**SUPER KLASS VA VORIS KLASS**

Obyektga yo'naltirilgan dasturlashning qulayliklaridan biri bu klasslardan boshqa klass yaratish imkoniyati. Bizga kerak bo'lgan yangi klass, avval yaratilgan boshqa klass bilan o'xshashlik joylari bo'lsa, biz bu klassdan voris klass yaratishimiz mumkin. Bunda asl klass - ota, yoki super klass deb ataladi.

Super klassdan yaratilgan voris klass otaning barcha yoki tanlangan xususiyatlari va metodlarini meros olish bilan birga, o'ziga xos xususiyat va metodlariga ega bo'ladi.

Keling boshlanishiga Shaxs klassini yaratamiz, bu klassimiz keyinchalik boshqa klasslar uchun super klass vazifasini bajaradi:

Copy

class Shaxs:

"""Shaxslar haqida ma'lumot"""

def \_\_init\_\_(self,ism,familiya,passport,tyil):

self.ism = ism

self.familiya = familiya

self.passport = passport

self.tyil = tyil

def get\_info(self):

"""Shaxs haqida ma'lumot"""

info = f"{self.ism} {self.familiya}. "

info += f"Passport:{self.passport}, {self.tyil}-yilda tug`ilgan"

return info

def get\_age(self,yil):

"""Shaxsning yoshini qaytaruvchi metod"""

return yil - self.tyil

Klassimizni tekshirib ko'ramiz:

Copy

inson = Shaxs("Hasan","Alimov","FB001122",1995)

print(f"{inson.get\_info()}. {inson.get\_age(2021)} yoshda.")

Natija: Hasan Alimov. Passport:FB001122, 1995-yilda tug`ilgan. 26 yoshda.

**VORIS KLASS YARATISH**

Endi avvalgi darsimizda yaratgan Talaba klassimizni qaytadan yaratamiz. Bu safar, avvalgidan farqli ravishda, Talaba ni yaratishda, Shaxs dan super klass sifatida foydalanamiz:

Copy

class Talaba(Shaxs):

"""Talaba klassi"""

def \_\_init\_\_(self, ism, familiya, passport, tyil):

"""Talabaning xususiyatlari"""

super().\_\_init\_\_(ism, familiya, passport, tyil)

Kodimizni tahlil qilaylik:

* 1-qatorda klass nomidan so'ng, qavs ichida super klass nomini berdik
* 5-qatorda (def \_\_init\_\_ ichida) klassimiz super klassning xususiyatlarini meros olishini ko'rsatdik

Yangi yaratgan Talaba klassimiz Shaxsning barcha xususiyatlari va metodlariga ega bo'ladi.

Copy

talaba = Talaba("Valijon","Aliyev","FA112299",2000)

print(talaba.get\_info())

Natija: Valijon Aliyev. Passport:FA112299, 2000-yilda tug`ilgan

Talaba klassi uchun alohida get\_info() metodini yozmagan bo'lsakda*,* bu metod Talabaga Shaxsdan meros o'tdi.

Huddi shu kabi get\_age() metodiga ham murojat qilishimiz mumkin:

Copy

>>>print(talaba.get\_age(2021))

21

Dastur davomida super klass voris klasslardan avval yozilgan (chaqirilgan) bo'lishi kerak.

**VORIS KLASSGA XOS XUSUSIYATLAR VA METODLAR**

Hozirgi ko'rinishda Talaba va Shaxs klasslari o'rtasida hech qanday farq yo'q. Keling Talaba klassimizga o'ziga xos xususiyatlar va metodlar yarataylik. Avvalosiga, talabaning bosqichi va ID raqamini xususiyat sifatida qo'shamiz. Bunda ID raqami obyekt yaratilishida parameter sifatida uzatiladi, bosqich esa standart qiymatga ega.

