

тузилганлиги билан бошқа мобил иловалардан тубдан фарқ қилади. 5330200 – Ахборот тизимлари ва технологиялари таълим йўналишининг ўқув режасидаги умумқасбий фанлар блокидаги “Сонли усуллар” фанидан мобил дастур талабаларнинг мустақил таълим олиш имкониятини такомиллаштириш мақсадида ишлаб чиқилди ва амалиётга жорий қилиниб, талабалар ва соҳа мутахассислариға фойдаланиш учун тавсия қилинди.

Яхши маълумки, мамлакатимиздаги олий таълим муассасаларида 2022-2023 ўқув йилидан бошлаб масофавий таълим шаклини жорий қилиш режалаштирилган. Масофавий таълимда бундай турдаги мобил

иловалардан фойдаланиш муҳим аҳамият касб этади.

Хулоса ўрнида шуни таъкидлаш лозимки, ўқув материалларини тақдим этишда АКТдан фойдаланиш фанни тўлиқ ўзлаштиришга ёрдам бериб, талабаларнинг билим ва кўникмаларини ривожланиш даражаси оширилади. Яратилган мобил дастур фойдаланувчи (талаба)ларнинг дарсдан ташқари мустақил фаолиятларини ривожлантирибгина қолмай, уларнинг келгусида мустақил ишлай олиш қobiliятларини оширишга хизмат қилади. Бу эса, уларнинг индивидуал фаолият олиб бориши билан бир қаторда, ўз имкониятлари ва келажакдаги муваффақиятли йўлларини белгилашда хизмат қилади.

Адабиётлар

1. Ўзбекистон Республикаси Президентининг 2020 йил 5 октябрдаги “Рақамли Ўзбекистон–2030 стратегиясини тасдиқлаш ва уни самарали амалга ошириш чора-тадбирлари тўғрисида” ПФ-6079-сонли Фармони (Манбаа: <https://lex.uz/docs/5030957>).
2. Бегимкулов У.Ш. Педагогик таълим жараёнларини ахборотлаштиришни ташкил этиш ва бошқариш назарияси ва амалиёти.: Пед. фан. докт.... дисс. автореф. - Т.: 2007. 37-б.
3. Мўминов Б.Б. Информатикадан амалий ва лаборатория машғулоти учун педагогик дастурий таъминот яратиш ва фойдаланиш методикаси (академик лицейларнинг аниқ фанлар йўналиши мисолида). Пед. фан. номзоди.... дисс. -Т.: 2009. 177-б.
4. Расулова З.Д. Талабаларнинг креативлигини ривожлантиришда дастурий таълим воситаларидан фойдаланиш методикасини такомиллаштириш (технологик таълим йўналиши мисолида). Педагогика фанлари бўйича фалсафа доктори (PhD). Дисс. 2021. 126-б.

РЕЗЮМЕ. Мазкур мақолада умумқасбий фанлар блокидаги “Сонли усуллар” фанидан ишлаб чиқилган мобил дастурнинг таълим соҳасида тутган ўрни ҳамда унинг қандай шаклда жорий этилиши бўйича таҳлиллар амалга оширилган. Тайёрланган мобил иловани ўқув жараёнига жорий этишда илмий асосланган ёндашув шакллантирилган ва тегишли хулосалар чиқарилган.

РЕЗЮМЕ. В данной статье анализируется роль мобильного приложения в сфере образования и способы его реализации, разработанного по предмету «Численные методы» в блоке общеобразовательных наук. Сформирован научно обоснованный подход к внедрению разработанного мобильного приложения в образовательный процесс и сделаны соответствующие выводы.

SUMMARY. This article analyzes the role of a mobile application in the field of education and ways to implement it, developed in the subject “Numerical Methods” in the block of general education sciences. A scientifically based approach to the implementation of the developed mobile application in the educational process has been formed and the corresponding conclusions have been drawn.

