

JAVA SCRIPT TILINING IFODALARI VA AMALLARI

Mamatqodirov Maxammadali Mamatisakovich

Farg'ona davlat universiteti axborot texnologiyalari kafedrası

katta o'qituvchisi maxam.uz.1976@gmail.com

Mo'sinjonova Aygiza Husanovna

Farg'ona davlat universiteti 3-kurs talabasi

aygizatessa@gmail.com

Annotatsiya: Mazkur maqola JavaScript dasturlash tilidagi ifodalar va amallarni o'rganishga bag'ishlangan. Maqolada ifodalar va amallarni aniqlash, ularning turlari va qanday ishlashi haqida batafsil tushuntirishlar berilgan. JavaScriptdagi arifmetik, mantiqiy, taqqoslash va bitwise amallari yoritilgan. Shuningdek, ifodalar va amallarni birlashtirish orqali qanday murakkab operatsiyalarni bajarish mumkinligi ko'rsatilgan. Ushbu maqola, web dasturchilar uchun JavaScript tilining asosiy tushunchalarini o'rganishni osonlashtirishni maqsad qilgan.

Kalit So'zlar: JavaScript, AJAX, arrow functions, let, var, const, ifodalar, amallar, arifmetik amallar, mantiqiy amallar, taqqoslash amallari, bitwise amallar, web dasturlash, o'zgaruvchilar.

Annotatsiya (Rus tilida): Данная статья посвящена изучению выражений и операторов в языке программирования JavaScript. В статье подробно рассматриваются различные виды выражений и операций, их синтаксис и принципы работы. Операции делятся на арифметические, логические, операторы сравнения и битовые операторы. Также объясняется, как можно комбинировать выражения и операторы для выполнения сложных вычислений и логических операций. Этот материал предназначен для начинающих разработчиков, чтобы помочь им лучше понять основные концепции языка JavaScript и эффективно использовать их при разработке веб-приложений.

Ключевые слова: *JavaScript, AJAX, стрелочные функции, let, var, const, выражения, операции, арифметические операции, логические операции, операции сравнения, побитовые операции, веб-программирование, переменные.*

Annotatsiya (Ingliz tilida): *This article is dedicated to studying expressions and operators in the JavaScript programming language. It provides an in-depth look at the different types of expressions and operations, their syntax, and how they work. The operations are categorized into arithmetic, logical, comparison, and bitwise operators. The article also explains how expressions and operators can be combined to perform complex calculations and logical operations. This material is aimed at beginner developers to help them better understand the core concepts of JavaScript and apply them effectively in web development.*

Keywords: *JavaScript, AJAX, arrow functions, let, var, const, expressions, operations, arithmetic operations, logical operations, comparison operations, bitwise operations, web programming, variables.*

Kirish:

JavaScript — bu web dasturlashning asosi bo‘lgan, keng tarqalgan va kuchli dasturlash tilidir. Bugungi kunda JavaScript yordamida web sahifalarining interaktivligi va dinamikligi ta'minlanadi. JavaScript tilida turli xil ifodalar va amallar yordamida murakkab hisob-kitoblar, mantiqiy va arifmetik operatsiyalarni amalga oshirish mumkin. Ushbu maqolada JavaScript tilining ifodalari va amallari haqida batafsil ma'lumot beriladi. Maqsad, dasturchilarga bu tilning asosiy konstruktsiyalarini tushunishga va ularni web dasturlar yaratishda samarali ishlatishga yordam berishdir. Bu maqolada JavaScript tilining ifodalari va amallariga oid asosiy tushunchalar, sintaksis va ularning web dasturlashdagi o‘rni haqida so‘z boradi.

JavaScript tilining tarixi va kelib chiqarilishi.

JavaScript, bugungi kunda web dasturlashning asosiy va eng mashhur tillaridan biri bo'lib, butun dunyo bo'ylab veb-sahifalarni dinamik qilish, interaktivligini ta'minlash, va foydalanuvchi tajribasini yaxshilash uchun keng qo'llaniladi. Ushbu dasturlash tili, dastlabgina veb-sahifalariga interaktiv elementlar qo'shish maqsadida yaratilgan bo'lsa-da, hozirda server tarafida ham ishlatiladigan keng qamrovli tildan biriga aylangan.

