# **Kodo Sakiniai, Išraiškos, konsolė, kintamieji**

**Statement, Expression and Variables**

## **Kodas - Code**

Šaltinio kodas (source code) arba tiesiog kodas, tai yra kombinacija instrukcijų kurios pasako kompiuteriui ką daryti. Paprastai kodas yra saugomas teksto faile su atitinkamu failo tipu. Javascript failo tipas turi .js pabaigą. Javascript koda taip pat galima rašyti tiesiogiai naršyklės konsolėje.

Atitinkamo formato instrukcijos vadinamos kompiuterio kalba arba dar vadinama sintaksė. Panašiai kaip mes kalbame žmonių kalba, turime atitinkamą sintaksę, kurios pagalba galime reikšti mintis ir perteikti informaciją kitiems.

## **Programavimo sakinys - Statement**

Kompiuterio kalba išreikšti žodžiai, skaičiai, aritmetiniai ir kt. operatoriai kurie nurodomi vykdyti kažkokią užduotį yra vadinama programavimo sakiniu arba ang. statement.

| const a = b \* 2; |
| --- |

Raidės a ir b yra vadinami kintamieji (ang. variables), tai lyg būtų konteineriai laikyti nustatytai reikšmei (ang. value). Programavime kintamieji išlaiko reikšmes kaip pavyzdžiui skaičių reikšmės, raidės, objektai ir masyvai, o vėliau jie panaudojami kode.

Simboliai "=" (lygybės ženklas) ir "\*" (daugybos ženklas) yra vadinami operatoriai. Jie atlieka atitinkamus veiksmus su kintamaisiais kaip pvz priskyrimas reikšmės arba matematinė daugyba.

Kompiuterinės programos tai yra tokių programavimo sakinių grupės. O juos sudėjus kartu yra aprašomi žingsniai kompiuteriui, kuriuos atlikus įsigyvendina jūsų norimos programos siekius..

Javascript programavimo sakinys paprastai uždaromas kabliataškiu. Nors šita daryti nėra būtina. Ir tai priklauso nuo kompanijos ar komandos susitarimo. Visgi teoriškai įmanoma padaryti programines klaidas, vien todėl, kad kabliataškis nebuvo uždėtas.

## **Išraiška - Expressions**

Programavimo sąkinys (statement) yra sudarytas iš vienos ar daugiau išraiškų.. Bet kokia išraiška yra nuoroda į kintamąjį, jo reikšmę arba rinkinį kintamųjų ir jų reikšmių įkombinavus operatorius.

Pavyzdys:

| **const a = b \* 2;** |
| --- |

Šis programavimo sakinys turi keturias ekspresijas:

1. Skaičius du yra tiesioginės reikšmės du išraiška.

2. Raidė b yra kintamojo išraiška.

3. b \* 2 yra aritmetinės operacijos išraiška.

4. a = b \* 2 yra reikšmės priskyrimo išraiška.

## **Reikšmė - Value**

## Reikšmė tai yra duomenų tipas ir jo vertė, kurį tuo momentų saugo išraiška. Tai jei turime išraišką 2 \* 2, tai šios išraiškos reikšmė yra 4.

Jei turime kintamąjį x, ir jam priskiriama išraišką x = 2 \* 2, tai galima sakyti, kad x kintamojo reikšmė yra lygi keturiems.

Programavime dažnai būna, kad tam tikros išraiškos dalys dar neturi reikšmės ir ji bus apskaičiuojama vėliau. Taip dažnai būna, kuomet kintamųjų ar funkcijų reikšmės dar nėra apskaičiuotos arba bus apskaičiuotos vėliau.

| b = 2; *// b reikšmė yra 2* a = b \* 2; *// a reikšmė yra 4* b = 3; *// b reikšmė yra pakeičiama į 3* a = b \* 2; *// a reikšmė irgi pasikeičia ir yra lygi 6* |
| --- |

Kaip matome pavyzdyje, antra ir ketvirta eilutės turi lygiai tą pačia išraišką, tačiau skirtingas reikšmės, todėl, kad galutinės reikšmės priklauso nuo b kintamojo reikšmės, ir jei b kintamojo reikšmė yra pakeičiama, tuomet ir a reikšmė keičiasi.

Todėl čia yra labai svarbu atskirti du konceptus, programatinė išraišką ir tos išraiškos sugeneruotą reikšmė. Kintamųjų ir funkcijų pagalba, mes galime turėti tas pačias išraiškas, bet keičiant kintamųjų reikšmes išgauti skirtingas reikšmes dar kitaip vadinant skirtinga rezultatą.