Copy

class Talaba(Shaxs):

"""Talaba klassi"""

def \_\_init\_\_(self, ism, familiya, passport, tyil,idraqam):

"""Talabaning xususiyatlari"""

super().\_\_init\_\_(ism, familiya, passport, tyil)

self.idraqam = idraqam

self.bosqich = 1

Endi yangi, Talaba obyektini yaratishda qo'shimcha idraqam parametrini ham kiritish talab qilinadi:

Copy

talaba = Talaba("Valijon","Aliyev","FA112299",2000,"0000012")

So'ngra, bu qiymatlarni qaytaruvchi alohida metodlar yozamiz:

Copy

class Talaba(Shaxs):

"""Talaba klassi"""

def \_\_init\_\_(self, ism, familiya, passport, tyil,idraqam):

"""Talabaning xususiyatlari"""

super().\_\_init\_\_(ism, familiya, passport, tyil)

self.idraqam = idraqam

self.bosqich = 1

def get\_id(self):

"""Talabaning ID raqami"""

return self.idraqam

def get\_bosqich(self):

"""Talabaning o'qish bosqichi"""

return self.bosqich

Metodlarni tekshirib ko'ramiz:

Copy

>>>print(f"{talaba.get\_info()}. ID raqami:{talaba.get\_id()}")

Valijon Aliyev. Passport:FA112299, 2000-yilda tug`ilgan. ID raqami:0000012

>>>print(f"{talaba.get\_bosqich()}-bosqich talabasi")

1-bosqich talabasi

Shu zayilda yangi klassimizga istalgancha yangi xususiyatlar va metodlar qo'shishimiz mumkin. Bunda, agar yangi xususiyat yoki metod super klassga ham aloqador bo'lsa uni birdan super klassga qo'shish tavsiya qilinadi.

Voris klass boshqa klass uchun super klass bo'lishi mumkin.

**POLIMORFIZM — SUPER KLASS METODLARINI QAYTA YOZISH**

Voris klassga super klassdan meros qolgan istalgan metodni qayta talqin qilish mumkin. Avvalgi misolimizdagi get\_info() super metodini ko'raylik, bu metod talabaning ismi, familiyasi, passport raqami va tug'ilgan yilini qaytaradi:

Copy

>>> print(talaba.get\_info())

Valijon Aliyev. Passport:FA112299, 2000-yilda tug`ilgan

Endiget\_info() metodi talabaga mos ma'lumotlar qaytarishi uchun, Talaba klassi ichida huddi shu nomli metodni qayta yozamiz:

Copy

class Talaba(Shaxs):

"""Talaba klassi"""

def \_\_init\_\_(self,ism,familiya,passport,tyil,idraqam):

"""Talabaning xususiyatlari"""

super().\_\_init\_\_(ism, familiya, passport, tyil)

self.idraqam = idraqam

self.bosqich = 1

def get\_id(self):

"""Talabaning ID raqami"""

return self.idraqam

def get\_bosqich(self):

"""Talabaning o'qish bosqichi"""

return self.bosqich

def get\_info(self):

"""Talaba haqida ma'lumot"""

info = f"{self.ism} {self.familiya}. "

info += f"{self.get\_bosqich()}-bosqich. ID raqami: {self.idraqam}"

return info

Metodni tekshirib ko'ramiz:

Copy

>>> print(talaba.get\_info())

Valijon Aliyev. 1-bosqich. ID raqami: 0000012

**OBYEKT ICHIDA OBYEKT**

Ba'zida klassimiz xususiyatlar va ular bilan ishlaydigan metodlarga to'lib ketishi, bu esa o'z navbatida obyektga murojat qilishni qiyinlashitirishi mumkin. Shunday holatlarda ba'zi xususiyatlarni alohida klass ko'rinishida yozish va keyinchalik bu klassdan yaratilgan obyektni boshqa obyektning xususiyati sifatida foydalanish mumkin.

