

### Pythonda meros tushunchasi:

Biz o'tkan darslarda class haqida tanishgan edik bu darslik ham deyarli class haqida desak mubolag'a bo'lmaydi!

Pythonda meros tushunchasi. Bu nafaqat python tilida umumiy dasturlash tillarida mavjud tushuncha hisoblanadi Xo'sh meros o'zi qanday tushuncha? Meros bizga boshqa sinfdan barcha usullar va xususiyatlarni meros qilib oladigan sinfni aniqlash imkonini beradi. Ya'ni tushunarli qilib aytganda boshqa class da ishlatilgan argumentlarni bu argumentga meros sifatida qoldirish hisoblanadi. Avvalgi darslarimizda copy() metodiga o'xshash.

Order class - bu meros qilib olingan sinf, uni asosiy sinf deb ham ataladi.

Child class - bu boshqa sinfdan meros bo'lib qolgan sinf, hosila sinf deb ham ataladi.

Sizlar birinchi ko'rishimiz kerak bo'lgan meros bu ota-ona merosi hisoblanadi:

Har qanday sinf ota-sinf bo'lishi mumkin, shuning uchun sintaksis boshqa sinflarni yaratish bilan bir xil: **Kod\_1** ga e'tibor bering.

### Standart qiymat

Pythonda obyektning ba'zi xususiyatlarini parametr yordamida uzatmasdan, klass yaratishda unga standart qiymat berib ketishimiz mumkin. Keling, yangi klass yaratamiz. Bu klassimiz 3 ta xususiyatga ega bo'lishi lozim ya'ni: ism, familiya, tyil. Biz yana bitta qo'shimcha, sinf nomli xususiyat qo'shamiz, va unga standart qiymat sifatida 9 beramiz **Kod\_2**.

### Standart qiymatni o'zgartirish:

Obyektning standart qiymatiga ham boshqa xususiyatlar kabi nuqta orqali murojat qilishimiz va uning qiymatini almashtirishimiz mumkin:

**Kod\_3** Bu usulda asl o'zgaruvchiga berilgan qiymat o'zgarmay qolish extimoli juda yuqori! Shu sababdan **Kod\_4** da gi kabi o'zgartirish maslahat beriladi. Umuman olganda buni xar xil yo'llar bilan yangilashimiz mumkin. update\_sinf() metodiga murojat qilganimizda talabning bosqichi oshib boradi **Kod\_5** bu misolimizda 1 kursdan 2 kursga meros qolmoqda. Yani ota-ona merosi bolaga qolmoqda. Kod\_4 da esa otadan onaga meros qolmoqda edi yani bunda asl o'zgaruvchi ham o'zgarmoqda.

Super funksiyasi. Python shuningdek, super() bolalar sinfini ota-onasidan barcha usullar va xususiyatlarni meros qilib oladigan funktsiyaga ega: **Kod\_6** yani Talaba nomli sinfdan Yangitalaba nomli class ga meros qoldirmoqdamiz.

**Graduationyear** Sinfga chaqirilgan xususiyatni qo'shish: **Kod\_7** tamomlash funksiyasini qo'shamiz yani birinchi 4 keyin esa 6 yil berdik kod\_7 ga axamiyat bering.

Obyektga yo'naltirilgan dasturlashning afzalligi, turli obyektlar o'rtasida o'zaro munosabatlarni oson yo'lga qo'yish mumkinligida. Buni misolda ko'rish uchun, yangi Fan degan klass yaratamiz va fanimizga talabalar qo'shish imkoniyatini ham yaratamiz (add\_student() metodi) add ingliz tilidan tarjima qilinganda qo'shish degan manoni bildiradi hamda ko'p manoli so'z hisoblanadi **Kod\_8** ga e'tibor bering.

Fanimizga talaba qo'shish uchun add\_student() metodini chaqiramiz. Bu metod parametr sifatida Talaba obyektini qabul qiladi va uni talabalar ro'yxatiga qo'shadi. Shuningdek, bu metod yangi talaba qo'shilganda talabalar\_soni ni 1 taga oshirib qo'yadi.

Keling boshlanishiga yangi fan obyektini va bir nechta talabalarni yaratamiz **Kod\_9** ga etibor bering.

