



OB'EKTGA YO'NALTIRILGAN DASTURLASH ZAMONAVIY DASTURLASHNING ASOSI SIFATIDA

Farmonov Sherzodbek Raxmonjonovich

Farg'ona davlat universiteti amaliy matematika va
informatika kafedrası katta o'qituvchisi
farmonovsh@gmail.com

Rasuljonova Zarnigor Egamnazar qizi

Farg'ona davlat universiteti 2-kurs talabasi
zarnigorrasuljonova@gmail.com

<https://doi.org/10.5281/zenodo.10477330>

ARTICLE INFO

Qabul qilindi: 01-January 2024 yil

Ma'qullandi: 04- January 2024 yil

Nashr qilindi: 10- January 2024 yil

KEY WORDS

Ob'ektga yo'naltirilgan dasturlash (OYD), C# dasturlash tili, Sinflar va ob'ektlar, Sinf a'zolari, Kirish modifikatorlari, Xususiyatlar va usullar, C# da konstruktorlar, Inkapsulyatsiya, OYDning asosiy tamoyillari.

ABSTRACT

Ushbu maqolada ob'ektga yo'naltirilgan dasturlash(OYD)ning asosiy tamoyillari haqida fikr yuritilib, ularning mohiyati ochib berilgan. Bundan tashqari maqolada OYDning yutuq va kamchiliklari tahlil qilingan.

Ob'ektga yo'naltirilgan dasturlash(OYD) - bu ishlab chiquvchilarga o'z kodlarida real dunyo ob'ektlarini modellashtirish imkonini beruvchi kuchli paradigma. OYD ning asosiy xususiyatlaridan biri bu ob'ektlarni yaratish uchun sxema bo'lib, xizmat qiladigan sinflarni yaratish va ulardan foydalanish qobiliyatidir. Microsoft tomonidan ishlab chiqilgan ko'p qirrali va keng qo'llaniladigan dasturlash tili C# da sinflar kodni tuzishda asosiy rol o'ynaydi. Ushbu maqolada biz C# da sinfni qanday e'lon etish va uning a'zolari bilan ishlashni o'rganamiz.

Obyektga yo'naltirilgan dasturlash asoslari – dasturlashning bir paradigmasini o'z ichiga olgan prinsiplar va konseptlardir. Bu asoslar, dasturchilarga kodni yordamchi, tuzatishi oson va tuzatishga ishonchli dasturlar yozishda qulayliklar ta'minlaydi. Quyidagi OYD asoslarini ko'rib chiqamiz:

- Obyekt (Object): Dasturlashda barcha narsalar obyekt sifatida ko'rib chiqiladi. Obyekt, o'z malumotlari va qilishlarini ifodalovchi xususiyatlar va metodlarga ega bo'lgan narsa hisoblanadi. Misol uchun, bir shaxs obyekt, uning ismi, yoshi, manzili kabi xususiyatlarni o'z ichiga oladi.
- Sinf (Class): Sinf, bir yoki bir nechta o'zaro bir qator xususiyatlarni (malumotlar) va funksiyalarni (metodlar) o'z ichiga olgan narsani ifodalovchi abstrakt tavsifdir. Sinfdan obyekt yaratiladi va shu obyektning xususiyatlari va metodlari klass tomonidan belgilanadi.
- Xususiyatlar (Properties): Obyektning malumotlari yoki o'zgaruvchilari, sinfning

xususiyatlarini ifodalaydi. Misol uchun, avtomobil obyekti uchun marka, model, yil va h.k.

- **Metodlar (Methods):** Sinfning bajarishi mumkin bo'lgan qilishlarni ifodalovchi funksiyalar. Metodlar, obyektning xususiyatlari bilan ishlash uchun yoziladi.
- **Vorislik (Inheritance):** Bu asos, bir sinfning boshqa klassdan xususiyatlarni va metodlarni o'zlashtirish imkonini beradi. Misol uchun, "Hayvon" klassining "Uchuyorgi" klassi bilan nasli bor.
- **Polimorfizm:** Bu asos, bir qo'llanuvchi (funksiya yoki interfeys) orqali bir nechta klassni ishlatish imkonini anglatadi. Bu, bir qo'llanuvchini barcha qo'llanma shakllarini qabul qilish va ularga moslashtirilgan harakatlarni bajarishni ta'minlaydi.
- **Inkapsulyatsiya (Encapsulation):** Bu asos, obyektning ichidagi ma'lumotlarga faqat obyektning xususiyatlariga murojaat qilish orqali murojaat qilishni ta'minlaydi. Bu, obyektning ichidagi ma'lumotlarni yashirish va ularga faqat obyektni tuzatgan sinf orqali murojaat qilish imkonini beradi.

