

**JAVASCRIPT FUNKSIYALARI VA ULARNING TURLARI**

Farg'ona davlat universiteti axborot texnologiyalari kafedrası katta o'qituvchisi

maxam.uz.1976@gmail.com

Mamatqodirov Mahammadali

Farg'ona davlat universiteti 3-kurs talabasi

mamurovaroxatoy79@gmail.com

Ma'murova Rohatoy Ilhomjon qizi

Annotatsiya: JavaScript dasturlash tili veb-sahifalarni interaktiv va dinamik qilish uchun keng qo'llaniladi. Bu tilda funksiyalar dasturiy ta'minotning eng muhim elementlaridan biri hisoblanadi. Funksiyalar kodni qayta ishlatish, modullashtirish va samarali boshqarish imkoniyatlarini yaratadi. Ushbu ishda JavaScript funksiyalarining turlari, jumladan, an'anaviy funksiyalar, ifodaviy funksiyalar, arrow funksiyalar, anonim funksiyalar va rekursiv funksiyalar tahlil qilinadi. Har bir funksiyaning afzalliklari va foydalanish sohalari keltiriladi. Shuningdek, funksiyalar bilan ishlashda parametrlar, default parametrlar, rest parametrlar va spread operatorining roli muhokama qilinadi. Ushbu tezis JavaScript funksiyalarini o'rganish orqali dasturlashda samarali yondashuvlarni taqdim etadi.

Kalit so'zlar: JavaScript, Funksiyalar, An'anaviy funksiyalar, Arrow funksiyalar, Rekursiv funksiyalar, Parametrlar, Rest parametrlar, Default parametrlar, Spread operatori, Dasturlash.

Аннотация: Язык программирования JavaScript активно используется в разработке динамических веб-приложений. Одним из важнейших компонентов этого языка являются функции, которые обеспечивают модульность кода, его повторное использование и упрощение разработки. В данной работе рассматриваются различные типы функций JavaScript, такие как традиционные функции, функции-выражения, стрелочные функции, анонимные функции и рекурсивные функции. Также анализируются параметры функций, значения по умолчанию, параметры rest и spread, которые играют важную роль в улучшении



гибкости и эффективности кода. Работа помогает глубже понять, как правильно использовать функции в JavaScript для создания более читаемых и эффективных программ.

Ключевые слова: JavaScript, Функции, Традиционные функции, Стрелочные функции, Рекурсивные функции, Параметры, Параметры rest, Значения по умолчанию, Spread оператор, Программирование.

Annotation: JavaScript is a widely used programming language in web development, essential for creating dynamic and interactive web applications. Functions in JavaScript are one of the core components that allow modularity, code reuse, and efficient management of complex tasks. This thesis examines various types of functions in JavaScript, including traditional functions, function expressions, arrow functions, anonymous functions, and recursive functions. The paper also explores the use of parameters, default parameters, rest parameters, and the spread operator, which play a crucial role in enhancing the flexibility and efficiency of code. The work provides valuable insights into how functions can be effectively utilized in JavaScript to improve code quality and maintainability.

Keywords: JavaScript, Functions, Traditional functions, Arrow functions, Recursive functions, Parameters, Rest parameters, Default parameters, Spread operator, Programming.

Kirish: JavaScript dasturlash tilida funksiyalar dastur kodining asosiy qismlaridan biridir. Ular kodni modullashtirish, takrorlanadigan vazifalarni bajarish va umumiy sintaksisni soddalashtirishga yordam beradi. Funksiyalar kodni qayta ishlatish va o'qilishi oson qilib tashkil etishda muhim rol o'ynaydi.

Funksiya — bu ma'lum bir vazifani bajaradigan kod blokidir. Funksiyalar yordamida kodni modullashtirish mumkin, shuningdek, ularni qayta ishlatish va dastur samaradorligini oshirish mumkin.

Funksiya sintaksisi:

```
function funksiyaNomlari(parametr1, parametr2) {  
  // kodlar
```

```
return natija; // funksiyaning natijasi (ixtiyoriy)
}
```

Funksiya chaqirilganda, argumentlar (parametrlar) beriladi va funksiyaning ichidagi kod bajariladi, natijada return orqali qiymat qaytadi.

1. An'anaviy funksiyalar (Function Declarations)

An'anaviy funksiyalar JavaScriptning eng oddiy va keng tarqalgan funksiyalaridir. Ular function kalit so'zi bilan e'lon qilinadi va o'z nomlariga ega.

