

# Uvod u kontrolni tok

U ovom ćemo istražiti kako možemo da koristimo blokove JavaScript-a kako bi pisali programe koji donose odluke.

*Kontrolni tok omogućava JavaScript programima da donose odluke i izvršavaju kod na osnovu stanja. Ako je određeno stanje tačno, izvršava se jedan blok koda. Ako je stanje netačno, izvršava se drugi blok koda.*

## if/else iskazi

Primer:

```
var potrebanOdmor = true;
if (potrebanOdmor === true)
{
  document.write('Pauza');
}
else
{
  document.write('Nastavi sa radom');
}
```

1. Linije koda između vitičastih zagrada nazivaju se blokovi. `if/else` iskazi imaju dva bloka koda. Ako je promenjiva `potrebanOdnor` jednaka `true`, program će izvršiti prvi blok koda. U suprotnom će izvršiti drugi blok koda.
2. `potrebanOdnor` je stanje koje treba proveriti unutar `if` zagrada. Pošto je jednako `true`, program će izvršiti kod unutar prve vitičaste zgrade `{ }`. Ignorisaće `else { ... }` deo. U ovom slučaju ćemo videti `Pauza` ispisano na ekranu.
3. U slučaju da je `potrebanOdnor` bilo `false`, samo bi kod unutar `else` bloka bio izvršen.

## True i False vrednosti

U prethodnom primeru smo imali `if/else` iskaze. Međutim, postoje drugi tipovi podataka osim Boolean-a. Sada ćemo istražiti `true` i `false` koncept za promenjive ostalih tipova podataka.

U JavaScript-u, sve promenjive i svi uslovi imaju svoju `true` i `false` vrednost.

```

var promenjiva1 = Postoji!;
if (promenjiva1)
{
  //
}
else
{
  //
}

```

U prvoj liniji gore navedenog programa kreirana je promenjiva `promenjiva1`. Vrednost ove varijable je string umesto boolean.

U drugoj liniji koda program proverava stanje `if (promenjiva1)`. Pisanjem naziva promenive kao uslova proveravamo tačnost `promenjiva1`. U ovom slučaju, `promenjiva1` vraća vrednost `true`.

Ukoliko bi promenili `if (promenjiva1)` u `if (promenjiva2)`, tada bi uslov bio netačan jer ne postoji promenjiva pod nazivom `promenjiva2`.

Sve promenjive koje su kreirane, postavljene su kao `true` (tj. računaće se kao `true` u uslovima kontrolnog toka), osim ako one sadrže jednu od šest vrednosti navedenih u nastavku:

- `false`
- `0` ili `-0`
- `""` ili `' '` (prazan string)
- `null`
- `undefined`
- `NaN` (Not a Number)

## True i False vrednosti II

U programiranju često proveravamo da li je neki izraz tačan ili netačan.

```

var padaKisa = true;
if (padaKisa)
{
  document.write('Ponesi kisobran!');
}
else
{
  document.write('Uzivaj u suncu!');
}

```

U primeru iznad iskaz je jednostavan `if (padaKisa)`.

JavaScript takođe pruža operator za zamenu vrednosti true i false - uzvičnik (!).

```
var baterijaPuna = true;
if (!baterijaPuna)
{
    document.write('Prikljucite punjac!');
}
else
{
    document.write('Baterija je puna!');
}
```

U primeru iznad program proverava da li `baterijaPuna` vraća rezultat `false`. Ukoliko `baterijaPuna` je `true`, drugi blok koda će biti izvršen.

## Operatori poređenja

Pored provere da li promenjiva ima vrednost `true` ili `false`, ponekad je potrebno upoređivati promenjive sa drugim vrednostima. To možemo postići sa operaterima poređenja.

Vrste poređenja:

- Manje: `<`
- Veće: `>`
- Manje ili jednako: `<=`
- Veće ili jednako: `>=`

Rezultat ovih poređenje vraća vrednost `true` ili `false`.

## Operatori poređenja II

Postoje još dva poređenja koja možemo koristiti. Možemo da proverimo da li dve vrednosti jesu ili nisu jednake jedna drugoj.

1. Da proverimo da li su dve vrednosti jednake jedna drugoj koristimo `==` ili `===`.  
`==` proverava da li su dve vrednosti jednake  
`===` proverava da li su dve vrednosti jednake i da li im je isti tip podataka
2. Da proverimo da li su dve vrednosti nisu jednake jedna drugoj koristimo `!=` ili `!==`.  
`!=` proverava da li su dve vrednosti različite

`!==` proverava da li su dve vrednosti različite i da li im je različit tip podataka

`=` (jedno jednako) ne treba mešati sa poređenjem, ono se koristi samo za dodavanje vrednosti.

## else if iskazi

Upoznali smo se sa `if/else` iskazima koji odgovaraju na pitanja koja su da ili ne, ali to nam nije dovoljno kada imamo pitanja sa više mogućih stanja.

Više mogućih stanja kod `if/else` iskaza možemo dodati sa `else if`.

U primenu ispod možemo videti kako se to uklapa u dosadašnje `if/else` iskaze:

```
var semafor = 'zeleno';
if (semafor === 'crveno')
{
  document.write('Stop');
}
else if (semafor === 'zuto')
{
  document.write('Uspori');
}
else if (semafor === 'zeleno')
{
  document.write('Kreni');
}
else
{
  document.write('Oprez, nepoznato stanje');
}
```

1. Kreirali smo promenjivu `semafor` i dodali joj vrednost `zeleno`.
2. Nakon toga napravili smo `if/else` iskaz sa više mogućih stanja, pomoću `else if`. U ovom primeru `else if` omogućava višestruku proveru vrednosti promenjive `semafor` i ispis različitih stvari u zavistosti od upisanje boje..
3. Poslednji blok označen samo sa `else` hvata svaku nepredviđenu opciju i ispisuje opštu poruku.