1. JavaScript yaratilish tarixi.

JavaScript 1995-yilning dekabr oyida, Netscape Communications kompaniyasining dasturchisi **Brendan Eich** tomonidan yaratilgan. Dastlab JavaScript "Mocha" deb nomlangan, keyinroq esa "LiveScript" deb o'zgartirilgan va nihoyat **JavaScript** deb nomlangan. JavaScriptning yaratilishiga asosiy sabab — o'sha davrda veb-sahifalar statik edi, ya'ni foydalanuvchi faqat sahifani ko'rishi mumkin edi, lekin uning bilan interaktiv aloqada bo'lishi yoki o'zgartirishlar kiritishi imkonsiz edi.

2. JavaScriptning dastlabki versiyasi.

JavaScriptning birinchi versiyasi Netscape Navigator 2.0 brauzeri bilan birga chiqarildi. Bu versiya juda oddiy edi va faqatgina foydalanuvchi bilan sahifa o'rtasida ba'zi oddiy interaktivlikni ta'minladi, masalan, tugmalarni bosish orqali formalarni yuborish. JavaScript dastlab faqat brauzerda ishlaydigan skript tili sifatida ishlab chiqilgan.

3. JavaScriptning rivojlanishi.

JavaScriptning rivojlanishi bir necha bosqichlardan o'tdi:

- **ECMAScript 3 (ES3)** (1999-yil) — Bu versiya JavaScriptni asosiy imkoniyatlar bilan ta'minladi va katta o'zgarishlar kiritilmadi.
- **ECMAScript 5 (ES5)** (2009-yil) — Bu versiya ko'plab yangi xususiyatlar kiritdi, masalan, "strict mode" (katta ehtiyotkorlikda kod yozish rejimi), JSON (JavaScript Object Notation) qo'llab-quvvatlashi, yangi metodlar va funksiyalar.
- **ECMAScript 6 (ES6)** yoki **ES2015** (2015-yil) — JavaScript uchun eng katta yangilanishlardan biri bo'ldi. ES6 yangi sintaksis va imkoniyatlar

kirildi, jumladan, **let** va **const** o'zgaruvchilarni, **arrow functions** (funksiya sintaksisi), **Promises**, **Modules** va boshqa ko'plab yangi xususiyatlar.

- **ES7 (ES2016)**, **ES8 (ES2017)** va undan keyingi versiyalar (ES2018, ES2019, va hokazo) JavaScriptni yanada kuchaytirdi va yangi imkoniyatlar taqdim etdi, masalan, **async/await**, **Object.entries()** va **Object.values()** kabi metodlar, **Array.includes()** va boshqalar.

4. JavaScriptning hozirgi holati.

Bugungi kunda JavaScript butunlay rivojlangan va ko'plab imkoniyatlarga ega tildan biriga aylangan. Uning ishlatilishi nafaqat brauzerda, balki **Node.js** kabi server tarafida ham keng tarqalgan. JavaScript ekotizimi juda kengaygan, ko'plab kutubxonalar va ramkalar (frameworks) ishlab chiqilgan. Masalan, **React**, **Angular**, **Vue.js** kabi mashhur frontend ramkalar, va **Express.js** kabi backend ramkalar JavaScriptning imkoniyatlarini yanada oshirgan.

5. JavaScriptning kelajagi.

JavaScriptning kelajagi juda yorqin ko'rinadi. Yangi imkoniyatlar, tizimlar va texnologiyalar ishlab chiqilishi bilan JavaScript dasturlash tili yanada kuchayib, o'zining dominanti saqlaydi. Web dasturlash va mobil ilovalar yaratish uchun eng mashhur va zarur tildan biri sifatida JavaScriptni o'rganish va undan foydalanish hozirgi va kelajakdagi dasturchilar uchun juda muhim.

Asosiy qism:

JavaScript tilining ifodalari va amallari.

JavaScriptda **ifodalar** (expressions) dasturchilarga turli ma'lumotlarni ishlash va manipulyatsiya qilish imkonini beradi. Ifoda — bu qiymatni qaytaradigan va ko'pincha boshqa ifodalar bilan kombinatsiyalanuvchi kod qismini anglatadi. Ifodalarning to'liq ro'yxati — arifmetik, mantiqiy, taqqoslash, bitli va boshqa turlarni o'z ichiga oladi. Har bir ifoda dasturdagi turli vazifalarni bajarishda foydalaniladi. JavaScriptda ifodalarning ahamiyati shundaki, ular yordamida dasturchilar o'zgaruvchilarni o'zgartirish, qiymatlarni taqqoslash, funksiyalarni chaqirish va boshqa amallarni bajarish imkoniga ega bo'ladilar.