OYD - dasturchilarga kodni modular va yomonlashganlikni kamaytirish, qayta ishlashni osonlashtirish va o'zgartirishlarni xavfsizlashtirish imkonini taqdim etadi. Bu sababli, juda ko'p dasturlash tillarida va texnologiyalarda OYD qo'llanilmoqda.

Bundan tashqari OYDning o'ziga xos yutuq va kamchiliklari mavjud. Obyektga yo'naltirilgan dasturlash yutuqlari - dasturchilarga kodni qulay, oson tuzatish, va modulyar tuzatish uchun imkoniyatlar taqdim etishi bilan bog'liq. OYDning yutuqlarilari quyidagilardan iborat:

- a. **Modulyarlik:** Kodni modullarga bo'lib tuzish, dasturni tuzatish va boshqa modullar bilan integratsiya qilish imkonini osonlashtiradi.
- b. **Qo'shimcha funksiyalar:** Yangi xususiyatlarni va metodlarni qo'shish oson va tuzatishni ta'minlaydi.
- c. **Qayta ishlash osonligi:** Kodni qayta ishlash va boshqa dasturlar bilan integratsiya qilish oson bo'lib, dasturni o'zgartirish va kengaytirish imkonini beradi.
- d. **Xavfsizlik:** Encapsulation va nasl asoslari, obyektning ichidagi ma'lumotlarni himoyalashni va o'zgartirishni qavatlovchi sifatda ishlatiladi.
- e. **Tuzatilishning tezligi:** Klasslar va obyektlar, tasvirlangan narsalar va xizmatlar uchun model yaratishda yordam beradi, shuningdek, dastur tuzilishini osonlashtiradi.
- f. **Kodni qulaylik:** OYD, kodni tuzatish, boshqarish va o'zgartirishda kodni qulaylik ta'minlaydi. Dasturchilar qulaylik bilan kodni tuzatishadi.

Obyektga yo'naltirilgan dasturlash paradigmasi, bugungi kunda ko'pgina dasturlash tillarida va texnologiyalarda juda mashhur. Uning foydalari va imkoniyatlari, dasturchilar uchun kuchli va samarali bir kod yozish modelini taqdim etadi.

Obyektga yo'naltirilgan dasturlash paradigmasining barcha yutuqlari va afzalliklari bo'lganligiga qaramasdan, uning kamchiliklari ham mavjud. OOPning kamchiliklari quyidagicha:

1. **Complexity (Murakkablik).** OOP dasturlash odatda murakkab tuzilganlikka ega bo'ladi. Agar dastur murakkab bo'lsa, obyektlar, klasslar va ularning bog'liq bo'lgan muloqotlari to'g'risidagi mantiqiy tushunchalarni o'rganish va tushuntirish qiyinlashishi mumkin.
2. **Performance (Ishlab chiqish).** Obyektga yo'naltirilgan dasturlash, ba'zi hodisalarda yaddosh va ishlab chiqishga ko'proq zaxira zarrachalar ishlatadi. Bu esa dastur bajarilishi vaqtini ortishi mumkin.

3. Learning Curve (O'rganish davri). OOP paradigmasini tushunish va uning prinsiplarini o'rganish uchun dasturchilarga ko'plab vaqt va tajribaga ehtiyot bo'lishi kerak bo'lishi mumkin. Bu tarz dasturlashning konseptlari o'rganish uchun vaqt sarflanishi mumkin.
 4. Memory Consumption (Xotira sarfligi). Obyektga yo'naltirilgan dasturlashda obyektlar va ularning mantiqiy bog'liq xususiyatlari va metodlari ko'p xotira sarflaydi. Bu esa kattalikda xotira sarfligiga olib kelishi mumkin.
 5. Maintenance (Qo'llab-quvvatlash). OOP da, dasturlash tuzilmalarining qo'llab-quvvatlashi va bozorlarga moslashishining saqlanishi va ta'mirlanishi bir necha muammolar tugrisidagi cheteliklardan biri bo'lishi mumkin.
 6. Overhead (Yuzasidan o'tish). Obyektga yo'naltirilgan dasturlashda, dasturlash tillarida qo'shimcha funktsiyalar va qo'llab-quvvatlash vositalari uchun qo'shimcha o'lchovlar ishlatilishi mumkin. Bu esa kodning o'zini murakkablashtirishi mumkin.
 7. Performance (Ishlab chiqish). Ba'zi muhitlar va dastur turlari uchun OOP, ishlab chiqish vaqti va maqsadlarni bajarish uchun mos kelmaydi. Yoki, shu jarayonda OOP paradigmasi qiyinchiliklar tugrisida e'tirof etish mumkin.
 8. Abstraction Gap (Axborotlar orasidagi farq). OOP da axborotlar, klasslar va obyektlar orasidagi farqli tushunchalar va bog'liq qurilmalar yuzasidan hosil bo'ladi. Shunday bo'lishi mumkinki, o'zgaruvchilar yoki metodlar axborotlarga murojaat qilganda, ko'p farqli qurilmalar o'rtasidagi muammo paydo bo'ladi.
- Obyektga yo'naltirilgan dasturlashning kamchiliklari, boshqalar bilan solishtirilganda, dasturchining maqsad va loyihalari uchun moslashtirilishi mumkinligi mavjud. Har bir dasturlash paradigmasining o'ziga xos muammo va afzalliklari mavjud bo'ladi.

