

## **DASTURLASHDA ISTISNOLARNING AHAMIYATI**

***Farmonov Sherzodbek Raxmonjonovich***

*Farg'ona davlat unversiteti amaliy matematika va*

*informatika kafedrası katta o'qituvchisi*

*farmonovsh@gmail.com*

***Mamatova Sarvinoz Mahmudjon qizi***

*Farg'ona davlat unversiteti 2-kurs talabasi*

***Annotatsiya:*** *Ushbu maqola dasturlashda istisnolarning ahamiyatini ta'kidlab, ularning kutilmagan vaziyatlar va xatolarni hal qilishdagi rolga e'tibor qaratadi. U turli jihatlarni, jumladan, istisnolarning ta'rifini, NullPointerException va ArrayIndexOutOfBoundsException kabi turlarini va istisnolarni samarali boshqarish uchun try-catch bloklaridan foydalanish muhimligini qamrab oladi. Matnda istisnolar kodning o'qilishi, modulliligi va mustahkamligiga qanday hissa qo'shishi ta'kidlanadi, bu esa ishlab chiquvchilarga kutilmagan vaziyatlarni oqilona hal qilish imkonini beradi. Shuningdek, u kod xavfsizligini, resurslarni boshqarishni va dasturning turli qismlari o'rtasidagi hamkorlikni yaxshilashdagi istisnolarning rolini ta'kidlaydi. Istisnolardan foydalanishning afzalliklari, jumladan, xatolarni tuzatish, modulli kodlash amaliyoti va yaxshilangan kodni o'qish mumkinligi batafsil tavsiflanadi. Matn FileNotFoundException kabi keng tarqalgan istisnolar misollarini taqdim etadi va ularning ruxsati haqida tushuncha beradi. Umuman olganda, dasturlashda istisnolardan foydalanish ishonchli, xavfsiz va foydalanuvchilarga qulay dasturiy ta'minot ilovalarini yaratish uchun muhim ekanligini ta'kidlaydi.*

***Kalit so'zlar:*** *dasturlashda istisno, xatolarni qayta ishlash, xatolarga chidamlilik, nosozliklarni tuzatish va muammolarni bartaraf etish, xavfsizlik, modullilik, NullPointerException, ArrayIndexOutOfBoundsException, FileNotFoundException, ArithmeticException, try-catch, dasturiy ilovalar.*

**Istisnolar va ularning dasturlashdagi ahamiyati.**

Dasturlashda istisno - bu dasturning normal bajarilishini to'xtatuvchi hodisa. Bu dastur hal qila olmaydigan xato yoki kutilmagan holat yuzaga kelganda yuzaga keladi. Istisno yuzaga kelganda, dastur tugatiladi yoki xato xabari qaytariladi va foydalanuvchi muammo haqida xabardor qilinadi.

Istisnolar turli sabablarga ko'ra yuzaga kelishi mumkin, jumladan apparat xatolari, tarmoqdagi nosozliklar, foydalanuvchi kiritish xatolari va dasturlash xatolar. Ular dasturning kutilmagan vaziyatlarni hal qilish va halokatli nosozliklarni oldini olish usulidir.

Istisno o'rnatilganda, dastur tegishli istisno ishlovchisini qidirib, qo'ng'iroqlar to'plamini yechib olishni boshlaydi. Agar istisno ishlov beruvchi topilsa, dastur o'sha ishlov beruvchiga o'tadi va u yerdan bajarishni davom ettiradi. Agar ishlov beruvchi topilmasa, dastur tugatiladi va xato xabari ko'rsatiladi.

Istisnolardan foydalanib, dasturchilar yanada ishonchli va mustahkam kod yozishlari mumkin. Ular dasturning ishdan chiqishi yoki kutilmagan natijalarga olib kelishi o'rniga, yuzaga kelishi mumkin bo'lgan xatolarni oldindan bilishlari va ularni oqilona hal qilishlari mumkin.