JavaScript operatorlari va ularning turlari.

JavaScriptda operatorlar — bu ifodalar va amallarni bajarishga xizmat qiladigan maxsus belgilar yoki kalit soʻzlar. Operatorlar yordamida turli xil amallarni bajarish mumkin, masalan, *arifmetik*, *mantiqiy*, *taqqoslash* va *bitli* operatorlar. Har bir operator turli xil vazifalarni bajaradi:

- **Arifmetik operatorlar** yordamida sonli ma'lumotlar ustida amallar bajariladi (masalan, qoʻshish, ayirish, koʻpaytirish, boʻlish).
- **Mantiqiy operatorlar** (AND, OR, NOT) esa, koʻpincha shartli ifodalar bilan birga ishlatiladi va mantiqiy qiymatlarni taqqoslashga yordam beradi.
- **Taqqoslash operatorlari** (masalan, ==, ===, <, >) yordamida qiymatlarning tengligini yoki kattaligini tekshirish mumkin.
- **Bitli operatorlar** esa, bitli ma'lumotlar bilan ishlashga imkon beradi va odatda past darajadagi dasturlashda ishlatiladi.

Operatorlar JavaScriptda ifodalarning ishlashini aniqlaydi va dasturchilarga kodni aniq va samarali yaratishda yordam beradi.

JavaScriptda *oʻzgaruvchilar* dasturlashning muhim qismidir. Ular ma'lumotni saqlash va unga murojaat qilish uchun ishlatiladi. Oʻzgaruvchilar turli turlarga ega boʻlishi mumkin: *string*, *number*, *boolean*, *object*, va boshqalar. JavaScriptda oʻzgaruvchilarni eʼlon qilish uchun var, let, yoki const kalit soʻzlari ishlatiladi.

Oʻzgaruvchilar ifodalar va amallarda ishlatiladi va ular yordamida dasturdagi barcha hisob-kitoblar, taqqoslashlar va boshqa manipulyatsiyalar amalga oshiriladi.

Shartli operatorlar va tsikllar.

JavaScriptda *shartli operatorlar* va *tsikllar* ifodalar va amallarni boshqarish uchun ishlatiladi. Shartli operatorlar yordamida dasturda turli shartlarni tekshirish va unga mos ravishda harakat qilish mumkin (masalan, if, else if, switch operatorlari). *Tsikllar* esa bir necha marta biror amalni bajarish uchun ishlatiladi, masalan, for, while, do...while tsikllari yordamida kodni takrorlash mumkin.

Shartli operatorlar va tsikllar kodni moslashtirish va avtomatizatsiya qilishda juda muhimdir, chunki ular dasturdagi qarorlar va algoritmlarni optimallashtirishga yordam beradi.

JavaScriptda ifodalar va amallar dasturlashning asosiy elementlaridan biridir. Ular yordamida qiymatlarni manipulyatsiya qilish, mantiqiy va arifmetik operatsiyalarni bajarish mumkin. JavaScriptning bu asosiy elementlari web dasturlarni yaratishda muhim o‘rin tutadi, chunki ular sahifadagi dinamik o‘zgarishlarni va foydalanuvchi bilan o‘zaro aloqani ta‘minlashga yordam beradi.

JavaScript Tilining Afzalliklari va Kamchiliklari.

JavaScript — bugungi kunda web dasturlashda eng mashhur va kuchli til bo‘lib, uning ko‘plab afzalliklari mavjud. Ammo, har bir texnologiya kabi, JavaScriptning ham o‘ziga xos kamchiliklari bor. Quyida JavaScript tilining afzalliklari va kamchiliklari haqida batafsil ma'lumot beraman.

JavaScript tilining afzalliklari.

1. Brauzerda ishlash (Client-Side) JavaScriptning eng katta afzalliklaridan biri shundaki, u brauzerda ishlaydi, ya'ni foydalanuvchining qurilmasida, serverga bog‘lanmasdan. Bu esa veb-sahifalar va ilovalarni tezroq va interaktivroq qiladi. JavaScript yordamida sahifa yuklanishidan so‘ng sahifadagi kontentni va interaktiv elementlarni dinamik tarzda o‘zgartirish mumkin.