TALABALARGA OBYEKTGA YO‘NALTIRILGAN DASTURLASHNI O‘RGATISHNING FOYDALI JIHATLARI (Python dasturlash tili misolida)

D.E.Saidova – o‘qituvchi
Qarshi davlat universiteti

Tayanch so‘zlar: obyektga yo‘naltirilgan dasturlash, Python, sinf, obyekt, usul, atribut, xatti-harakat, funksiya, holat, o‘zgaruvchilar, inkapsulyatsiya, meros, polimorfizm.

Ключевые слова: объектно-ориентированное программирование, Python, класс, объект, метод, атрибут, поведение, функция, состояние, переменная, инкапсуляция, наследование, полиморфизм.

Key words: object-oriented programming, Python, class, object, method, attribute, behavior, function, state, variable, encapsulation, inheritance, polymorphism.

Ўзбекистон Respublikasining oliy va o‘rta ta‘lim tizimida dasturlash tillari sifatida uzoq yillardan buyon Paskal, C++, C#, Visual Basic, Java va boshqa tillar o‘qitilib kelinmoqda. Faqat so‘ggi yillarda Python dasturlash tili maktab darsliklari va oliy ta‘limning ayrim yo‘nalishlarida o‘qitilmoqda. Shuning uchun maktab o‘quvchilari va talabalar yuqoridagi dasturlash tillarida obyektga yo‘naltirilgan dasturlash bo‘yicha amaliy ko‘nikmalarga ega bo‘lishsa-da, Python tilida bu borada yetarlicha ko‘nikmalarga ega emaslar.

Avvalo shuni aytishimiz mumkinki, Python obyektga yo‘naltirilgan dasturlash tili sifatida yaratilgan. Bu quyidagi tamoyillarni hisobga olgan holda qurilganligini anglatadi [5]:

1. Undagi barcha ma‘lumotlar obyektlar bilan ifodalangani.
2. Dastur bir-biriga xabar yuboradigan o‘zaro ta’sir qiluvchi obyektlar to‘plami sifatida tuzilishi mumkin.
3. Har bir obyekt xotiraning o‘ziga xos qismiga ega va boshqa obyektlardan iborat bo‘lishi mumkin.
4. Har bir obyektning turi bor.
5. Bir xil turdagi barcha obyektlar bir xil xabarlarini qabul qilishlari va bir xil harakatlarni bajarishlari mumkin.

Boshqa keng tarqalgan dasturlash tillari bilan solishtirganda, Python tili obyektga yo‘naltirilgan dasturlash bilan bog‘liq quyidagi xususiyatlarga ega [7]:

1. Har qanday berilgan (qiymat) obyektidir. Raqam, satr, ro‘yxat, massiv va boshqalar hamma narsa obyektidir.

O‘rnatilgan sinflar obyektlari (oldingi jumlada sanab o‘tilganlar kabi) va foydalanuvchi tomonidan belgilangan sinflar obyektlari (dasturchi yaratadiganlar) mavjud. Yagona o‘zaro ta’sir mexanizmi uchun operatorni ortiqcha yuklash usullari taqdim etiladi.

2. Klass ham o‘z nom maydoniga ega obyektidir. Bu darslar turkumining hech bir joyida aytilmagan. Shuning uchun “obyekt” so‘zi o‘rniga “instansiya” so‘zini qo‘llash to‘g‘riroq bo‘ladi.

3. Python tilida inkapsulyatsiyaga unchalik ahamiyat berilmaydi. Boshqa dasturlash tillarida odatda sinfda e‘lon qilingan xususiyatga bevosita kirish mumkin emas. Uni o‘zgartirish uchun maxsus usul taqdim etilishi mumkin. Pythonda buni sinfning mulkiga tashqaridan kirish orqali amalga oshirish oson. Shunga qaramay, Python hali ham sinfdagi o‘zgaruvchilarga kirishni cheklashning maxsus usullarini taqdim etadi

Ma‘lumki, obyektga yo‘naltirilgan dasturlash asosan real hayotdagi obyektlarni modelashga asoslangan. Demak, obyektga yo‘naltirilgan dasturlashni o‘rgatish obyektlar va ularning o‘zaro ta’siri tushunchasidan boshlanishi kerak [1].