Foydalanilgan adabiyotlar ro'yhati:

1. John Paul Mueller. C# 10.0 All-in-One For Dummies. Published by: John Wiley & Sons, Inc., 2022. — 830 c.
2. A.Troelsen, R.Japikse. Pro C# 8 with .NET Core 3. Foundational Principles and Practices in Programming. Ninth Edition. Apress, 2020. – 1223 c.
3. A.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. Farmonov, S. (2023). C# DASTURLASH TILIDA GRAY KODI BILAN ISHLASH. Центральноазиатский журнал образования и инноваций, 2(12 Part 2), 71-74.
6. Farmonov, S., & Jo'rayeva, M. (2023, December). DASTURLASHDA POLIMORFIZMNING ANAMIYATI. In Международная конференция академических наук (Vol. 2, No. 13, pp. 5-8).
7. Farmonov, S., & Toirov, S. (2023). NETDA DASTURLASHNING ZAMONAVIY TEXNOLOGIYALARINI O'RGANISH. Theoretical aspects in the formation of pedagogical sciences, 2(22), 90-96.
8. Farmonov Sherzodbek Raxmonjonovich, & Xolmatov Oxunjon Xasan o'g'li. (2024). DASTURLASHDA SANA VA VAQTLAR BILAN ISHLASH . Ta'lim Innovatsiyasi Va Integratsiyasi, 11(11), 3–6. Retrieved from <http://web-journal.ru/index.php/ilmiy/article/view/2481>
9. 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).
10. Farmonov Sherzodbek Raxmonjonovich, & Kazimjonova Madinaxon Habibullo qizi.

- (2024). FLIPPED CLASSROOM TEXNOLOGIYASIDAN TA'LIM JARAYONIDA FOYDALANISH METODIKASI . Ta'lim Innovatsiyasi Va Integratsiyasi, 11(11), 7–11. Retrieved from <http://web-journal.ru/index.php/ilmiy/article/view/2482>
11. Raxmonjonovich, F. S. (2023). Array ma'lumotlar tizimini talabalarga o'qitishda Blockchain metodidan foydalanish. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 541-547.
12. Raxmonjonovich, F. S. (2023). Dasturlashda interfeyslardan foydalanishning ahamiyati. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 425-429.
13. Raxmonjonovich, F. S. (2023). Dasturlashda obyektga yo'naltirilgan dasturlashning ahamiyati. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 434-438.
14. Raxmonjonovich, F. S. (2023). Dasturlash tillarida fayllar bilan ishlash mavzusini Blended Learning metodi yordamida o'qitish. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 464-469.
15. Raxmonjonovich, F. S. (2023). DASTURLASHDA ISTISNOLARNING AHAMIYATI. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 475-481.
16. Raxmonjonovich, F. S. (2023). Dasturlashda abstraksiyaning o'rni. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 482-486.
17. Raxmonjonovich, F. S., & Ravshanbek o'g'li, A. A. (2023). Zamonaviy dasturlash tillarining qiyosiy tahlili. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 430-433.
18. Raxmonjonovich, F. S. (2023). C# dasturlash tilida fayl operatsiyalari qo'llashning qulayliklari haqida. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 439-446.
19. Raxmonjonovich, F. S. (2023). C# tilida ArrayList bilan ishlashning afzalliklari. Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari, 2(2), 470-474.
20. Farmonov Sherzodbek Raxmonjonovich, & Rustamova Humoraxon Sultonbek qizi. (2024). C# DASTURLASH TILIDA TO'PLAMLAR BILAN ISHLASH . Ta'lim Innovatsiyasi Va Integratsiyasi, 11(10), 210–214. Retrieved from <http://web-journal.ru/index.php/ilmiy/article/view/2480>