Misol uchun, yuqoridagi Shaxs klassimizga yana bir manzil degan xususiyat qo'shaylik. Odatda manzil bir nechta qismlardan iborat bo'ladi (xonadon, ko'cha, mahalla, tuman/shahar, viloyat, indeks va hokazo) va ularning har biri uchun Shaxs ichida alohida xususiyat yaratmasdan, alohida manzil degan klassga yuklash maqsadga muvofiq bo'ladi.

Copy

class Manzil:

"""Manzil saqlash uchun klass"""

def \_\_init\_\_(self,uy,kocha,tuman,viloyat):

"""Manzil xususiyatlari"""

self.uy = uy

self.kocha = kocha

self.tuman = tuman

self.viloyat = viloyat

def get\_manzil(self):

"""Manzilni ko'rish"""

manzil = f"{self.viloyat} viloyati, {self.tuman} tumani, "

manzil += f"{self.kocha} ko'chasi, {self.uy}-uy"

return manzil

Talaba klassimizga ham qo'shimcha manzil xususiyatini qo'shamiz:

Copy

class Talaba(Shaxs):

"""Talaba klassi"""

def \_\_init\_\_(self,ism,familiya,passport,tyil,idraqam,manzil):

"""Talabaning xususiyatlari"""

super().\_\_init\_\_(ism, familiya, passport, tyil)

self.idraqam = idraqam

self.bosqich = 1

self.manzil = manzil

def get\_id(self):

"""Talabaning ID raqami"""

return self.idraqam

def get\_bosqich(self):

"""Talabaning o'qish bosqichi"""

return self.bosqich

def get\_info(self):

"""Talaba haqida ma'lumot"""

info = f"{self.ism} {self.familiya}. "

info += f"{self.get\_bosqich()}-bosqich. ID raqami: {self.idraqam}"

return info

Keling endi talaba obyektini qayta yaratamiz. Bu safar talabaning manzili ham alohida obyekt sifatida talaba ga uzatiladi:

Copy

talaba\_manzil = Manzil(12,'Olmazor',"Bog'bon","Samarqand")

talaba = Talaba("Valijon","Aliyev","FA112299",2000,"0000012",talaba\_manzil)

Obyekt ichidagi obyektning xususiyatlari va metodlariga ham avvalgidek nuqta orqali murojat qilishimiz mumkin:

Copy

>>> print(talaba.manzil.get\_manzil())

Samarqand viloyati, Bog'bon tumani, Olmazor ko'chasi, 12-uy

>>> print(talaba.manzil.tuman)

Bog'bon

**AMALIYOT**

Talaba klassiga yana bir, fanlar degan xususiyat qo'shing. Bu xususiyat parametr sifatida uzatilmasin va obyekt yaratilganida bo'sh ro'yxatdan iborat bo'lsin (self.fanlar=[])

Fan degan yangi klass yarating va bu klassdan turli fanlar uchun alohida obyektlar yarating.

Talaba klassiga fanga\_yozil() degan yangi metod yozing. Bu metod parametr sifatida Fan klassiga tegishli obyektlarni qabul qilib olsin va talabaning fanlar ro'yxatiga qo'shib qo'ysin.

Talabaning fanlari ro'yxatidan biror fanni o'chirib tashlash uchun remove\_fan() metodini yozing. Agar bu metodga ro'yxatda yo'q fan uzatilsa "Siz bu fanga yozilmagansiz" xabarini qaytarsin.

Yuqoridagi Shaxs klassidan boshqa turli voris klasslar yaratib ko'ring (masalan Professor, Foydalanuvchi, Sotuvchi, Mijoz va hokazo)

Har bir klassga o'ziga hoz xususiyatlar va metodlar yuklang.

Barcha yangi klasslar uchun get\_info() metodini qayta yozib chiqing.

Voris klasslardan yana boshqa voris klass yaratib ko'ring. Misol uchun Foydalanuvchi klassidan Admin klassini yarating.

Admin klassiga foydalanuvchida yo'q yangi metodlar yozing, masalan, ban\_user() metodi konsolga "Foydalanuvchi bloklandi" degan matn chiqarsin.