Ispis podataka

Pre nego što počnemo o tipovima podataka i metodama, moramo znati kako da ispišeno vrednosti na ekranu.

document.write() je komanda koja se koristi za ispis podataka na ekranu. Kao u sledećem primeru:

document.write("Hello!");

Hello! će biti ispisano na ekranu.

Na kraju svakog reda poželjno je staviti tačku zarez, iako će JavaScript često ispravno funkcionisati i bez toga jer na svaki red gleda zasebno.

# Tipovi podataka

Tipovi podataka su osnovni elementi svih programskih jezika, u njih smeštamo vrednosti koje kasnije u programu obrađujemo.

Ispod je naveden ispis četiti odnovna tipa podataka za JavaScript:

document.write('New York City');

document.write(40.7);

document.write(true);

document.write(null);

Njih čine:

* Stringovi — grupa karaktera (slova, razmaci, brojevi, or simboli) označeni jednostrukim ('Hello') ili dvostrukim navodnicima ("World!"). Iz primera iznad, 'New York City' je tip podataka string.
* Brojevi — bilo kakva brojčana vrednost, uključujući brojeve sa decimalama: 4, 1516, .002, 23.42. Iz primera iznad, 40.7 je tip podataka broj.
* Boolean — Može biti true ili false, piše se bez navodnika. Iz primera iznad, true je tip podataka boolean.
* Null — null nedefinisana vrednost.

# Definisanje promenjivih

Za definisanje promenljivih se koristi ključna reč var iza koje sledi ime promenljive.

U imenu promenljive se ne može koristiti razmak ili znak interpunkcije, ime promenjive ne može početi cifrom.

var name;

var x = 4;

Matematički operatori

JavaScript podržava sledeće matematičke operatore:

1. Sabiranje: +
2. Oduzimanje: -
3. Množenje: \*
4. Deljenje: /

document.write(3+4); // Equals 7  
document.write(5-1); // Equals 4  
document.write(4\*2); // Equals 8  
document.write(9/3); // Equals 3

# Osobine podataka

Kada unesemo novi podatak u JavaScript program, on se čuva kao instanca tipa podataka.

JavaScript će sačuvati 'Hello' kao string u memoriji računara, 40.7 se čuva kao broj,

Ali pored osnovne vrednosti podatka string 'Hello' ima dodatne vrednosti vezane za njega

Na primer svaki string ima osobinu length u kojoj se čuva vrednost od koliko se karaktera sastoji taj string

Toj vrednosti možemo pristupiti tako što dodamo tačku pa naziv osobine.

document.write('Hello'.length);

U primeru iznad, pozivamo osobinu length za string 'Hello'. Biće ispisan proj  5 zato što se Hello sastoji od pet karaktera.

Ugrađene metode

Dok se dužina stringa izračunava prilikom kreiranja, instanca stringa takođe ima metode koje izračunavaju nove informacije po potrebi. Kada se ove ugrađene metode pozovu, one kreiraju neke izlazne podatke. Ugrađene metode se pozivaju tako što se nakon instance podatka doda tačka, pa naziv metode i otvorena pa zatvorena obicna zagrada.   
Kao u primeru ispod:

document.write('Hello'.toUpperCase()); // 'HELLO'

document.write('Hey'.startsWith('H')); // true

* U prvoj liniji,  .toUpperCase() metoda je pozvana nad stringom 'Hello'. Kao rezultat metoda će vratiti string sa svim velikim slovima: 'HELLO'.
* U drugoj liniji .startsWith() metoda je pozvana nad stringom "Hey". Ova metoda uzima karakter iz zagrade, u ovom slučaju 'H' i proverava da li string nad kojim je pozvana počinje na taj karakter. Pošto 'Hey'počinje na 'H', kao rezultat dobijamo boolean true.

Listu ugrađenih metoda možete naći u [JavaScript documentaciji](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/prototype).

Biblioteke

Metode na instancama obično zahtevaju da kreirate instancu pre nego što ih možete koristiti. Ukoliko želite da pozovete metodu bez instance potrebne su JavaScript biblioteke.

Jedna od tih je Math biblioteka koja sadrži matematičke metode.

Primer .random() metode:

document.write(Math.random()); // random number between 0 and 1

Ovakvu metodu pozivamo tako što navedemo naziv biblioteke, tačku , naziv metode, pa otvorenu i zatvorenu zagradu. Metoda .random() kao rezultat vraca nasumičan broj od 0 do 1 .

Da bi generisali nasumičan broj od 0 do 50, potrebno je da rezultat pomnožimo sa 50,   
kao ovde:

Math.random() \*50;

Kao rezultat dobićemo decimalni broj, ukoliko želimo da ga zaokružimo na ceo broj možemo iskoristiti ugrađenu metodu Math.floor() na ovaj način:

Math.floor(Math.random() \*50);

Proces:

1. Math.random generiše nasumičan broj izneđu 0 i 1.
2. Množenjem sa 50, dobijamo broj između 0 i 50.
3. Na kraju, Math.floor zaukružuje vrednost na najbliži ceo broj.

