

OB'EKTGA YO'NALTIRILGAN DASTURLASH

O'ROLOVA MARJONA ABDUSALOM QIZI*Denov tadbirkorlik va pedagogika instituti Tadbirkorlik**va boshqaruv fakulteti "Kompyuter injineri"**yo'nalishi 2- bosqich bakalavr talabasi**marjonamorolova@gmail.com*

Annotatsiya: Ob'ektga yo'naltirilgan dasturlash (OOP) - bu kompyuter dasturlash modeli bo'lib, u funksiyalar va mantiq emas, balki ma'lumotlar yoki ob'ektlar atrofida dasturiy ta'minot dizaynini tashkil qiladi. Ob'ektni noyob atributlar va xatti-harakatlarga ega bo'lgan ma'lumotlar maydoni sifatida aniqlash mumkin.

Kalit so'zlar: OOP, inkapsulatsiya, abstrakt, metod, meros, polimorfizm.

Abstract: Object-oriented programming (OOP) is a computer programming model that organizes software design around data or objects rather than functions and logic. An object can be defined as a field of data with unique attributes and behaviors.

Keywords: OOP, encapsulation, abstract, method, inheritance, polymorphism.

Аннотация: Объектно-ориентированное программирование (ООП) — это модель компьютерного программирования, которая организует разработку программного обеспечения вокруг данных или объектов, а не функций и логики. Объект можно определить как поле данных с уникальными атрибутами и поведением.

Ключевые слова: ООП, инкапсуляция, абстракция, метод, наследование, полиморфизм.

OOP asosiy e'tiborni ishlab chiquvchilar ularni manipulyatsiya qilish uchun zarur bo'lgan mantiqqa emas, balki boshqarmoqchi bo'lgan ob'ektlarga qaratadi. Dasturlashning bunday yondashuvi katta, murakkab va faol yangilanadigan yoki xizmat ko'rsatuvchi dasturlar uchun juda mos keladi. Bunga ishlab chiqarish va dizayn dasturlari, shuningdek, mobil ilovalar kiradi; masalan, OOP tizimi simulyatsiya dasturlarini ishlab chiqarish uchun ishlatilishi mumkin. Ob'ektga yo'naltirilgan dasturni tashkil etish, shuningdek, loyihalar guruhlariga bo'lingan holda hamkorlikda ishlab chiqish uchun usulni foydali qiladi. OOP ning o'shimcha afzalliklariga kodning qayta ishlatilishi, kengaytirilishi va samaradorligi kiradi.

OOPda birinchi qadam dasturchi manipulyatsiya qilmoqchi bo'lgan barcha ob'ektlarni to'plash va ularning bir-biri bilan qanday bog'liqligini aniqlashdir - bu mashq ma'lumotlarni modellashtirish deb nomlanadi. Ob'ektga misollar ism va manzil kabi xususiyatlar bilan tavsiflangan inson kabi jismoniy shaxslardan tortib, vidjetlar kabi kichik kompyuter dasturlarigacha bo'lishi mumkin. Ob'ekt ma'lum bo'lgach, u

o'z ichiga olgan ma'lumotlar turini va uni boshqarishi mumkin bo'lgan har qanday mantiqiy ketma-ketlikni belgilaydigan ob'ektlar sinfi bilan etiketlanadi. Har bir aniq mantiqiy ketma-ketlik usul sifatida tanilgan. Ob'ektlar xabarlar deb ataladigan yaxshi belgilangan interfeyslar bilan aloqa qilishi mumkin.

Ob'ektga yo'naltirilgan dasturlashning tuzilishi yoki qurilish bloklari quyidagilarni o'z ichiga oladi:

Sinflar - bu foydalanuvchi tomonidan belgilangan ma'lumotlar turlari bo'lib, ular alohida ob'ektlar, atributlar va usullar uchun loyiha sifatida ishlaydi.

Ob'yektlar - bu maxsus belgilangan ma'lumotlar bilan yaratilgan sinf namunalari. Ob'ektlar real dunyo ob'ektlariga yoki mavhum ob'ektga mos kelishi mumkin. Sinf dastlab aniqlanganda, tavsif belgilangan yagona ob'yekt hisoblanadi.

