

**INKAPSULYATSIYA, ATRIBUTLAR VA XUSUSIYATLAR.
VORISIYLIK****Tojimamatov Israiljon Nurmamatovich***Farg'ona davlat universiteti*israiltojimatov@gmail.com**Qaxramonova Muxlisa Jumaxo'ja qizi***Farg'ona davlat universiteti*qahramonovakmalxuja@gmail.com

Annotatsiya. *Ushbu maqolada obyektga yo'naltirilgan dasturlashning asosiy tushunchalari — inkapsulyatsiya, atributlar, xususiyatlar va vorislikning nazariy asoslari tahlil qilinadi. Python dasturlash tili doirasida ushbu tushunchalarning ahamiyati, ular orqali dasturiy tizimlarni modellashtirish, ma'lumotlarni himoyalash va kodni qayta ishlatish imkoniyatlari yoritilgan. Maqola mazkur tushunchalarning dasturiy amaliyotdagi o'rni va ular o'rtasidagi mantiqiy bog'liqlikni tahlil qilishga qaratilgan.*

Аннотация. *В статье анализируются теоретические основы объектно-ориентированного программирования - инкапсуляция, атрибуты, продукция и наследование. Подчеркивается важность этого процесса в языке программирования Python, а также для моделирования программ, защиты данных и восстановления кода. В статье анализируется конкретная логическая связь этой деловой активности и практики.*

Abstract. *This article analyzes the theoretical foundations of object-oriented programming - encapsulation, attributes, production and inheritance. The importance of this process in the Python programming language is highlighted, as well as for modeling programs, data protection and code recovery. The article analyzes the specific logical connection of this business activity and practice.*



Kalit so'zlar. *Obyektga yo'naltirilgan dasturlash, inkapsulyatsiya, atribut, xususiyat, vorislik, Python, dasturlash nazariyasi, ma'lumotlarni himoyalash, kodni qayta ishlatish, dasturiy model.*

Ключевые слова. *Объектно-ориентированное программирование, инкапсуляция, атрибут, свойство, наследование, Python, теория программирования, защита данных, повторное использование кода, модель программирования.*

Keywords. *Object-oriented programming, encapsulation, attribute, property, inheritance, Python, programming theory, data protection, code reuse, programming model.*

Kirish

Zamonaviy dasturlashda obyektga yo'naltirilgan dasturlash muhim o'rin tutadi. Ushbu yondashuv real dunyo obyektlarini dasturiy modelga ko'chirish imkonini beradi. Obyektga yo'naltirilgan dasturlashning asosiy tushunchalaridan inkapsulyatsiya, atributlar, xususiyatlar va vorislik — murakkab dasturiy tizimlarni modellashtirishda muhim nazariy va amaliy asos bo'lib xizmat qiladi.

Inkapsulyatsiya dasturchiga ma'lumotlarni tashqi aralashuvdan himoya qilish, faqat kerakli qismlarni ochiq saqlash imkonini beradi. Bu esa dasturiy tizimning ishonchliligi va barqarorligini ta'minlaydi. Atributlar esa obyektlarning holatini tavsiflovchi xususiyatlar bo'lib, ular orqali obyektlar o'zaro farqlanadi yoki umumlashtiriladi. Shu bilan birga, xususiyatlar orqali atributlarga nazorat ostida murojaat qilish va kerakli cheklovlarni qo'llash mumkin bo'ladi. Bu esa ma'lumotlarning yaxlitligi va dasturda intizomli ishlov berilishini ta'minlaydi.

Vorislik esa mavjud klasslar asosida yangi klasslar hosil qilishga, kodni takrorlamasdan qayta foydalanishga xizmat qiladi. Bu yondashuv dasturiy modullarning o'zaro bog'liqligini samarali tashkil etish va kodni kengaytirish imkonini yaratadi.

Inkapsulyatsiya dasturda axborot xavfsizligini ta'minlaydi, ya'ni obyektning ichki holati faqat u bilan bog'liq metodlar orqali boshqarilishi mumkin bo'ladi. Bu esa noto'g'ri ishlov berishdan himoya qiladi hamda modul ichidagi murakkablikni



tashqi foydalanuvchi uchun soddalashtiradi. Atributlar obyektning individual holatini saqlashda va unga xos xususiyatlarni aniqlashda muhim ahamiyat kasb etadi. Ular yordamida obyektlar bir-biridan farqlanadi yoki umumiy guruhlariga ajratiladi.

Xususiyatlar atributlarga dinamik va nazoratli kirish imkonini beradi. Bu yordamchi mexanizm orqali dasturchi foydalanuvchidan atributlar ustida to'g'ridan-to'g'ri amallar bajarilishining oldini oladi. Shu bilan birga, bu mexanizm orqali dasturda yaxlitlik va barqarorlikni ta'minlash mumkin bo'ladi.

Vorislik obyektga yo'naltirilgan dasturlashda kodni takrorlamasdan, mavjud strukturani kengaytirish orqali yangilik yaratish vositasidir. Bu yordamida biror bir umumiy model asosida bir nechta ixtisoslashtirilgan modellarga o'tish mumkin bo'ladi. Vorislik nafaqat kod samaradorligini oshiradi, balki tizimni kengaytirish va saqlab borishni yengillashtiradi.