# Komentari

Kako pišemo JavaScript program, možemo kreirati komentare u našem kodu.

Komentari ne utiču na program prilikom izvršavanja, ali mogu biti korisni za vas i druge programere, jer opisuju šta kod radi.

Postoje dve vrste komentara u JavaScript-u:

1. Jednolinijski komentar se označava sa dve kose crte //

// The first 5 decimals of pi

document.write('Pi is equal to '+3.14159);

1. Višelinijski komentar se označava sa /\* za početak komentara, a \*/ za kraj komentara.

/\*

document.write('All of this code');

document.write('Is commented out');

document.write('And will not be executed);

\*/

Ugrađeni operatori za računanje

U ovom delu razmatramo kako možemo da koristimo promenjive i matematičke operatore da izračunamo nove vrednosti i dodelimo ih promenljivoj.

var x = 4;

x = x + 1;

U primeru iznad kreirali smo promenjivu x i dodali jos vrednost 4. U sledecoj liniji smo pomoću x = x + 1 povećali vrednost x sa 4 na 5.

U JavaScript-u ovaj proces se može skratiti korišćenjem ugrađenih operatora za računanje kao u primerima ispod:

var x=4;

x+=2; // x equals 6

var y=4;

y-=2; // y equals 2

var z=4;

z\*=2; // z equals 8

var r=4;

r++; // r equals 5

var t=4;

t--; // t equals 3

1. Prve tri operacije(+=, -=, i \*=) izvršavaju matematičku operaciju (+, -, or \*) koristeći broj sa desne strane,a zatim dodaju novu vrednost promenjivoj.
2. Druge dve operacije su operacija inkrementa (++) i dekrementa (--) operators. Ovi operatori su zaduženi za uvećavanje, odnosno umanjivanje vrednosti za jedan.

# Umetanje stringova

JavaScript omogućava umetanje vrednosti promenjivih unutar stringova.

var myPet='armadillo';

document.write('I own a pet '+myPet+'.'); // Output: 'I own a pet armadillo.'

U ovom primeru sačuvali smo vrednost 'armadillo' u promenjvu myPet .U sledećoj liniji pomocu operatora + kombinovali smo tri stringa: I own a pet, vrednost promenjive myPet, i ..Kao rezultat dobijamo:

I own a pet armadillo.

Umetanje stringova II

Drugi način za umetanje stringova:

1. Prilikom ispisa umesto navodnika koristimo backticks (nalazi se levo od broja 1 ).
2. Promenjivu na ovaj način ubacimu unutar rečenice ${myVariable}, ovaj put nam nije potreban operator  +.

Primer:

var myPet='armadillo'

document.write(`I own a pet ${myPet}.`) // Output: 'I own a pet armadillo.'

Rezultat:

I own a pet armadillo.

Prompt

Metod prompt() prikazuje prozor koji omogućava korisniku unos podataka.

Prompt se obično koristi ako želite da korisnik unese podatak pre ulaska na stranicu.

Metod prompt() vraća unesenu vrednost ako korisnik klikne na "OK", ako korisnik klikne na "cancel", metoda vraća null.

var person=prompt('Please enter your name');

Zadatak 1:

# Konvertor temperatura

**1.**Kreirajte promenjivu kelvin i napravite da se pomocu prompt-a unosi njena vrednost.

**2.**Celzijus je sličan Kelvinu, jedina razlika je da je Celzijus 273 stepeni manji od Kelvina.

Tako da se Kelvini konvertuju u Celzijuse oduzimanjem 273 od vrednosti promenjive kelvin. Rezultat sačuvajte u promenjivu celzijus.

**3.**Pomoću navedene jednačine izračunajte Farenhajte i rezultat sačuvajte u promenjivu farenhajt.

farenhajt = celzijus \* (9/5) + 32

**4.**Kada konvertujete Celzijuse u Farenhajte, često dobijate decimalni broj. Iskoristite ugrađenu metodu da dobijete ceo broj. Rezultat sačuvajte u promenjivu farenhajt .

**5.**Rezultat ispisite na ekranu u formatu:

Temperatura je TEMPERATURA stepeni farenhajta.

TEMPERATURA zamenite sa vrednošću promenjive farenhajt.

**6.**Pokrenite program.

Zadatak 2:

Na osnovu koda prethodnog zadatka napraviti program za izračunavanje površine i obima pravougaonika.  
  
Program treba da funkcioniše tako da se promoću dva prompt-a unose dužine stranica a i b.  
Nakon toga program treba da izbaci rezultat u formatu:

Povrsina pravougaonika stranica a = A i b = B iznosi POVRSINA, a obim OBIM.

A i B zamenite sa vrednošću dužina stanica a i b.

POVRSINA zamenite sa vrednošću površine pravougaonika.

OBIM zamenite sa vrednošću obima pravougaonika.