Sinflar va obyektlarni batafsil ko‘rib chiqishdan oldin, biz uchun sinflar va obyektlar tushunchasi Python tilida qanday amalga oshirilishini bilish muhimdir. Chunki sinf va obyektning odatda dasturlash tajribasiga ega bo‘lganlar uchun tushunish juda qiyin. Python tiliga kelsak, muammolar nafaqat yangi boshlanuvchilar uchun, balki C++, Java,

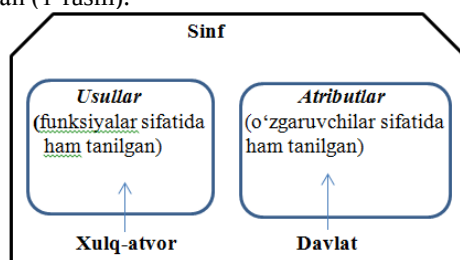
C # va boshqalar kabi tillar misolida obyektga yo'naltirilgan dasturlash usullari bilan tanish bo'lgan o'quvchilar uchun ham paydo bo'lishi mumkin.

Shuningdek, Python tilida obyektga yo'naltirilgan dasturlashni amalga oshirish mexanizmlari va tamoyillari C++, Java va C# da qo'llaniladiganlardan sezilarli darajada farq qiladi. Fundamental farqlar hatto terminologiya darajasida ham mavjud [4].

Oddiy qilib aytganda, obyekt - bu ham obyektlar bo'lgan ma'lumotlarga (o'zgaruvchilarga) ta'sir qiluvchi usullar (funksiyalar) to'plami. Ushbu obyektlar uchun sxema sinfdir. Sinfni obyekt haqidagi barcha tafsilotlarga ega bo'lgan eskiz yoki prototip sifatida tasavvur qiling[3].

Sinfdan foydalanish obyektidagi xatti-harakatlar va holatni birlashtirish imkonini beradi. Xulq-atvor so'zini eshitganingizda, funksiyani o'ylab ko'ring, ya'ni biror narsani bajaradigan (yoki xohlasangiz, xatti-harakatni amalga oshiradigan) kodlar bo'lagi. Davlat so'zini eshitganingizda, o'zgaruvchilarni o'ylab ko'ring, ya'ni sinf ichidagi qiymatlarni saqlash joyi. Biz sinf xatti-harakat va holatni birlashtiradi, deb ta'kidlaganimizda, biz shunchaki sinf funksiyalar va o'zgaruvchilarni paketlashini bildiramiz [2].

Sinflar usullari va atributlariga ega. Pythonda sinfning xatti-harakati usulni yaratish bilan belgilanadi. Metod so'zi - bu sinf ichida aniqlangan funksiyaga berilgan obyektga yo'naltirilgan dasturlash nomi. Nega usullar shunchaki sinf funksiyalari sifatida tan olinmaganligi, sinf o'zgaruvchilari nom atributi bilan ma'lum bo'lgani kabi, vaqt o'tishi bilan yo'qolgan (1-rasm).

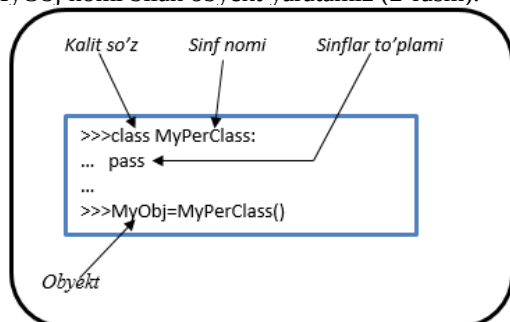


1-rasm. Sinf tarkibiy qismlari

Sinfdan foydalanish uchun siz undan obyekt yaratasiz (quyida bunga misolni ko'rasiz). Bu obyektни instantatsiya-lash deb nomlanadi. Instant so'zini eshitganingizda, ya'ni obyekt yaratish uchun sinfni chaqirasiz.

Ehtimol, ajablanarlisi shundaki, siz hech qanday holat yoki xatti-harakatlarga ega bo'lmagan sinf yaratishingiz mumkin, ammo Python uchun hali ham sinf. Aslida, bunday sinf bo'sh. Keling, sinf misollarimizni bo'sh misol bilan boshlaymiz va u yerdan narsalarni olamiz. Biz tarjimonning >>> ko'rsatmasi bo'yicha ishlaymiz.