Dasturiy ta'minotni ishlab chiqishda istisnolardan foydalanishning ahamiyati Istisnolar bilan ishlash dasturiy ta'minotni ishlab chiqishning muhim tarkibiy qismidir, chunki u dasturiy ilovalarning barqarorligi va ishonchliligini ta'minlashga yordam beradi. To'g'ri istisnolardan foydalanilmasa, dastur kutilmagan hodisalar ro'y berganda ishdan chiqishi yoki o'zini oldindan aytib bo'lmaydigan tarzda harakat qilishi mumkin, natijada ma'lumotlar yo'qolishi, xavfsizlik zaifligi va foydalanuvchining umidsizlikka tushishi mumkin.

Istisnolarni ushlash va ularga ishlov berish orqali ishlab chiquvchilar dasturning ishdan chiqishini oldini olishlari va foydalanuvchilarga nima noto'g'ri bo'lganini va uni qanday tuzatish kerakligini tushunishga yordam beradigan mazmunli xato xabarlarini taqdim etishlari mumkin. Istisnolarni qayta ishlash, shuningdek, ishlab chiquvchilarga o'z kodlarini yanada samarali disk raskadrovka qilish va sinab ko'rish imkonini beradi, bu esa xatolarni aniqlash va tuzatishni

osonlashtiradi.

Bundan tashqari, yanada mustahkam va moslashuvchan kod yaratish uchun maxsus istisnolardan foydalanish mumkin. Maxsus istisnolarni belgilash orqali ishlab chiquvchilar aniqroq xato xabarlarini taqdim etishlari va o'zlarining kodlariga maxsus xatti-harakatlarni qo'shishlari mumkin, bu esa o'zlarining dasturiy ilovalarining umumiy sifati va qulayligini yaxshilaydi.

Umuman olganda, istisnolardan foydalanishni o'zlashtirish har qanday dasturchi uchun muhim mahoratdir va u dasturiy ilovalarning sifati, ishonchliligi va foydalanuvchi tajribasini yaxshilashga yordam beradi.

Istisnolar bir necha sabablarga ko'ra dasturlash tillarida hal qiluvchi rol o'ynaydi:

**1. Xatolarni qayta ishlash:** Istisnolar dasturni bajarish jarayonida yuzaga kelishi mumkin bo'lgan xatolar va istisno holatlar bilan ishlashning tizimli usulini ta'minlaydi. Bu ishlab chiquvchilarga xatolarni boshqarish mantig'ini asosiy dastur oqimidan ajratish imkonini beradi, bu esa toza va barqaror kodni targ'ib qiladi.

**2. Xatolarga chidamlilik:** Istisnolar dasturni to'xtatmasdan kutilmagan vaziyatlarni oqilona hal qilish mexanizmini taqdim etish orqali dasturlarni xatolarga chidamliroq qilishga yordam beradi. Bu mustahkam va ishonchli dasturiy ta'minot uchun zarurdir.

**3. Nosozliklarni tuzatish va muammolarni bartaraf etish:** Istisnolar sodir bo'lgan xato haqida qimmatli ma'lumotlarni, shu jumladan xatolik turini va u sodir bo'lgan koddagi joyni beradi. Ushbu ma'lumotlardan nosozliklarni tuzatish va muammolarni bartaraf etish uchun foydalanish mumkin, bu esa dasturdagi muammolarni aniqlash va tuzatishni osonlashtiradi.

**4. Boshqaruv oqimi:** Istisnolar chiziqli bo'lmagan boshqaruv oqimiga ruxsat beradi, bu esa ishlab chiquvchilarga muayyan xatolar yuzaga kelganda dastur qanday harakat qilishini aniqlash imkonini beradi. Bu moslashuvchanlik dastur harakatining umumiy mustahkamligi va bashorat qilinishiga yordam beradi.