Misol:

```
function yigindi(a, b) {
  return a + b;}
```

Bu yerda yigindi nomli funksiya ikki parametрни qabul qiladi (a va b), ularni qo'shib, natijani qaytaradi.

2. Ifodaviy funksiyalar (Function Expressions)

Ifodaviy funksiyalar o'zgaruvchilarga tayinlanadi va ularni faqat o'sha o'zgaruvchi orqali chaqirish mumkin. Bunday funksiyalar odatda anonim bo'ladi.

Misol:

```
const ko'paytir = function(x, y) {
  return x * y;
};
```

Bunday funksiyalarni faqat o'zgaruvchi orqali chaqirish mumkin, chunki ular funksiya ifodasini o'z ichiga oladi.

3. Arrow funksiyalar (ES6 arrow functions)

Arrow funksiyalari JavaScriptning yangi sintaksisi bo'lib, ular qisqa va oson yoziladi. ES6 versiyasidan kiritilgan bu funksiyalar o'ziga xos sintaksisga ega va ko'proq funksiyalarga ehtiyoj seziladigan holatlarda ishlatiladi.

Misol:

```
const salom = (ism) => `Salom, ${ism}!`;
```

Yuqoridagi misolda salom funksiyasi qisqa sintaksisda yozilgan va faqat bitta parametрни oladi. Arrow funksiyalar faqat bitta satrda bajarilsa, ularning return operatorini yozish shart emas.



4. Anonim funksiyalar (*Anonymous Functions*)

Anonim funksiyalar nomsiz bo'lib, ko'pincha funksiyalarni argument sifatida uzatishda yoki callback sifatida ishlatiladi.

Misol:

```
setTimeout(function() {  
    console.log("3 soniyadan so'ng chiqar");  
}, 3000);
```

Bu yerda anonim funksiya setTimeout metodiga callback sifatida uzatilgan. Funksiya nomini kiritishning hojati yo'q, chunki uni faqat bitta joyda ishlatish kerak.

5. Rekursiv funksiyalar (*Recursive Functions*)

Rekursiv funksiya o'zini o'zi chaqiradigan funksiyadir. Bu turdagi funksiyalar ko'pincha murakkab vazifalarni, masalan, faktorial yoki Fibbonachchi sonlari kabi hisoblashlarda ishlatiladi.

Misol:

```
function faktorial(n) {  
    if (n === 0) return 1;  
    return n * faktorial(n - 1);  
}
```

Bu funksiya o'zini o'zi chaqiradi va n qiymati nolga teng bo'lguncha bajariladi. Bu usul, masalan, faktorial kabi hisoblashlarda samarali bo'ladi.

Funksiyalar bilan ishlash imkoniyatlari:

JavaScriptda funksiyalar bilan ishlashda ko'plab imkoniyatlar mavjud, ular quyidagilarni o'z ichiga oladi:

1. Parametrlar va return qiymatlar

Funksiyalarni yaratishda parametrlar yordamida ma'lumotlar uzatiladi. Shuningdek, funksiyalar biror natija qaytarishi mumkin. Parametrlar yordamida funksiyalarni moslashtirish va maxsus qiymatlar bilan ishlash mumkin.

2. Default parametrlar

Funksiya parametrlariga default qiymatlar belgilash imkoniyati JavaScript ES6da kiritildi. Agar funksiya chaqirilganida parametrlar berilmasa, default qiymatlar ishlatiladi.

**Misol:**

```
function salom(ism = "Dunyo") {  
  return `Salom, ${ism}!`;}
```

Agar salom() funksiyasi chaqirilsa, u "Dunyo" deb qaytaradi.

3. Rest parametrlar

Rest parametrlar yordamida funksiyaga ko'p miqdordagi argumentlarni uzatish mumkin. Bu parametrlar ... belgisidan foydalaniladi.

Misol:

```
function qo'shish(...sonlar) {  
  return sonlar.reduce((a, b) => a + b, 0);}
```

Bu funksiyada ...sonlar barcha argumentlarni array shaklida oladi va ular ustida operatsiyalar bajariladi.

4. Spread operatori

Spread operatori (...) funksiya argumentlarini individual qiymatlar sifatida ajratish uchun ishlatiladi. Bu operator yordamida array yoki objectni funksiya parametrlariga uzatish mumkin.