Biz MyPerClass deb nomlangan bo'sh sinf yaratishdan boshlaymiz. Biz buni sinf nomiga class kalit so'zi bilan old qo'yish orqali amalga oshiramiz, so'ngra sinfni amalga oshiradigan kodlar to'plamini taqdim etamiz va darhol undan MyObj nomi bilan obyekt yaratamiz (2-rasm).



2-rasm: Sinf va obyekt deklaratsiyasi

Ushbu sinf to'plamida Python kalit so'zi passni o'z ichiga olganligiga e'tibor bering, bu Pythonning bo'sh bayonoti (bu hech narsa qilmaydi).

Python dasturlash hamjamiyatida kichik harflar (ta'kidlash uchun pastki chiziq bilan) yordamida funktsiyalarni nomlash bo'yicha yaxshi o'rnatilgan konvensiya mavjud, sinflarni nomlash uchun esa UpperCase (birlashtirilgan so'zlar, bosh harflar) ishlatiladi. Ushbu konvensiyadan so'ng, my_per_class() funksiya chaqiruvi ekanligi aniq bo'lishi kerak, chunki MyPerClass() obyektни yaratadi. Xuddi shu sinfdan yaratilgan obyektlar sinf bilan bir xil xatti-harakatlarga ega (sinfda belgilangan almashish usullari), lekin holatning o'z nusxasini (atributlari) saqlaydi.

Inkapsulyatsiya - obyektни tashkil etuvchi komponentlarga (usullar va o'zgaruvchilar) kirishni cheklash. Inkapsulyatsiya ba'zi komponentlarni faqat sinf ichida mavjud qiladi. Pythonda inkapsulyatsiya faqat qaysi atributlar ochiq va qaysi biri ichki ekanligi haqida dasturchilar o'rtasidagi kelishuv darajasida ishlaydi. Atribut nomining boshidagi bitta pastki chiziq o'zgaruvchi yoki usul sinf usullaridan tashqarida foydalanish uchun mo'ljallanmaganligini, lekin atribut shu nom bilan mavjudligini bildiradi.

```
>>> class A:
...     def _private(self):
...         print("Bu shaxsiy usul ")
...
...
>>> a = A()
>>> a._private()
Bu shaxsiy usul
```

Atribut nomining boshida qo'sh chiziq ostidagi chiziq qo'shimcha himoyani ta'minlaydi: atributga bu nom orqali kirish imkoni bo'lmaydi.

```
>>> class B:
...     def __private(self):
...         print("Bu shaxsiy usul")
...
...
>>> b = B()
>>> b.__private()
```

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

Attribute Error: 'B' object has no attribute '__private'

Biroq, bu to'liq himoya qilmaydi, chunki atribut hali ham _ClassName__AttributeName nomi ostida mavjud:

```
>>> b._B__private()
Bu shaxsiy usul
```

Meros deganda, bolalar sinfi ota-sinfning barcha atributlarini o'z ichiga oladi, shu bilan birga ularning ba'zilarini bekor qilinishi yoki bolaga qo'shilishi mumkin. Masalan, biz o'zimizning lug'atga o'xshash sinfimizni yaratishimiz mumkin:

```
>>> class Mydict(dict):
...     def get(self, key, default = 0):
...         return dict.get(self, key, default)
>>> a = dict(a=1, b=2)
>>> b = Mydict(a=1, b=2)
```

Mydict klassi xuddi lug'at kabi ishlaydi, bundan tashqari get usuli sukut bo'yicha None (yo'q) o'rniga 0 ni qaytaradi.

```
>>> b['c'] = 4
>>> print(b)
{'a': 1, 'c': 4, 'b': 2}
>>> print(a.get('v'))
None
>>> print(b.get('v'))
0
```

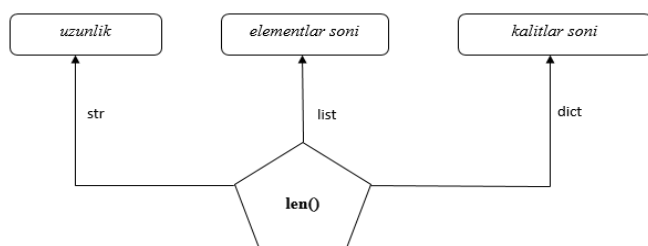
Polimorfizm dasturlashda juda muhim g'oyadir. U har xil foydalanish holatlarida har xil turlarni ifodalash uchun bitta obyektдан (usul, operator yoki obyekt) foydalanishdan iborat. Polimorfizm ko'p shakllarni anglatadi [6].