**5. Resurslarni boshqarish:** Istisnolar, hatto xatolar mavjud bo'lganda ham,

xotirani bo'shatish, fayllarni yopish yoki tarmoq ulanishlarini chiqarish kabi resurslarni to'g'ri tozalashni ta'minlash uchun ishlatilishi mumkin. Bu resurslarning sizib chiqishini oldini olishga yordam beradi va tizim barqarorligini saqlaydi.

**6. Xavfsizlik:** Istisnolarni boshqarish ilova xavfsizligida rol o'ynaydi, masalan, kirishni tekshirishdagi nosozliklar va kutilmagan xatti-harakatlar kabi potentsial xavfsizlik zaifliklarini boshqariladigan va tuzilgan tarzda boshqarish vositalarini taqdim etadi.

**7. Birgalikda ishlash:** Istisnolar dasturning turli qismlari yoki turli dasturiy ta'minot komponentlari o'rtasida o'zaro hamkorlikni osonlashtiradi, bu esa xatolarni tizim chegaralari bo'ylab tarqatish va qayta ishlash imkonini beradi.

**8. Kodning o'qilishi va barqarorligi:** Istisnolarni ko'rib chiqish orqali dasturchilar xatolarni boshqarish mantig'ini oddiy dastur mantig'idan ajratib, kodni yanada o'qilishi va saqlab turishi mumkin. Istisnolar aniqroq va tartibli kodni yaratishga imkon beradi, chunki xatolarni boshqarish kodi markazlashtirilgan va dastur davomida qayta ishlatilishi mumkin.

#### **Dasturlashda istisnolardan foydalanishning afzalliklari:**

Istisnolar dasturlashda muhim rol o'ynaydi, chunki ular kodning umumiy sifati va ishonchliligini oshirishga yordam beradigan bir qator afzalliklarni taklif qiladi. Istisnolardan foydalanishning ba'zi asosiy afzalliklari:

- **Xatolarni qayta ishlash:** Istisnolar dasturni bajarish jarayonida yuzaga kelishi mumkin bo'lgan xatolarni boshqarishning tuzilgan va samarali usulini ta'minlaydi. Try-catch bloklaridan foydalangan holda, ishlab chiquvchilar muayyan istisnolarni qo'lga olishlari va boshqarishlari mumkin, bu xatolarni tashxislash va tuzatishni osonlashtiradi.

- **Modullilik:** Istisnolar kodda modullikni yaxshilash imkonini beradi, chunki ular xatolarni boshqarish mantiqini dastur mantig'ining qolgan qismidan ajratib turadi. Bu kodni o'qish, tushunish va saqlashni osonlashtiradi.

- **Istisnolar zanjiri:** istisnolar bir-biriga bog'lanishi mumkin, bu esa xatolar haqida batafsilroq va ma'lumot beruvchi hisobot tizimini yaratishga imkon beradi. Istisnolarni o'rash va qaytadan tashlash orqali ishlab chiquvchilar xato haqida

ko'proq kontekst va ma'lumotlarni taqdim etishlari mumkin, bu muammoning asosiy sababini aniqlashni osonlashtiradi.

- **Aqli tugatish:** Istisno yuzaga kelganda, bu dasturni to'satdan ishdan chiqishdan ko'ra nozik tarzda tugatish imkonini beradi. Bu resurslarni to'g'ri bo'shatish va tozalashni ta'minlaydi, potentsial xotira oqishini va boshqa manbalar bilan bog'liq muammolarni oldini oladi.

- **Nosozliklarni tuzatish:** Istisnolar kodni disk raskadrovka qilishni osonlashtiradi, chunki ular istisnoga olib kelgan usul chaqiruvlari ketma-ketligini aks ettiruvchi stek izini beradi. Bu ishlab chiquvchilarga istisno sodir bo'lgan joyni aniq belgilashga yordam beradi va muammoni hal qilishni osonlashtiradi.

- **Mustahkamlik:** Istisnolardan foydalangan holda, ishlab chiquvchilar kutilmagan vaziyatlarni oldindan ko'ra oladi va ularni yanada mustahkamroq tarzda hal qiladi. Bu kutilmagan kirish yoki tashqi omillarni oqilona hal qila oladigan yanada bardoshli va ishonchli dasturiy ta'minotni yaratishga yordam beradi.