Misol:

```
const sonlar = [1, 2, 3];  
console.log(qo'shish(...sonlar)); // 6
```

Xulosa:

JavaScript funksiyalari dasturlashda juda muhim vositalardan biri bo'lib, ularning samarali ishlatilishi dastur kodining sifatini va unumdorligini sezilarli darajada oshiradi. Funksiyalar kodni modullashtirish, qayta ishlatish, va uni o'qilishi oson qilib tashkil etish imkoniyatlarini taqdim etadi. JavaScriptda funksiyalar turli xil shakllarda bo'ladi, har bir turi o'zining afzalliklari va qo'llanilish joylariga ega.

An'anaviy funksiyalar (function declarations) juda oddiy va aniq sintaksisga ega, ularning ishlash tartibi dasturlashda eng ko'p uchraydigan va eng keng qo'llaniladigan funksiyalardir. Ifodaviy funksiyalar (function expressions) esa ayniqsa o'zgaruvchilar bilan ishlashda yoki funksiyalarni argument sifatida uzatishda qulay. Arrow funksiyalar esa ES6'dan boshlab ishlab chiqilgan bo'lib, ularning sintaksisi



qisqaroq bo'lib, ayniqsa bir satrli funksiyalar uchun juda qulay. Bu funksiyalarni o'z ichiga olgan kodlar ancha o'qilishi oson va samarali bo'ladi.

Anonim funksiyalar va rekursiv funksiyalar esa maxsus vaziyatlar uchun ishlatiladi. Anonim funksiyalar ko'pincha callback funksiyalar sifatida qo'llaniladi va dasturlashda noaniq vaziyatlar uchun juda foydali bo'ladi. Rekursiv funksiyalar esa o'zini o'zi chaqirish xususiyatiga ega bo'lib, ular murakkab hisob-kitoblar, masalan, faktorial yoki fibonachchi sonlarini hisoblashda ishlatiladi.

JavaScriptda funksiyalarni ishlatishda parametrlar va return qiymatlarining o'rnini katta. Funksiyalar parametrlarni qabul qilib, muayyan bir ishini bajarib, natijaviy qiymat qaytarishi mumkin. Shuningdek, default parametrlar, rest parametrlar, va spread operatori kabi imkoniyatlar funksiyalarni yanada moslashuvchan va kuchli qiladi. Bu imkoniyatlar kodni qisqartirish va dastur samaradorligini oshirishga yordam beradi.

Shu bilan birga, JavaScriptda funksiyalar o'zgaruvchilar kabi "birinchi darajali fuqarolar" sifatida qoraladi, ya'ni ular boshqa funksiyalarga uzatilishi yoki o'zgaruvchilarga tayinlanishi mumkin. Bu holat dasturlashda yuqori darajadagi abstraktsiyalarni yaratishga va kodni qayta ishlatish imkoniyatlarini kengaytirishga yordam beradi.

Natijada, JavaScript funksiyalari dasturlashdagi eng muhim komponentlardan biri bo'lib, ularning turlari va imkoniyatlarini tushunish, samarali va moslashuvchan kod yozishda katta yordam beradi. Har bir dasturchi, xususan, veb dasturchilar, JavaScript funksiyalarini puxta o'rganish va ularni to'g'ri ishlatish orqali o'z kodining sifatini va samaradorligini oshirishi mumkin.

FOYDALANILGAN ADABIYOTLAR RO'YHATI:

- 1) "Машинное обучение: практическое руководство" (А. Жеребцов);
- 2) "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow" (Aurélien Géron);
- 3) "Введение в машинное обучение" (Эндрю Нг);
- 4) "Deep Learning" (Ian Goodfellow, Yoshua Bengio, Aaron Courville).



- 5) Shoaxmedova N.X., Abdullayeva I.N., "Iqtisodiyotda axborot texnologiyalar va tizimlar";
- 6) J.Rustamov., A.Ernazarov., K.Shadiyarova ., N.Tojiyev., J.Umirov., "Iqtisodiyotda axborot kommunikatsion texnologiyalar va tizimlar";
- 7) Jomonqulova F.E., Shadmanov I.E., Iqtisodiyotda axborot-kommunikatsion texnologiyalar va tizimlar ;
- 8) Kenneth C.Loudon , Jane P.Loudon. Management information systems. New York, 2019.