare ile gelinmesi olabilir. (HTML'de form nesnelerini unutan arkadaşlarımıza tekrar bakmalarını salık veririm. Bu olaylara ilerdeki **Olaylar** sayfasında ayrıntılı olarak değineceğim. Ama şimdilik **onclick** (tıklanıldığı takdirde) olay parametresini kullanacağız. Bu parametre form araçlarına olduğu gibi, hiperlink, resim gibi diğer HTML nesnelerine de atanabilir. **onclick** = *Fonksiyon Adı()* şeklinde kullanılır.

**Argümansız Fonksiyonlar**

Fonksiyonlar çeşitli argümanları kullanarak işlem yapabilirler. Bu argümanlar değişken, nesne veya değer ifadesi olabilir. Bunun yanısıra fonksiyonların argüman kullanma zorunluluğu yoktur; aşağıdaki uygulama argümansız bir fonskiyon içermektedir.

**<html>**

**<head>**

**<title>Argümansız Fonksiyon</title>**

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

**//Fonksiyonu hazırlıyoruz.**

**function mesaj()**

**{**

**alert("Hiç bir mesajınız yok.");**

**document.write("Mesaja baktınız...");**

**}**

**</script>**

**</head>**

**<body>**

**<form>**

**Mesaja bakmak istiyorsanız, düğmeye basın.<br>**

**<!--Düğmeye basıldığında fonksiyon çağrılır.-->**

**<input type="button" onclick="mesaj()" value="Mesaj Kontrol">**

**</form>**

**</body>**

**</html>**

Eeee, nasıl? Kendinizi programcı gibi hissetmeye başladınız mı? Daha neler yapacağız, neler. Bu uygulamamızda **mesaj()** isimli argümansız bir fonksiyon kullandık. Bu fonksiyonu kullanmamız şart değildi elbette. Fonksiyon yaratmak yerine, kodları **<body>** bölümüne yazabilirdik. Ancak bu fonksiyonu buradaki gibi bir defa değil de beş-on yerde kullansaydık, fonksiyon sayesinde bir sürü kod yazmaktan kurtulacaktık. (Bunu bir defa daha açıklamaya kalkarsam, okuyucuların zekasını küçümsemiş olacağım.)

**Argümanlı Fonksiyonlar**

Şimdi de argümanlı bir fonksiyon yapalım. Aşağıdaki uygulamada iki argümanlı bir fonksiyon göreceksiniz:

**<html>**

**<head>**

**<title>Argümanlı Fonksiyon</title>**

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

**//Fonksiyonu hazırlıyoruz.**

**function puan(a, b)**

**{**

**alert("Verdiğiniz notu göndermek için basın.");**

**document.write(a + " isimli aktriste " + b + " notunu verdiniz.");**

**}**

**</script>**

**</head>**

**<body>**

**<form>**

**<input type="button" value="Notu Gönderin" onclick= "puan('Onur Unurlu', '100')">**

**</form>**

**</body>**

**</html>**

Bu uygulamada fonksiyonumuzun iki tane argümanı var. Bunlar: **'Onur Unurlu'** ve **'100'**. Yanlarındaki **'** işaretleri bu argümanların bir metin ifadesi olduğunu anlatıyor.

**Değer Atamalı Fonksiyonlar**

Bir başka tür fonksiyon yapısı daha vardır. Buna **değer atamlı fonksiyon** diyoruz. Bu fonksiyon türü argümanlı veya argümansız olabilir. Ayırtedici özelliği yaptığı işlemleri bir değere dönüştürüp, bu değeri değişkenlere veya HTML nesnelerine aktarması. Bu fonksiyonun yapısı şöyledir:

**function** *fonksiyonun adı* **(***argüman1, argüman2,....,argümanN***)**

**{**

*İŞLEMLER*

**return** *değer*

**}**

Buradaki **return** kodu, işlemlerle bulunan bir değeri fonksiyona atamakta kullanılır. Aşağıdaki uygulamayla daha iyi anlayacaksınız.

**<html>**

**<head>**

**<title>Değer Atamalı Fonksiyon</title>**

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

**//Fonksiyonu hazırlıyoruz.**

**function ustubes(x)**

**{**

**x = x \* x \* x \* x \* x;**

**return x**

**}**

**</script>**

**</head>**

**<body>**

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

**var a1, a2, a3, a4, a5, a6, a7, a8, a9;**

**//Dokuz değişkeni ustubes() fonksiyonunu kullanarak değer atanıyor.**

**a1 = ustubes(1);**

**a2 = ustubes(2);**

**a3 = ustubes(3);**

**a4 = ustubes(4);**

**a5 = ustubes(5);**

**a6 = ustubes(6);**

**a7 = ustubes(7);**

**a8 = ustubes(8);**

**a9 = ustubes(9);**

**document.write("1,2,3,4,5,6,7,8 ve 9 sayılarının üstü beş hali sırasıyla aşağıda yazılmıştır.<br>");**

**document.write(a1 +", " + a2 +", " + a3 +", " + a4 +", " + a5 +", " + a6 +", " + a7 +", " + a8 +", " + a9);**

**</script>**

**</body>**

**</html>**