

Koşul operatörleri: if, '?'

Bazı durumlarda koşula göre farklı eylemler yapmak isteyebilirsiniz.

"?" operatörü veya if cümlesi bu koşulları kontrol etmenizi sağlar.

"if" cümlesi

"if" cümlesi koşulu alır ve kontrol eder sonucunda true ise kodu çalıştırır.

Örneğin:

```
let yil = prompt('ECMAScript-2015 standartları hangi yıl yayınlanmıştır?', '');  
if (yil == 2015) alert( 'Evet doğru!' );
```

Yukarıdaki örnekte "if" basit bir eşitliği kontrol eder `yil == 2015`, fakat bu kontrol elbette daha karmaşık olabilir.

Eğer birden fazla komut çalıştırmak isterseniz bunu süslü parantez içinde yapabilirsiniz.

```
if (yil == 2015) {  
    alert( "Evet doğru!" );  
    alert( "Bravo!" );  
}
```

Her if kullandığınızda süslü parantez kullanmanız okunurluğu artıracaktır. Tek satır bile yazsanız süslü parantez içinde yazmanız önerilir.

Boolean dönüştürme

`if(...)` cümlesi parantez içine yazdığımız ifadeyi çalıştırır ve bunu sonucu boolean tipinde dönderir.

[Tip Dönüşümleri](#) bölümünü hatırlarsanız:

- 0, boş karakter "", null, undefined ve NaN false olarak döndürülür. Bunlara falsy yani yanlış değerler de diyebiliriz.
- Diğer değerler ise true olur ve bunlara truthy veya doğru değerler de denebilir. (Not: Bunların tam karşılıklarını bulamadım)

Örneğin aşağıdaki kod satırı hiçbir zaman çalışmayacaktır:

```
if (0) { // 0 false döndürür  
    ...  
}
```

... Aşağıdaki ise her zaman çalışacaktır:

```
if (1) { // 1 her zaman true döndürür  
    ...  
}
```

Ayrıca şart yazmadan önce de kodun değerlendirilmesi yapılabilir.

```
let sonuc = (year == 2015); // eşitlik doğru yanlış değerlendirmesi yapmaya yarar.  
if (sonuc) {  
    ...  
}
```

“else” cümlesi

if cümlesi opsiyonel olarak “else” bloğu da içerebilir. Bu eğer if parantezi içerisinde yazdığımız kod yanlış ise çalışır.

Örneğin:

```
let yil = prompt('ECMAScript-2015 standartları hangi yıl yayınlanmıştır?', '');  
if (yil == 2015) {  
    alert( 'Doğru!' );  
} else {  
    alert( 'Nasıl bu kadar yanlış yapabiliyorsun!' ); // 2015 dışındaki her değerde  
    burası çalışır.  
}
```

Birden fazla koşul: “else if”

Bazı durumlarda birden fazla durumu kontrol etmeniz gerekebilir. Bu durumlarda else if cümlesi kullanabilirsiniz.

Örneğin:

```
let yil = prompt('ECMAScript-2015 standartları hangi yıl yayınlanmıştır?', '');  
if (yil < 2015) {  
    alert( 'daha sonra...' );  
} else if (yil > 2015) {  
    alert( 'daha önce' );  
} else {  
    alert( 'Kesinlikle!' );  
}
```

Yukarıdaki kodda önce `yil < 2015` kontrolü yapılır. Eğer bu değerlendirme yanlış ise bir sonraki koşula geçilir. Eğer `year > 2015` doğru ise bu koşul içindeki alarm fonksiyonu çalışır. Diğer hallerde son alert fonksiyonu çalışır.

Sonuncusunda bir tane daha else if bloğu olabilirdi: `else if (yil == 2015)`

Üçlü operatör ‘?’

Bazen alert yerine bir değişkene değer atamak isteyebilirsiniz.

Örneğin:

```
let girisIzni;  
  
let yas = prompt('Kaç yaşındasın?', '');  
  
if (yas > 18) {  
    girisIzni = true;  
} else {  
    girisIzni = false;  
}  
  
alert(girisIzni);
```

“üçlü” veya “soru işareti” operatörü yukarıdaki işlemi daha kolay yapmanızı sağlar.

Operatör "?" işareti ile ifade edilir. Resmi tanımında “üçlü” kullanılmasının sebebi üç tane operanddan oluşmasından dolayıdır. Aslında JavaScript dilinde 3 tane operandı olan başka bir operatör yoktur.

Yazımı şu şekildedir:

```
let sonuc = koşul ? deger1 : deger2
```

koşul değerlendirildikten sonra eğer doğru döner ise deger1 yanlış döner ise deger2 sonuç değişkenine atanır.

Örneğin:

```
let girisIzni = (yas > 18) ? true : false;
```

Aslında `yas > 18` etrafındaki parantezleri de kaldırabiliriz. Çünkü soru işareti `>` göre daha düşük önceliğe sahiptir. `yas > 18 ? true : false` şeklinde yazılsa da sonuç değişmeyecektir.

```
// the comparison operator "age > 18" executes first anyway
```

```
// (no need to wrap it into parentheses)
```

```
let girisIzni = yas > 18 ? true : false;
```

... Fakat yine de parantez kullanmanız önerilir. Bu okunabilirliği artırmaktadır.