Python turli xil argumentlarni qabul qilishi mumkin bo'lgan ba'zi funktsiyalarga ega. Bunday funktsiyalardan biri len(). U har xil turdagi ma'lumotlarni qabul qilishi

mumkin. Keling, bu qanday ishlashiga misolni ko'rib chiqaylik.

```
>>> print(len("Python"))
6
>>> print(len(["Python", "Java", "C++"]))
3
>>> print(len({"Name": "Dilfuza", "Address": "Uzbeki-
stan"}))
2
```

Bu yerda biz len() funksiyasi bilan qator, ro'yxat, kortej, to'plam va lug'at kabi turli xil ma'lumotlar turlari ishlashi mumkinligini ko'rishimiz mumkin. Biroq, biz har bir ma'lumot turiga xos ma'lumotlarni qaytarishini ko'rishimiz mumkin (3-rasm).



3-rasm: len() funksiyasining polimorfizmi.

Polimorfizm - bu turli sinflarda bir xil usulning har xil xatti-harakatlari. Misol uchun, biz ikkita raqamni qo'shishimiz mumkin va ikkita satr qo'shishimiz mumkin. Bu holda biz boshqa natijaga erishamiz, chunki raqamlar va satrlar turli sinflardir.

Yuqorida biz Pythonning obyektga yo'naltirilgan dasturlashni qo'llab-quvvatlashining barcha asoslarini ko'rib chiqdik. Ma'lumotlar atributlari va moslashtirilgan usullar bilan oddiy sinflarni qanday yaratishni ko'rdik. Shuningdek, biz sinflarni qanday meros qilib olish va qo'shimcha ma'lumotlar atributlari va qo'shimcha usullarni qanday qo'shish va usullarni qanday qilib "amalga oshirish" mumkinligini ko'rib chiqdik. Agar biz sinfni meros qilib olsak, lekin quyi sinfimiz taqdim etadigan usullarni cheklamoqchi bo'lsak, uni amalga oshirish kerak bo'ladi, lekin undan ehtiyotkorlik bilan foydalanish kerak, chunki u pastki sinfdan uning asosiy sinflaridan biri ishlatilishi mumkin bo'lgan joyda foydalanish mumkin degan taxmini buzadi, ya'ni, polimorfizmni buzadi. Maxsus sinflar muammosiz birlashtirilishi mumkin, shunda ular Pythonning o'rnatilgan va kutubxona sinflari bilan bir xil sintaksislarni qo'llab-quvvatlaydi. Bunga maxsus usullarni qo'llash orqali erishiladi.

Adabiyotlar

1. Kölling, M., The Problem of Teaching Object-Oriented Programming, Part 1: Languages, Journal of Object-Oriented Programming, 11(8): 8-15, 1999.
2. Paul Barry. Head First Python. O'Reilly Media, 2010. -P. 494 . Series: Head First.
3. Python programming: 2 in 1. Theory and Practice by Techs Ed Academy, 2020.
4. Васильев А.Н. Python на примерах. Практический курс по программированию. - СПб.: Наука и Техника, 2016. - 432 с.
5. Сузи Р.А. Язык программирования Python. Курс лекций. - СПб: Питер. 2005. - 206 с.
6. Saidova D. E. Analysis of the Problems of the Teaching Object-Oriented Programming to Students //International Journal of Social Science Research and Review. - 2022. - T. 5. - №. 6. - C. 229-234.
7. Жураев Т. Информатика дарслари самарадорлигини оширишда замонавий педагогик технологиялардан фойдаланиш. НГПИ имени Ажинияза. // «Фан ва жамият». 2015. №. 1. -С.47-50.
8. <https://habr.com/ru> - Полиморфизм в Python.
9. <https://python.ivan-shamaev.ru> - Classes, methods Python 3. OOP examples.
10. <https://pythonworld.ru> - Python3 для начинающих: Инкапсуляция, наследование, полиморфизм.