2. Keng qo‘llaniluvchi (Uchinchi tomon kutubxonalar) JavaScriptning eng katta afzalliklaridan biri — uning juda keng kutubxonalar va ramkalar ekotizimiga ega bo‘lishidir. Masalan, **React**, **Angular**, **Vue.js** kabi ramkalar va **jQuery**, **D3.js**, **Three.js** kabi kutubxonalar web dasturlarni ishlab chiqishni juda osonlashtiradi va tezlashtiradi. JavaScriptning ushbu kutubxona va ramkalari murakkab veb-ilovalarni yaratishda juda foydalidir.

3. Interaktivlik va dinamik sahifalar JavaScript web sahifalariga interaktivlik qo‘shish imkoniyatini beradi. Foydalanuvchi bilan o‘zaro aloqada bo‘lish uchun, masalan, sahifani yangilamasdan dinamik tarzda kontentni o‘zgartirish (AJAX yordamida) mumkin.

4.Oson o'rganish va keng qo'llanishi JavaScriptning sintaksisi boshqa dasturlash tillariga qaraganda o'rganish uchun nisbatan oson va sodda. Shuningdek, JavaScript barcha asosiy brauzerlarda ishlaydi, bu esa uning keng tarqalganligi va qo'llanilishini ta'minlaydi. Yangi boshlovchilar uchun juda yaxshi dasturlash tilidir.

5.Dinamik Typing JavaScriptda o'zgaruvchilar dinamik ravishda, ya'ni ularning turi bajarilayotgan amallarga qarab aniqlanadi. Bu dasturlashni tezlashtiradi, ammo ba'zi hollarda xatoliklarni aniqlashni qiyinlashtirishi mumkin.

JavaScript tilining kamchiliklari.

1.Dasturlashdagi murakkabliklar JavaScriptning dinamik tipi va ba'zi sintaksis xususiyatlari dasturchilar uchun noaniqlik yaratishi mumkin. Masalan, o'zgaruvchilarni deklaratsiya qilishda yoki qiymatlarni taqqoslashda kutulmagan xatoliklar yuzaga kelishi mumkin. == va === operatorlarining farqi kabi kichik noaniqliklar dasturchilarning to'g'ri kod yozishlarini qiyinlashtirishi mumkin.

2.Kross-Brauzer muammolar JavaScriptning birinchi versiyalarida brauzerlar o'rtasida to'liq moslik yo'q edi. Garchi hozirda bu muammo katta darajada hal etilgan bo'lsa-da, ba'zi eski brauzerlar (ayniqsa, Internet Explorer) hali ham JavaScriptning ba'zi xususiyatlarini to'liq qo'llab-quvvatlamaydi. Shuningdek, turli brauzerlar JavaScriptni turlicha ishlatishi va qo'llab-quvvatlash darajasi har xil bo'lishi mumkin.

3.Xavfsizlik muammolar JavaScript ba'zan xavfsizlikka tahdid soladigan skriptlar yaratishda ishlatilishi mumkin. **Cross-Site Scripting (XSS)** kabi hujumlar, JavaScriptning web sahifalarga qo'shilishi orqali foydalanuvchilarni zararli skriptlar bilan aldaydi. Bu muammoni bartaraf etish uchun JavaScriptni xavfsiz ishlatish va to'g'ri sanitatsiya qilish juda muhimdir.

4.Notog'ri asinxronizatsiya (Callback Hell) JavaScriptda asinxron dasturlash, masalan, **AJAX** so'rovlari yoki **Promises** yordamida amalga oshiriladi. Ammo ko'p hollarda bir-biriga bog'langan asinxron funksiyalarni yozish qiyinlashadi. "Callback hell" (callbacaklar uyumi) deb ataladigan

muammo shundan kelib chiqadi, ya'ni har xil callback funksiyalarining ichma-ich o'rnatilishi kodni o'qish va tushunishni qiyinlashtiradi.