Dikkate değer:

Yukarıdaki örnekte soru işaretini kaldırmak mümkündür çünkü karşılaştırmanın kendisi true/false döndürür:

```
// the same
```

```
let girisIzni = yas > 18;
```

Coklu '?'

Birden fazla "?" işareti kullanarak birden fazla koşula göre değer döndürme işlemini sağlayabilirsiniz.

Örneğin:

```
let yas = prompt('yaş?', 18);

let mesaj = (yas < 3) ? 'Ne tatlı şeysin sen öyle!' :

(yas < 18) ? 'Merhaba!' :

(yas < 100) ? 'Merhaba efendim!' :

'Ne garip bir yaşın var!';

alert( mesaj );
```

İlk başta neyin ne olduğunu anlamak zaman alabilir. Fakat daha yakından bakınca sıra ile değerin kontrol edildiğini görebilirsiniz.

1. İlk soru işareti `yas < 3` kontrolünü yapar.
 2. Eğer doğru ise Ne tatlı şeysin sen öyle! döndürür. Diğer türlü sonraki `":"`e gider ve `yas < 18` kontrolünü yapar.
 3. Eğer doğru ise Merhaba döndürür. Diğer türlü sonraki `":"`e gider ve `yas < 100` kontrolünü yapar.
 4. Eğer doğru ise Merhaba efendim döndürür. Diğer türlü sonraki `":"`e gider ve 'Ne garip bir yaş' döner.
- aynı if.else mantığı gibi

```
if (yas < 3) {

    mesaj = 'Ne tatlı şeysin sen öyle!';

} else if (yas < 18) {

    mesaj = 'Merhaba!';

} else if (yas < 100) {

    mesaj = 'Merhaba efendim!';

} else {

    mesaj = 'Ne garip bir yaşın var!';

}
```

Geleneksel olan olmayan '?' kontrolü

Bazı durumlarda '?' if in yerine kullanılabilir:

```
let firma = prompt('JavaScript hangi firma tarafından yaratılmıştır?', '');

(firma == 'Netscape') ?

    alert('Doğru!') : alert('Yanlış.');
```

Koşula göre `firma == 'Netscape'`, soru işaretinden sonra birinci bölüm veya ikinci bölüm çalışır.

Sonucu bir değere atanmamıştır. Amaç duruma göre doğrudan kodu çalıştırmak.

Soru işaretinin bu amaç doğrultusunda kullanılması önerilmez.

Yazımı if yazımından daha kısa olsa bile daha az okunabilir durumdadır. Aşağıda if ile yazımını görmekteyiz.

```
let firma = prompt('JavaScript hangi firma tarafından yaratılmıştır?', '');  
if (firma == 'Netscape') {  
    alert('Doğru!');  
} else {  
    alert('Yanlış.');
```

Okurken kodu dikey olarak okuruz. Bundan dolayı yazımın bir kaç satıra dağıtılması okumayı uzun satırlara göre daha kolay hale getirir.

'?' işaretinin ideal kullanımı sadece o ya da bu sorusudur. Daha uzun bir cümle için if kullanmalısınız.

Görevler

Eğer bir karakter 0 ise

önem: 1

Alert gösterilecek mi?

```
if ("0") {  
    alert( 'Merhaba' );  
}
```

JavaScript'in ismi

önem: 2

if..else bloğu kullanarak “JavaScript’in resmi ismi nedir?” sorusunu sorun.

Eğer kullanıcı “ECMAScript” cevabı verirse “Doğru” diğer türlü “ECMAScript olduğunu bilmiyormusun?” yazısını alarm olarak gösterin.

if..else cümlesi kullanarak prompt komutu ile soru sorup alert ile sonucu gösterin.

- 1, değer sıfırdan büyük ise,
- -1, sıfırdan küçük ise,
- 0, sıfır ise.

Bu örnekte kullanıcının gireceği tüm değerlerin sayı olduğunu varsayabilirsiniz. [demo src=“if_sign”]

'if' ile yazılan koşulu '?' çevirin.

önem: 3

Aşağıdaki if cümlesini “üçlü” operatöre '?' olarak yazın:

```
if (a + b < 4) {  
    sonuc = 'Düşük';  
} else {
```

```
sonuc = 'Yüksek';  
}
```

'if..else' i '?' formunda yazınız

önem: 4

Aşağıdaki if..else cümlesini “üçlü” operatörler '?' olarak yazın:

Okunabilirlik açısından kodu satırlara bölmeniz önerilir.

```
let mesaj;  
if (giris == 'Çalışan') {  
    mesaj = 'Merhaba';  
} else if (giris == 'Yönetici') {  
    mesaj = 'Sana da merhaba patron';  
} else if (giris == '') {  
    mesaj = 'Giriş yasak';  
} else {  
    mesaj = '';  
}
```