REZYUME. Maqolada talabalarga obyektga yo'naltirilgan dasturlashni o'rgatish metodologiyasi tadqiq etilgan bo'lib, Python dasturlash tilining xususiyatlari va uning obyektga yo'naltirilgan dasturlashdagi afzalliklari hamda o'quv jarayonida qo'llaniladigan ta'lim metodlariga integratsiyalangan holatda tavsiflangan.

РЕЗЮМЕ. В статье исследуется методика обучения студентов объектно-ориентированному программированию, описываются особенности языка программирования Python и его преимущества в объектно-ориентированном программировании, а также его интеграция в методы обучения, используемые в процессе обучения.

SUMMARY. The article explores the methodology for teaching students object-oriented programming, describes the features of the Python programming language and its advantages in object-oriented programming, as well as its integration into teaching methods used in the learning process.

YOSHLARNI JISMONIY TARBIYA VA SPORT ORQALI SOG'LOMLASHTIRISHDA IJTIMOY FAOLLIK SIFATLARINING TAKOMILLASHTIRISH ZARURIYATI

A.Y.Saidova – o'qituvchi

E.A.Erkaev – o'qituvchi

Farg'ona politexnika instituti

Tayach so'zlari: ta'lim, tarbiya, barkamol avlod, integratsiya, shaxs, jismoniy tarbiya, ma'naviy-axloqiy tarbiya, harakatli o'yinlar, ma'naviy-axloqiy mezon, didaktik tamoyil.

Ключевые слова: образование, воспитание, гармонично развитое поколение, интеграция, личность, физическое воспитание, духовно-нравственное воспитание, подвижные игры, духовно-нравственные критерии, дидактический принцип.

Key words: education, upbringing, harmoniously developed generation, integration, personality, physical education, spiritual and moral education, action games, spiritual and moral criteria, didactic principle.

Mamlakat sportchisi - bu millat vakili, mavjud jamiyat kishisining axloqiy namunasidir. O'zbek sportchisining axloq normalari yuqori darajada shakllangan, fikrlash doirasi keng, ma'naviy boy, kirisuvchan, yaxshi tashkilotchi bo'la oladigan insonni tarbiyalash nazarda tutiladi. Bu mezonlar ayniqsa navqiron mamlakatimiz yoshlari uchun taalluqlidir. Sportchi axloqiy tarbiyasining xususiyatlari, uni jismoniy tarbiya jarayonida amalga oshirilishiga quyidagicha yo'nalish va shakl berilmoqda. Jamoatchilik burchiga ijtimoiy munosabatda bo'lish, faoliyatni o'z shaxsiy ishi bilan teng qo'yish, jamiyat va davlat ishini past darajada deb bilmaslik. O'zining jismoniy tarbiyasiga ongli munosabatda bo'lish, Vatani talabiga

doimo tayyor turish - bu har bir jamiyat a'zosining umumiy burchi hisoblanadi. Mamlakat sportchisini xalqaro maydondagi g'alabalarini hal qiluvchi omili, o'z xalqi, Vatani oldidagi javobgarlik burchini his qila olishida.

Mamlakatimizning yangi taraqqiyot bosqichida yoshlar ijtimoiy faolligini oshirishning institusional tizimini takomillashtirish zarurati yuzaga kelganligi va yoshlar ijtimoiy faolligini oshirish - aniq maqsadga qaratilgan tashkiliy-pedagogik jarayon ekanligi, unda yoshlarda shakllangan ijtimoiy faollik sifatlarining takomillashtirilishi va boyitib borilishi zarurati asoslanadi. Zero, «bugungi kunda mamlakatimizni yangilash va modernizatsiya qilish, uni innovatsion asosda rivojlantirish, o'z oldimizga qo'ygan