5. Ishlashning tezligi JavaScript, boshqa tillar (masalan, C++ yoki Java) bilan taqqoslaganda, ba'zi operatsiyalarni bajarishda sekinroq ishlashi mumkin. JavaScript dinamik tildir, va bu turli xil xatoliklar va samaradorlikning pastligi bilan bog'liq bo'lishi mumkin. Xususan, kattaroq va murakkabroq hisoblashlarda JavaScriptning ishlash tezligi pasayishi mumkin.

Funksiyalar va ularning roli.

Funksiyalar JavaScriptning asosiy komponentlaridan biri bo'lib, ular kodni modularlashtirish va qayta ishlatish imkonini beradi. Funksiyalar yordamida bir xil amallarni turli joylarda qayta ishlatish mumkin, bu esa dastur kodining samaradorligini oshiradi. JavaScriptda funksiyalarni aniqlash va chaqirish juda oson. Funksiya parametrlarini olish, qiymat qaytarish va boshqa amallarni bajarish orqali dasturdagi murakkab operatsiyalarni soddalashtirish mumkin.

Asinxron dasturlash va Callback funksiyalari. JavaScriptda **asinxron dasturlash** imkoniyatlari ham juda muhimdir. Asinxron kod yordamida JavaScriptda uzoq davom etadigan operatsiyalarni, masalan, ma'lumotlar bazasiga so'rov yuborish yoki faylni yuklashni boshqarish mumkin. Bu operatsiyalarni bajarishda **callback funksiyalari** ishlatiladi. Callback funksiyalar kodni yozishdagi murakkabliklarni bartaraf qilishga yordam beradi, ammo ular ba'zida "callback hell" muammosiga olib kelishi mumkin, bu esa kodni o'qishni va tushunishni qiyinlashtiradi.

Xulosa:

JavaScript dasturlash tili bugungi kunda web-dasturlashning eng muhim vositalaridan biri hisoblanadi. Ushbu maqolada JavaScript tilining ifodalari va amallari, ya'ni uning sintaksisi, turlari, operatorlari, va ularning qanday ishlashi haqida batafsil ma'lumot berildi. JavaScript tilida ifodalarning va amallarning o'rni juda katta bo'lib, ular dasturchilarni web-illovalar yaratishda kuchli vositalar bilan ta'minlaydi.

JavaScript tilining ifodalari va amallari dasturlashda muhim rol o'ynaydi. Ifodalar yordamida dasturchilar turli ma'lumotlar ustida ishlaydi, operatorlar orqali ma'lumotlarni manipulyatsiya qiladi, shartli operatorlar va tsikllar yordamida dasturdagi qarorlar qabul qilinadi. Funktsiyalar va asinxron dasturlash imkoniyatlari esa dastur kodini soddalashtirish va optimallashtirishda yordam beradi. JavaScriptning ifodalari va amallari web dasturlashda samarali va dinamik ilovalar yaratishda asosiy vositalardan biri bo'lib, ularning to'g'ri qo'llanilishi dasturchilar uchun muhim ahamiyatga ega.

JavaScript 1995-yilda yaratilganidan beri web dasturlash sohasidagi eng kuchli va ommabop tilga aylangan. Uning rivojlanishi va standartlashtirilishi esa nafaqat web dasturlashni osonlashtirdi, balki JavaScriptni har xil platformalarda ishlatishga imkon berdi. Hozirgi kunda JavaScript butun dunyoda ishlatiladigan eng muhim dasturlash tillaridan biri bo'lib qolmoqda.

FOYDALANILGAN ADABIYOTLAR:

1. **Flanagan, D.** (2020). *JavaScript: The Definitive Guide* (7th Edition). O'Reilly Media.
2. **MDN Web Docs (Mozilla Developer Network).** (2024). *JavaScript Guide*. <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>
3. **Eich, B.** (2015). *The Evolution of JavaScript*. Communications of the ACM, 58(10), 56-63.
4. **ECMA International.** (2024). *ECMAScript® 2023 Language Specification*. <https://www.ecma-international.org/publications-and-standards/standards/ecma-262/>
5. **Zakas, N. C.** (2012). *JavaScript Patterns: Build Better Applications with Coding and Design Patterns*. O'Reilly Media.
6. **Flanagan, D.** (2006). *JavaScript: The Good Parts*. O'Reilly Media.
7. **W3Schools.** (2024). *JavaScript Tutorial*. <https://www.w3schools.com/js/>