Umuman olganda, dasturlashda istisnolardan foydalanish kodning barqarorligi, barqarorligi va umumiy sifatiga hissa qo'shadigan ko'plab afzalliklarni beradi. Xatolarni samarali hal qilish va ma'lumot beruvchi xato xabarlarini taqdim etish orqali istisnolar yanada ishonchli va foydalanuvchilarga qulay dasturiy ta'minotni yaratishga yordam beradi.

### **Dasturlashdagi istisnolarga misollar**

Istisnolar dasturlashning muhim qismidir, chunki ular ishlab chiquvchilarga dasturni bajarish paytida yuzaga kelishi mumkin bo'lgan xatolarni qayta ishlash va ularni tiklashga imkon beradi. Umumiy istisno turlariga misollar keltiramiz:

**1. NullPointerException:** Ushbu istisno dastur metodni chaqirishga yoki null bo'lgan ob'ektdagi o'zgaruvchiga kirishga harakat qilganda yuzaga keladi. Tegishli null tekshiruvlar mavjudligini ta'minlash orqali uni oldini olish mumkin.

**2. ArrayIndexOutOfBoundsException:** Bu istisno, dastur massiv chegarasidan tashqarida joylashgan indeksli massiv elementiga kirishga harakat qilganda yuzaga keladi. Ushbu istisnoning oldini olish uchun ishlab chiquvchilar

elementlarga kirishdan oldin massiv chegaralarini tekshirishlari kerak.

**3. FileNotFoundException:** Ushbu istisno dastur mavjud bo'lmagan faylga kirishga harakat qilganda yuzaga keladi. Istisnoni qo'lga olish va muqobil harakat yo'nalishini ta'minlash uchun try-catch bloklari yordamida ishlov berish mumkin.

**4. ArithmeticException:** Bu istisno arifmetik amal natijasida toshib ketish yoki nolga bo'linish sodir bo'lganda yuzaga keladi. Tegishli tekshiruvlarni amalga oshirish va ushbu stsenariylarga ehtiyotkorlik bilan munosabatda bo'lish orqali uni oldini olish mumkin.

**5. ClassCastException:** Ushbu istisno ob'ekt mos kelmaydigan turga o'tkazilganda yuzaga keladi. Turni tekshirish va mos quyish mexanizmlari yordamida undan qochish mumkin.

**6. IllegalArgumentException:** Ushbu istisno usulga noqonuniy argument uzatilganda yuzaga keladi. Usulni chaqirishdan oldin kirish parametrlarini tekshirish orqali hal qilinishi mumkin.

Bu dasturlashda yuzaga kelishi mumkin bo'lgan ko'plab istisnolarning bir nechta misollari. Istisnolarni samarali tushunish va boshqarish orqali ishlab chiquvchilar o'z dasturlarining ishonchliligi va mustahkamligini oshirishlari mumkin.

**Xulosa.** Xulosa qilib aytganda, matn dasturlash sohasida, xususan, C# tilida istisnolarning hal qiluvchi rolini ta'kidlaydi. U NullPointerException va ArrayIndexOutOfBoundsException kabi istisnolarning har xil turlarini aniq tasvirlab beradi va kutilmagan xatolarni oqilona boshqarish uchun try-catch bloklaridan foydalanish muhimligini ta'kidlaydi. Muhokama istisnolardan foydalanishning kengroq afzalliklarini o'z ichiga oladi: kod modulliligi va o'qilishini yaxshilashdan kod xavfsizligini mustahkamlash va resurslarni boshqarishni osonlashtirishgacha. FileNotFoundException kabi misollarni ko'rsatish orqali matn maxsus istisnolardan foydalanishning amaliy qo'llanilishini tushuntiradi. Umuman olganda, hikoya mustahkam, xavfsiz va foydalanuvchilarga qulay dasturiy ta'minot ilovalarini yaratishga hissa qo'shadigan ishlab chiquvchilar uchun ajralmas vosita sifatida istisnolardan strategik foydalanishni targ'ib qiladi.