## Logički operatori

Od logičkih operatora JavaScript koristi "i" i "ili".

1. Za “i” koristimo `&&`.
2. Za “ili” koristimo `||`.

Primer:

```
if (semafor === 'zeleno' && pesaci === false)
{
  document.write('Kreni');
}
else
{
  document.write('Stop');
}
```

1. U primeru iznad proveravamo da li je vrednost promenjive `semafor` `'zeleno'` i (`&&`) da li je vrednost promenjive `pesaci` `false` ukoliko su oba uslova ispunjena na ekranu će biti ispisano `Kreni`.
2. Ukoliko bilo koji od uslova nije ispunjen, na ekranu će biti ispisano `Stop`.

Kao i svi operatori koje smo spominjali ranije, logički operatori takođe kao rezultat vraćaju `true` ili `false`.

Logički operatori su korisni prilikom pisanja `if/else` iskaza jer nam omogućavaju da istovremeno proveravamo više promenljivih.

## switch iskazi

Korišćenje `else if` je odlično kad nam je potrebno da program razmotri nekoliko različitih stanja, međutim to ima određena ograničenja.

Ukoliko bi želeli da napišemo program sa 25 različitih stanja, to bi zahtevalo mnogo koda i bilo bi tesko za čitanje i razumevanje.

Kada nam je potrebno toliko puno `else if` stanja, možemo ih pretvoriti u `switch` iskaze pomoći kojih će kod biti mnogo jasniji i čitljiviji.

Za računar `switch` i `if/else` iskazi su potpuno isti.

`switch` izgleda ovako:

```
var proizvod = 'limun';
switch (proizvod)
{
case 'jabuka':
document.write('Jabuke su 81 dinar');
break;
case 'krompir':
document.write('Krompir je 54 dinara');
break;
case 'limun':
document.write('Limun je 149 dinara');
break;
default:
document.write('Nepoznat proizvod');
break;
}
```

1. `switch` ključna reč inicira iskaz i praćena je sa zagradama ( ... ), koje sadrže uslov sa kojim će svaki `case` biti upoređen. U primeru iznad uslov je `proizvod`.
2. Unutar bloka, { ... }, postoji slučaj `case`. `case` je kao `else if` deo `if/else` iskaza. Reč koju sadrži prvi `case` je `'jabuka'`. Ukoliko je `proizvod` jednako `'jabuka'`, sadržaj tog `case` -a će biti izvršen.
3. `proizvod` je jednako `'limun'`, tako da će prvi i drugi `case` biti preskočeni. Treći case će biti izvršen jer `'limun'` se poklapa sa vrednošću promenjive `proizvod`. Ovaj deo koda će ispisati `Limun je 149 dinara`.
4. Tada se program zaustavlja sa ključnom reči `break`. Ova ključna reč će sprečiti `switch` da izvrši ostale slučajeve. Bez `break` program bi izvršio sve slučajeve koji se poklapaju, uključujući default kod.
5. Na kraju svakog `switch` iskaza postoji `default` slučaj. Ukoliko nijedan od `case`-ova ne bude true, onda će ovaj kod biti izvršen.

## Zadatak 1

Potrebno je napisati JavaScript program koji na osnovu upisanog trenutnog vremena izpisuje prikladan pozdrav za to doba dana.

- Vreme se upisuje pomoću prompta u 24-časovnom formatu (samo sati) i čuva u promenljivoj
- Nakog toga se pomoću if, else i if else iskaza proverava vrednost promenjive i na ekranu ispisuje prikladan pozdrav

Uslovi:

- Ukoliko je manje od 10 časova ispisuje se Dobro jutro
- Ukoliko je manje od 19 časova ispisuje se Dobar dan
- Ukoliko je više od 19 ispisuje se Dobro veče

## Zadatak 2

Potrebno je napisati JavaScript program za pogađanje nasumičnog broja od 0 do 10.

- Nasumičan broj od 0 do 10 se generiše pomoću matematičke metode i smešta u promenjivu
- Nakon toga korisnik pomoću prompta unosi željeni broj, vrednost se čuva u promenljivoj
- pomoću if, else iskaza upoređuju se vrednosti unetog i nasumičnog broja i ispisuje se poruka da li je korisnik pogodio broj.



### Zadatak 3

Potrebno je napisati JavaScript program koji na osnovu upisanih bodova ispisuje dobijenu ocenu.

- Ocena se upisuje pomoću prompta i čuva u promenljivoj
- Nakog toga se pomoću if, else i if else iskaza proverava vrednost promenjive i na ekranu ispisuje ocena.

*preko 95 bodova = ocena 10*

*85-94 bodova = ocena 9*

*75-84 bodova = ocena 8*

*65-74 bodova = ocena 7*

*55-64 bodova = ocena 6*

*ispod 55 bodova = vidimo se u sledecem roku*

#### Zadatak 4

Potrebno je napisati JavaScript program koji daje nasumične predefinisane odgovore na zadata pitanja.

Pitanje se unosi pomoću prompta i čuva u promenljivoj.

Nasumični odgovori se definišu unutar switch iskaza, a biraju pomoću ugrađene matematičke metode za nasumičan broj.

*Primeri odgovora:*

*Definitivno da, Bez sumnje, Najverovatnije, Bolje da ne znas, Pitajte kasnije, Ishod nije tako dobar, Znaci ukazuju na ne, Moj odgovor je ne.*

Pitanje i nasumično izabrani odgovor se ispisuju na ekranu.