Bugungi kunda dasturlash muhitida yuqori darajadagi murakkablik va ma'lumotlar xavfsizligiga bo'lgan talab ortib bormoqda. Shunday sharoitda obyektga yo'naltirilgan dasturlash — xususan, inkapsulyatsiya, vorislik va atributlar kabi konseptlar — dasturiy tizimlarni modularlashtirish, kengaytirish va texnik xizmat ko'rsatishda muhim rol o'ynaydi. Inkapsulyatsiya orqali dasturiy tizimning har bir moduli alohida boshqariladi, bu esa texnik nosozliklar ehtimolini kamaytiradi. Atributlar va xususiyatlar yordamida obyektlar holatini qat'iy nazorat qilish, noto'g'ri foydalanish holatlarini bartaraf etish imkoniyati yaratiladi. Vorislik esa mavjud kodlarni qayta ishlamasdan kengaytirish, yangi funksiyalarni joriy qilish uchun qulaylik yaratadi.

Python tilida bu yondashuvlar orqali katta hajmdagi loyihalar, ayniqsa veb-ilovalar, avtomatlashtirish tizimlari, sun'iy intellekt modullari va ma'lumotlar tahlili sohalarida yuqori samaradorlikka erishish mumkin. Obyektga yo'naltirilgan dasturlash elementlarini chuqur o'zlashtirish, zamonaviy dasturchining kasbiy salohiyatini oshirishda muhim omil sanaladi.

Misollar. Ushbu misolda __ism va __yosh atributlari tashqaridan to'g'ridan-to'g'ri chaqirilmaydi. Ular bilan faqat get_ism va set_ism metodlari orqali ishlash mumkin.


```
main.py  Untitled2.py  +  
1 class Talaba:  
2     def __init__(self, ism, yosh):  
3         self.__ism = ism      # xususiy (private) atribut  
4         self.__yosh = yosh  
5  
6     def get_ism(self):  
7         return self.__ism  
8  
9     def set_ism(self, yangi_ism):  
10        self.__ism = yangi_ism  
11
```

Bu yerda turi — barcha Avtomobil obyektlari uchun umumiy atribut bo'lsa, model va rang har bir obyektga xos qiymatlarga ega bo'ladi.

```
main.py  Untitled3.py  Untitled4.py  +  
1 class Avtomobil:  
2     turi = "transport"      # klass atributi  
3  
4     def __init__(self, model, rang):  
5         self.model = model   # obyekt atributi  
6         self.rang = rang  
7
```

Endi foydalanuvchi nomi atributini to'g'ridan-to'g'ri o'qiy oladi yoki qiymatini o'zgartiradi, ammo bu jarayon nazorat ostida amalga oshadi.

```
main.py  Untitled3.py  Untitled4.py  +  
1 class Kitob:  
2     def __init__(self, nomi):  
3         self.__nomi = nomi  
4         @property  
5     def nomi(self):  
6         return self.__nomi  
7         @nomi.setter  
8     def nomi(self, yangi_nomi):  
9         if isinstance(yangi_nomi, str):  
10            self.__nomi = yangi_nomi  
11
```

**Xulosa**

Obyektga yo'naltirilgan dasturlash tushunchalarini chuqur o'rganish va ularni amalda to'g'ri qo'llash, zamonaviy dasturlash muhitida yuqori darajadagi samaradorlikka erishish imkonini beradi. Inkapsulyatsiya yordamida ma'lumotlarni himoyalash, atributlar orqali obyektlar holatini ifodalash, xususiyatlar bilan nazoratli kirishni ta'minlash va vorislik orqali mavjud funksional imkoniyatlarni kengaytirish dasturchining har bir loyihasida asosiy ustunlikka aylanishi mumkin. Python dasturlash tili ushbu konseptlarni qulay va tushunarli tarzda amalga oshirish uchun ideal muhit bo'lib, ushbu maqolada ko'rib chiqilgan g'oyalar bugungi kundagi dasturlash amaliyotining negizini tashkil qiladi.

FOYDALANILGAN ADABIYOTLAR RO'YXATI:

1. Абрамян М.Э. "Практикум по программированию на языке Python" — М.: Бином, 2021.
2. Шабалин Д. "Объектно-ориентированное программирование на Python", СПб: Питер, 2020.
3. Zelle, J. "Python Programming: An Introduction to Computer Science", Franklin, Beedle & Associates, 2016.
4. Lutz, M. "Learning Python", 5th Edition, O'Reilly Media, 2013.
5. Tadjibayeva D.S., Yusupova D.O. "Python dasturlash asoslari", Toshkent: Innovatsiya nashriyoti, 2022.
6. G'ulomov S., Qurbonov B. "Python dasturlash tili va amaliyot", Toshkent: Fan va texnologiya, 2021.
7. <https://docs.python.org/3/> — Python dasturlash tilining rasmiy hujjatlari va OOP bo'yicha ma'lumotlar (ingliz tilida).
8. <https://habr.com/ru/articles/443200/> — "Конструкторы и атрибуты объектов в Python" maqolasi (rus tilida, Habr platformasi).