**Foydalanilgan adabiyotlar ro'yhati.**

1. John Paul Mueller. C# 10.0 All-in-One For Dummies. Published by: John Wiley & Sons, Inc., 2022. — 830 с.
2. A.Troelsen, R.Japikse. Pro C# 8 with .NET Core 3. Foundational Principles and Practices in Programming. Ninth Edition. Apress, 2020. – 1223 с.
3. А.Васильев. Программирование на C# для начинающих. Основные свидания.: – М.: "Эксмо", 2018. – 592 с.
4. Фленов М. Е. Библия C#. — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2016. — 544 с.
5. Фармонов, Ш., & Камбарова, Д. (2022). КАК ПОМОЧЬ УЧЕНИКАМ РАЗВИТЬ ИНТЕРЕС К УЧЕБЕ. Центральноазиатский журнал образования и инноваций, 1(2), 118-120.
6. Farmonov, S., & Rahmatjonov, M. (2023). FLUTTER YORDAMIDA PLATFORMALARARO KUTUBXONANI ISHLAB CHIQUISH TECHNOLOGIYASI. Центральноазиатский журнал образования и инноваций, 2(6 Part 2), 124-127.
7. Raxmonjonovich, F. S. (2023). USE OF BLENDED LEARNING TECHNOLOGY IN ORGANIZING INDEPENDENT EDUCATION OF STUDENTS. MATERIALLAR TO'PLAMI, 352.
8. Farmonov, S., & Karimova, M. (2023). MODERN METHODS TO DEVELOP MATHEMATICAL THINKING IN SCHOOLCHILDREN. Бюллетень педагогов нового Узбекистана, 1(6 Part 2), 28-38.
9. Tojiyev, T., Boynazarov, A., & Farmonov, S. (2022). PHARMACOKINETICS IS A DESCRIPTION OF DRUGS AND THEIR BEHAVIOR IN THE HUMAN BODY BY BUILDING A MATHEMATICAL MODEL. Евразийский журнал медицинских и естественных наук, 2(13), 146-149.
10. Farmonov, S., & Nazirov, A. (2023). C# DASTURLASH TILIDA GRAY KODI BILAN ISHLASH. В CENTRAL ASIAN JOURNAL OF EDUCATION

AND INNOVATION (Т. 2, Выпуск 12, сс. 71–74). Zenodo.  
<https://doi.org/10.5281/zenodo.10417073>

11. Farmonov , S. ., & Kudratullayev , U. (2023). C# VA .NET CORE ZAMONAVIY DASTURLASHNING RIVOJLANISHI SIFATIDA. Бюллетень педагогов нового Узбекистана, 1(12), 70–73. извлечено от <https://in-academy.uz/index.php/yopa/article/view/24748>

12. Farmonov , S., & Qo'qonboyev , A. . (2023). C# 10 XUSUSIYATLARI: TILDAGI SO'NGGI YAXSHILANISHLARNI O'RGANISH. Бюллетень педагогов нового Узбекистана, 1(12), 77–79. извлечено от <https://in-academy.uz/index.php/yopa/article/view/24808>

13. Farmonov, S., & Hakimov , M. . (2023). C# DA DELEGATLAR VA HODISALARNING ROLI: VOQEALARGA ASOSLANGAN DASTURLASHGA CHUQUR KIRISH. Бюллетень педагогов нового Узбекистана, 1(12), 80–84. извлечено от <https://in-academy.uz/index.php/yopa/article/view/24809>

14. Фармонов, Ш., & Хайдарова, С. (2022). Обобщенный метод Бубнова-Галеркина для уравнений с дробно-дифференциальным оператором. Norwegian Journal of Development of the International Science, (99), 10-15.