Metodlar - bu ob'ektning xatti-harakatlarini tavsiflovchi sinf ichida aniqlangan funktsiyalar. Sinf ta'riflarida mavjud bo'lgan har bir usul misol ob'ektiga havola bilan boshlanadi. Bundan tashqari, ob'ekt tarkibidagi pastki dasturlarga misol usullari deyiladi. Dasturchilar qayta foydalanish yoki funksionallikni bir vaqtning o'zida bitta ob'ekt ichida saqlash usullaridan foydalanadilar.

Atributlar sinf shablonida aniqlanadi va ob'ekt holatini ifodalaydi. Ob'ektlar atributlar maydonida saqlanadigan ma'lumotlarga ega bo'ladi. Sinf atributlari sinfning o'ziga tegishlidir.

Ob'ektga yo'naltirilgan dasturlash quyidagi prinsiplarga asoslanadi:

Inkapsulatsiya. Ushbu tamoyil barcha muhim ma'lumotlar ob'ekt ichida joylashganligini va faqat tanlangan ma'lumotlarning ochiqqligini bildiradi. Har bir ob'ektning amalga oshirilishi va holati belgilangan sinf ichida shaxsiy saqlanadi. Boshqa ob'ektlarda ushbu sinfga kirish yoki o'zgartirishlar kiritish huquqi yo'q. Ular faqat davlat funktsiyalari yoki usullari ro'yxatini chaqirishlari mumkin. Ma'lumotlarni yashirishning bu xususiyati dasturning katta xavfsizligini ta'minlaydi va ma'lumotlarning kutilmagan buzilishlarini oldini oladi.

Abstraksiya. Ob'ektlar faqat boshqa ob'ektlardan foydalanish uchun tegishli bo'lgan ichki mexanizmlarni ochib beradi, keraksiz amalga oshirish kodini yashiradi. Olingan sinf o'zining funktsional imkoniyatlariga ega bo'lishi mumkin. Ushbu kontsepsiya ishlab chiquvchilarga vaqt o'tishi bilan qo'shimcha o'zgartirishlar yoki qo'shimchalar kiritishga yordam beradi.

Meros. Sinflar boshqa sinflardagi kodlarni qayta ishlatishi mumkin. Ob'ektlar o'rtasidagi munosabatlar va pastki sinflar tayinlanishi mumkin, bu esa ishlab chiquvchilarga yagona ierarxiyani saqlab qolgan holda umumiy mantiqni qayta ishlatishga imkon beradi. OOP ning bu xususiyati

ma'lumotlarni chuqurroq tahlil qilishga majbur qiladi, ishlab chiqish vaqtini qisqartiradi va yuqori darajadagi aniqlikni ta'minlaydi.

Polimorfizm. Ob'ektlar xatti-harakatlarni almashish uchun mo'ljallangan va ular bir nechta shaklga ega bo'lishi mumkin. Dastur ota-sinfdan ushbu ob'ektning har bir bajarilishi uchun qaysi ma'no yoki foydalanish zarurligini aniqlaydi va kodni takrorlash zaruratini kamaytiradi. Keyin ota- sinfning funktsionalligini kengaytiradigan bolalar sinfi yaratiladi. Polimorfizm har xil turdagi ob'ektlarning bir xil interfeys orqali o'tishiga imkon beradi.

Simula birinchi ob'ektga yo'naltirilgan dasturlash tili deb hisoblansa-da, bugungi kunda OOP bilan boshqa ko'plab dasturlash tillari qo'llaniladi. Ammo ba'zi dasturlash tillari OOP bilan boshqalarga qaraganda yaxshiroq birlashadi. Misol uchun, sof OOP tillari hisoblangan dasturlash tillari hamma narsani ob'ekt sifatida ko'rib chiqadi. Boshqa dasturlash tillari asosan OOP uchun mo'ljallangan, lekin ba'zi protsessual jarayonlarni o'z ichiga oladi.