Fanimizga yozilgan talabalar haqida tushunarli ma'lumot olish uchun esa Fan klassiga yana bir, get\_students() degan, yangi metod qo'shamiz. Bu metod talabalar ichidagi har bir talaba obyektiga murojat qilib, get\_info() metodi yordamida talabaning ma'lumotlarini oladi, ro'yxatga qo'shadi va shu ro'yxatni qaytaradi: **Kod\_10** Shu o'rinda Pythonda ro'yxatlar bilan ishlashning qulayliklaridan birini ham ko'rsatib o'tsak. Yuqoridagi kodimizning 294-qatorida e'tibor bersangiz, biz yangi ro'yxat shakllantirishda 1 qator koddan foydalandik.

Kodimizni tahlil qilsak, self.talabalar ichidagi har bir talaba uchun get\_info() metodini bajar degan ma'no kelib chiqadi. Kodni kvadrat qavslar ichiga olganimiz uchun esa, har bir tsikl natijasi avtomat ravishda ro'yxatga qo'shib boriladi.

Welcome tushunchasi: **Kod\_11** bu metod aksariyat hollarda ishlatilinmaydi va shu sababdan hozirgi vaqtda yo'qolib boruvchi metodlardan biridir. Bu metod bazi kirish joylari yoki sayt yaratilganda login parol to'g'ri kiritilganda Salom falonchiyev saytga hush kelibsiz! Kabi argumentlarni chiqarishda ishlatiladi. Kod\_11 da bunga yaqqol misolni ko'rishimiz mumkin.

### Nuatq yoki metod?

Pythondagi obyektning o'ziga xos xususiyatlaridan biri, obyektning xususiyatiga nuqta orqali murojat qilish mumkin. Misol uchun avval yaratgan talaba1 obyektining ismini bilish uchun talaba1.ism deb yozish kifoya. Bu o'ziga yarasha qulay bo'lsada, bu usuldan foydalanmagan afzal. Sababi, vaqt o'tib klassingiz takomillashishi, uning ba'zi xususiyatlari o'zgarishi, o'chirilishi yoki almashtirilishi mumkin. Shunday holatlarda nuqta orqali murojat qilish siz kutgan natijani bermasligi va dastur xato ishlashiga olib kelishi mumkin. Bunday holatlarning oldini olish uchun esa, obyektning xususiyatlarini metod orqali olishni odat qilish tavsiya qilinadi. Huddi shu kabi, obyektning xususiyatlarini yangilash uchun ham alohida metodlar yozgan afzal. **Dasturchilar orasida obyektning xususiyatlarini o'zgartiradigan metodlarni set (o'zgartir) so'zi bilan, xususiyatlarni qaytaradigan metodlarni esa get (olish) so'zi bilan boshlash qoida qilib olingan. Masalan: set\_name() va get\_name() kabi.**

Agar yuqoridagi qoidalarga amal qilgan holda, Talaba klassimizni yangilaydigan bo'lsak, u taxminan shunday ko'rinishga ega bo'ladi **Kod\_12**

Huddi shu kabi, Fan klassimizning yakuniy ko'rinishi quyidagicha bo'ladi: **Kod\_13**

**dir()** funksiyasi yordamida istalgan obyekt yoki klassning xususiyatlari va metodlarini ko'rib olishimiz mumkin: **Kod\_14**

Lekin bunda har bir klass bilan keluvchi maxsus dunder metodlar ham chiqib keldi. Dunder metodlar ikki pastki chiziq ( ) bilan boshlanadi va maxsus holatlar uchun saqlab qo'yilgan. Biz hozircha faqat \_\_init metodi bilan tanishdik, qolganlari bilan keyingi darslarimizda yana ko'rishamiz. Dunder metodlardan keyin esa biz murojat qilishimiz mumkin bo'lgan metodlar ro'yxati kelgan.

**Dunder — double underscore (ikki pastki chiziq) so'zlarining qisqartmasi.**

**Kod\_15 Dunder metodlarni tashlab, bizga kerak metodlarni qaytaruvchi soddada funksiya yozamiz.**

Yuqorida zikr qilingan dunder metodlardan biri bu dict metodi bo'lib, bu metod obyektning xususiyatlarini lug'at ko'rinishida qaytaradi: **Kod\_16**