



Conditional Statements



- › Programlamada program akışını çeşitlendirmek için kullanılır.
- › Javascript te bunun için if, veya switch blokları kullanılır.



Conditional Statements

```
var x = 4;  
if(x>10){  
    x=x-5;  
}  
console.log(x);
```

- › if(mantıksal_ifade){ ... } şeklinde kullanılır.
- › Eğer mantıksal_ifade doğru ise { } arasındaki kodlar gerçekleştirilir. Aksi halde gerçekleştirilmez.



Conditional Statements

PRACTISE

50 sayısını
girilen sayıya
bölen, bu işlemi
eğer girilen sayı
sıfırdan farklı ise
yapan programı
yazınız.



```
const sayi = prompt("Sayı giriniz:");  
  
if(sayi!="0"){  
    const bolum = 50/Number(sayi);  
    console.log(bolum);  
}
```



Conditional Statements

```
var x = 4;  
if(x>10){  
    x=x-5;  
}  
else{  
    x=x+3;  
}  
console.log(x);
```

- › Eğer iki durum var ise if ten sonraki durum için bir else bloğu kullanılabilir.
- › Program aynı if blokları içinde sadece bir bloğun içine girer.



Conditional Statements

PRACTISE

Girilen 3 sayının ortalamasını bulan, eğer ortalama, ilk sayıdan büyükse "kâr", değilse "zarar" yazan programı yapınız.

```
const sayi1 = prompt("1.sayıyı giriniz:");
const sayi2 = prompt("1.sayıyı giriniz:");
const sayi3 = prompt("1.sayıyı giriniz:");

const ortalama =
(Number(sayi1)+Number(sayi2)+Number(sayi3))/3;

if(ortalama>Number(sayi1)){
    console.log("kâr");
}
else{
    console.log("zarar");
}
```



Conditional Statements

PRACTISE

Kullanıcının girmiş olduğu metin uzunluğu 2 ile 50 arasında ise «merhaba» yazan değilse hata veren programı yapınız



```
const isim = prompt("Adınızı giriniz:");

if(isim.length>=2 && isim.length<=50){
    console.log("Merhaba");
}
else{
    console.log("Hatalı giriş");
}
```



Conditional Statements

PRACTISE

Boy ve kilo değerlerini alıp vücut kütle endeksini bulup, ideal aralık dışındaki değerler için «ideal kilo değil», ideal aralıkta ise «ideal kilo» mesajı veren programı yazınız

Vücut kütle indeksi = $\text{kilo} / \text{boy}^2$

İdeal aralık : 18.5 - 25

```
const boy = prompt("Boy giriniz:");
const kilo = prompt("Kilo giriniz:");

const vki = Number(kilo) / (Number(boy) * Number(boy));

if(vki<18.5 || vki>25){
    console.log("İdeal kilo değil");
}
else{
    console.log("İdeal kilo");
}
```



Conditional Statements

```
var x = 4;  
if(x>10){  
    x-=5;  
}  
else if(x>4){  
    x+=3;  
}  
console.log(x);
```

- › Birden fazla durum var ise ilk if ten sonra her bir durum için bir else if kullanılabilir.
- › Durum sayısı çok ise switch statement kullanılabilir.



Conditional Statements

PRACTISE

Girilen maaş in miktarına göre farklı oranlarda zam yapan programı yazınız.

1000-2000: %30

2000-4000: %15

4000+ : %5

```
const maasStr = prompt("Maaşı giriniz:");  
var maas = Number(maasStr);
```

```
if(maas>=1000 && maas<2000){  
    maas *= 1.3;  
}  
else if(maas>=2000 && maas<4000){  
    maas *= 1.15;  
}  
else if(maas>=4000){  
    maas *= 1.05;  
}  
console.log(maas);
```



Conditional Statements

PRACTISE

Klavyeden girilen bir sayının 3 ve 5'e tam bölünebilme durumunu gösteren programı yazınız



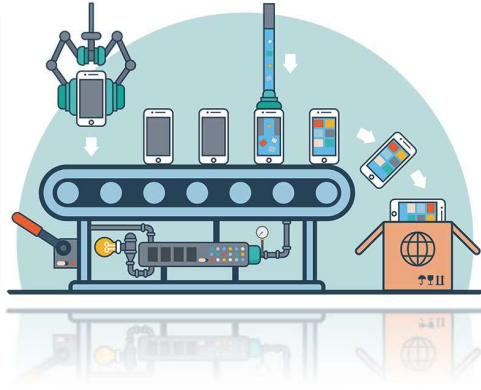
```
const sayi = prompt("Sayı giriniz:");
const kalan3 = sayi % 3;
const kalan5 = sayi % 5;
var sonuc="";

if(kalan3+kalan5==0){
    sonuc = "Sayı 3 ve 5'e tam bölünüyor";
}
else if(kalan3==0){
    sonuc = "Sayı 3'e tam bölünüyor";
}
else if(kalan5==0){
    sonuc = "Sayı 5'e tam bölünüyor";
}
else{
    sonuc = "Sayı 3'e ve 5'e tam bölünmüyor";
}

console.log(sonuc);
```



Loops



- › Belirtilen koşul sağlandığı sürece, belli kod bloklarını tekrar tekrar çalıştırmak için kullanılırlar.
- › Üretim hatlarındaki robot kollar buna örnek verilebilir.
- › Döngüler içinde conditional statement ler kullanılarak duruma göre farklı kodlar çalıştırılabilir.



Loops

JS Loops

› Javascript te farklı durumlar için kullanılacak döngü türleri bulunmaktadır

- › **For**
- › For of
- › For each
- › Map
- › **While**
- › **Do while**



Loops

For loop

```
for (var i=0; i<10; i++) {  
    .  
    .  
    .  
}
```

Döngünün
başlangıç
sayısı

Doğru
olduğu
süreçe
döngü tekrar
eder

Artış
miktarı

- › En temel döngü türüdür.
- › Tekrar ettirilecek kodun baştan kaç defa tekrar ettirileceği biliniyorsa bu tür döngüler kullanılabilir.



Loops

PRACTISE

Girilen 10 sayının ortalamasını bulan programı yapınız.

```
var toplam = 0;

for(var i=0;i<10;i++){
    const sayi = prompt("Bir sayı giriniz:");
    toplam += Number(sayi);
}
const ortalama = toplam / 10;

console.log(ortalama);
```