

C# DA OBYEKTGA YO'NALTIRILGAN DASTURLASH (OOP) ASOSLARI: XALQARO STANDARTLARGA MOS YONDASHUV

Nazarova Ominaxon¹

Ubaydullayeva Dilorom²

Namangan Davlat Universiteti

KEYWORDS

OOP, C#, sinflar, obyektlar, meros olish, inkapsulyatsiya, polimorfizm, interfeyslar, SOLID tamoyillari, dizayn naqshlari, dasturiy arxitektura.

ABSTRACT

Ushbu maqolada C# dasturlash tilida obyektga yo'naltirilgan dasturlash (OOP) tamoyillari xalqaro standartlar (IEEE, ACM, ISO/IEC 25010) asosida tahlil qilinadi. OOP ning modullilik, qayta foydalanish va kengaytirish imkoniyatlari amaliy misollar va ilmiy tadqiqotlar bilan isbotlangan. Sinflar, obyektlar, meros olish, inkapsulyatsiya, polimorfizm va interfeyslar kabi asosiy tushunchalar C# dagi kod misollari bilan tushuntirilgan. Maqolada shuningdek, SOLID tamoyillari va dizayn naqshlari (Singleton, Factory) ning zamonaviy dasturiy arxitekturadagi ahamiyati ta'kidlangan.

2181-2675/© 2025 in XALQARO TADQIQOT LLC.

DOI: **10.5281/zenodo.15333829**

This is an open access article under the Attribution 4.0 International (CC BY 4.0) license (<https://creativecommons.org/licenses/by/4.0/deed.ru>)

Kirish

Obyektga yo'naltirilgan dasturlash (OOP) haqiqiy dunyo ob'ektlarini modellashtirishga asoslangan dasturlash usulidir. C# — Microsoft tomonidan ishlab chiqilgan .NET platformasining asosiy tili bo'lib, OOP tamoyillarini amalga oshirishda keng qo'llaniladi. ISO/IEC 25010 standartiga ko'ra, OOP quyidagi afzalliklarga ega:

- Modullilik (murakkab tizimlarni oddiy qismlarga ajratish),
- Qayta foydalanish (meros olish va kompozitsiya orqali),
- Kengaytirish imkoniyati (polimorfizm va interfeyslar yordamida),
- Xavfsizlik (inkapsulyatsiya orqali ma'lumotlarni himoya qilish).

2. C# DA OOP NING ASOSIY TAMOYILLARI

2.1 Sinflar va Obyektlar

¹ Namangan Davlat Universiteti Fizika-matematika fakulteti Amaliy matematika yo'nalishi (1-bosqich talabasi)

² Namangan Davlat Universiteti Fizika-matematika fakulteti Raqamli ta'lim texnologiyalari kafedrasida o'qituvchisi

- Sinf — ob'ektlarning xususiyatlari (fields) va harakatlari (methods) ni belgilaydigan shablon.

C#

```
public class Avtomobil
{
    public string Model { get; set; } // Xususiyat
    public void Start() => Console.WriteLine("Dvigatel ishga tushdi."); // Metod
}
```

- Obyekt — sinfning namunasi.

C#

```
Avtomobil myCar = new Avtomobil();
myCar.Model = "Tesla Model S";
myCar.Start();
```

2.2 Meros Olish

- Bir sinfdan boshqa sinfni yaratish orqali kodni qayta ishlatish.

C#

```
public class ElektrAvtomobil : Avtomobil
{
    public int BatareyaQuvvati { get; set; }
}
```

- SOLID tamoyillaridan Liskov Almashtirish Prinsipi (LSP) ga mos keladi.

2.3 Inkapsulyatsiya

- Ma'lumotlarni himoya qilish uchun cheklangan kirish (private maydonlar va umumiy getter/setter lar).

C#

```
public class BankHisob
{
    private double _balans;
    public double BalansniOlish() => _balans;
}
```

2.4 Polimorfizm

- Metodlarni qayta yozish:

C#

```
- public class Hayvon
- {
-     public virtual void Ovoz() => Console.WriteLine("Ovoz");
```

- }
- public class It : Hayvon
- {
- public override void Ovoz() => Console.WriteLine("Vov-vov");
- }
-
- Interfeyslar : Turli sinflar uchun umumiy xatti-harakatlar.

C#

```
public interface IHaydash
{
    void Haydash();
}
public class Avtomobil : IHaydash
{
    public void Haydash() => Console.WriteLine("Avtomobil haydash");
}
```

2.5 Abstrakt Sinflar

- Namuna olish mumkin bo'lmagan, to'liq bo'lmagan shablonlar.

C#

```
public abstract class Shakl
{
    public abstract double Yuza();
}
public class Aylana : Shakl
{
    public double Radius { get; set; }
    public override double Yuza() => Math.PI * Radius * Radius;
}
```

3. Amaliy Qo'llanma va Eng Yaxshi Amaliyotlar

- Dizayn Naqshlari:
- Singleton: Sinf faqat bitta namunaga ega bo'lishini ta'minlaydi.

C#

```
public sealed class Singleton
{
    private static Singleton _instance;
    private Singleton() { }
    public static Singleton Instance => _instance ??= new Singleton();
}
```

}

- Factory Method: Ob'ekt yaratishni kichik sinflarga topshiradi.
- SOLID Tamoyillari:
 - Yagona Mas'uliyat Prinsipi (SRP): Sinf faqat bitta vazifani bajarishi kerak.
 - Bog'liqlikni Inversiya Qilish (DIP): Aniq sinflarga emas, abstraksiyalarga bog'lanish.

4. Xulosa

C# da OOP tamoyillari zamonaviy dasturiy ta'minot ishlab chiqishning asosidir. Sinflar, meros olish, inkapsulyatsiya va polimorfizmni o'rganish orqali ishonchli, xavfsiz va qulay dasturlar yaratish mumkin. Kelajakda bulutli texnologiyalar (Azure) va sun'iy intellekt bilan integratsiyani o'rganish mumkin.

Foydalanilgan adaliyotlar

1. Gamma, E. va boshqalar. (1994). Dizayn Naqshlari: Qayta Ishlatiladigan Obyektga Yo'naltirilgan Dasturiy Ta'minot Elementlari.
2. Microsoft .NET Hujjatlari. (2023). C# Dasturlash Qo'llanmasi.
3. ISO/IEC 25010:2011. Tizimlar va dasturiy ta'minot sifat talablari va baholash (SQuaRE).