Masalan, mashhur sof OOP tillariga quyidagilar kiradi:

Ruby

Skala

JADE

Zumrad

Asosan OOP uchun mo'ljallangan dasturlash tillariga quyidagilar kiradi:

Java

Python

C++

OOP bilan bog'langan boshqa dasturlash tillariga quyidagilar kiradi:

Visual Basic.NET

PHP

JavaScript

OOPning afzalliklari quyidagilardan iborat:

Modullilik. Inkapsulyatsiya ob'ektlarni o'z-o'zidan saqlashga imkon beradi, bu esa muammolarni bartaraf etish va hamkorlikda ishlab chiqishni osonlashtiradi.

Qayta foydalanish imkoniyati. Kod meros orqali qayta ishlatilishi mumkin, ya'ni jamoa bir xil kodni bir necha marta yozishi shart emas.

Hosildorlik. Dasturchilar bir nechta kutubxonalar va qayta ishlatiladigan kodlar yordamida yangi dasturlarni tezroq qurishlari mumkin.

Osonlik bilan yangilanadigan va kengaytirilishi mumkin. Dasturchilar tizim funktsiyalarini mustaqil ravishda amalga oshirishlari mumkin.

Interfeys tavsiflari. Ob'ektlar bilan aloqa qilish uchun ishlatiladigan xabarlarini uzatish usullari tufayli tashqi tizimlarning tavsiflari oddiy.

Xavfsizlik. Inkapsulyatsiya va abstraktsiyadan foydalanib, murakkab kod yashiringan, dasturiy ta'minotga texnik xizmat ko'rsatish osonroq va internet protokollari himoyalangan.

Moslashuvchanlik. Polimorfizm bitta funktsiyaga u joylashgan sinfga moslashish imkonini beradi. Turli ob'ektlar ham bir xil interfeys orqali o'tishi mumkin.

Ob'ektga yo'naltirilgan dasturlash modeli ishlab chiquvchilar tomonidan bir nechta sabablarga ko'ra tanqid qilingan. Eng katta muammo shundaki, OOP dasturiy ta'minotni ishlab chiqishning ma'lumotlar komponentiga ortiqcha urg'u beradi va hisoblash yoki algoritmlarga etarlicha e'tibor bermaydi. Bundan tashqari, OOP kodini yozish ancha murakkab va kompilyatsiya qilish ko'proq vaqt talab qilishi mumkin.

OOPning muqobil usullari quyidagilardan iborat:

Funksional dasturlash. Bunga telekommunikatsiya va nosozliklarga chidamli tizimlar uchun ishlatiladigan Erlang va Scala kabi tillar kiradi.

Strukturaviy yoki modulli dasturlash. Bunga PHP va C# kabi tillar kiradi.

Imperativ dasturlash. OOP ga alternativa modellarga emas, balki funktsiyaga e'tibor qaratadi va C++ va Java-ni o'z ichiga oladi.

Deklarativ dasturlash. Ushbu dasturlash usuli vazifa yoki istalgan natija nima ekanligi haqidagi bayonotlarni o'z ichiga oladi, lekin unga qanday erishish mumkin emas. Tillarga Prolog va Lisp kiradi. Mantiqiy dasturlash. Ko'pincha rasmiy mantiqqa asoslangan va Prolog kabi tillardan foydalanadigan bu usul muammoli sohaga oid faktlar yoki qoidalarni ifodalovchi jumlar to'plamini o'z ichiga oladi. U qoidaga asoslangan mantiqiy so'rovlardan foydalanishi mumkin bo'lgan vazifalarga qaratilgan. Ko'pgina ilg'or dasturlash tillari ishlab chiquvchilarga modellarni birlashtirishga imkon beradi, chunki ular turli xil dasturlash usullari uchun ishlatilishi mumkin. Masalan, JavaScriptdan OOP va funktsional dasturlash uchun foydalanish mumkin.

Xulosa: OOP va mikroservislar bilan ishlaydigan dasturchilar OOP tamoyillarini qo'llash orqali umumiy mikroservislar muammolarini hal qilishlari mumkin.

FOYDALILGAN ADABIYOTLAR RO'YHATI:

1. <https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming->
2. [-oriented-programming-OOP#:~:text=Object%2Doriented%20programming%20\(OOP\)%20is%20a%20computer%20programming%20model,has%20unique%20attributes%20and%20behavior.](https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming-)
3. https://en.wikipedia.org/wiki/Object-oriented_programming