Funkcijos pavyzdys:

| function sum(a, b) {  return a + b; }  sum(1, 5); *// Funkcijos Rezultatas 6* sum(2, 10); *// Funkcijos Rezultatas 12* sum(1,1); *// Funkcijos Rezultatas 2* |
| --- |

## **Paleisti koda - Code run/execution**

Kaip aprašytos grupės programavimo sakinių nurodo kompiuteriui ką daryti?

Tam programa turi būti paleistos (run). Sakinys kaip šis:

| const a = b \* 2; |
| --- |

yra suprantamas programuotojui, tačiau tiesiogiai kompiuteris to nesupranta. Tam jam reikalingas vertėjas (interpreter). Tai yra speciali programa/aplinka, kuri išverčia mūsų parašytos kalbos kodą, į kompiuteriui suprantamas komandas. Todėl tokios kalbos kaip Javascript yra vadinamos interpretuojamos kalbos.

Kai Kurios programavimo kalbos, yra kompiliuojamos, kas reiškia, kad jos pirmiausia turi būti išverčiamos į kompiuteriui suprantama kodą ir tik po to gali būti paleista (run/executed), kad pradėtų vykdyti komandas.

Javascript interpretavimas vyksta realiuoju laiku ir tai daro kaskart kai skaito kodą (dažniausiai tai daro interneto naršyklė). Kaip pavyzdys, kiekviena karta perkraunant internetinį puslapį. Kai puslapis perkraunamas, naršyklė nuskaito jūsų parašytą Javascript koda ir vykdo jam nurodytas komandas.

## **Konsolė - Console**

Javascript kodas gali būti leidžiamas (run) keliais būdais. Vienas būdas, kuomet mes nurodome javascript faila HTML dokumente. Tuomet Interneto naršyklė skaitydamas HTML kodą, tuo pačiu metu paleidžia (execution) ir Javascript kodą. Javascript kodas yra nuskaitomas ir vykdomos ten parašytos komandos.

Tačiau, javascript koda galime leisti ir nebūtinai iš failo, tai galime padaryti Javascript konsoleje. Kurią atsidaryti galite paspaudus dešinį pelės klavišą ant atidaryto puslapio Chrome naršyklėje, pasirinkę inspect. Ši veiksmas atidaro taip vadinamus programuotojo įrankius (developer tools). Iš šios vietos galime labai daug sužinoti apie kodą, ir gauti daug informacijos apie šiuo metu atidarytą puslapį. Viena iš tokių dev tool funkcijų yra konsolė (console). Pasirinkite console tab’a ir čia galite rašyti visą Javascript kodą, kurį suvedus ir paspaudus enter jis bus iš karto paleidžiamas.

## **Console.log()**

console.log() yra funkcija, kurios pagalba, mes galima išspausdintį gauta reikšmę į konsolę.

| console.log('Laba Diena'); *// Konsole išspausdins "Laba Diena"* console.log(1); *// Konsole išspausdins 1* console.log([]); *// Konsole išspausdins []* console.log(2 \* 2); *// Konsole išspausdins 4* console.log('Mano vardas', '{vietoj riestinių skliaustų įrašykite savo varda}'); |
| --- |

Į skliaustelius mes galime įdėti išraišką arba reikšmę, ir ji bus išspausdinti konsolėje. Noriu atkreipti dėmesį, kad spausdinama bus reikšmė o ne pati išraiška. Tai jei išraiška yra 2 \* 2, tai konsolėje išspausdins reikšmę/rezultatą lygų 4.

Šią funkciją mes naudosime per visą mūsų kursą, tam kad išspausdinti reikšmes ir jas galėtume matyti, kitu atveju, javascript interpretatorius, atliks visus veiksmus ir rezultatus laikys tik kompiuterio darbinėje atmintyje, kurios pažiūrėti mes negalime. Tai yra pagrindinis įrankis, kad galėtume patikrinti ar mūsų norimos reikšmės yra teisingos ar mūsų nurodytos komandos atlieka teisingus veiksmus ir neįvėlėme klaidos.

Konsolėje taip pat yra spausdinamos Javascript interpretavimo klaidos, jei naršyklė negalėjo nuskaityti jūsų parašyto kodo, dėl padarytos sintaksės ar kt. klaidos, ši klaida su trumpu aprašymu bus išspausdinta konsolėje.

Todėl tai yra labai galingas ir svarbus įrankis programuotojui, ir jį reikia gerai suprasti ir mokėti skaityti bei pasinaudoti.