PYTHON DASTURLASH TILINI O‘QITISHDA FUNKSIYALARDAN FOYDALANISH METODIKASI

Nazarova Munojat Raxmonberdi qizi, Guliston davlat pedagogika instituti
“Ta’limda axborot texnologiyalari” 1-kurs Magistratura talabasi
e-mail: tmunojat471@gmail.com

Annotatsiya. Hozirgi zamonaviy axborot texnologiyalari keng taraqqiy etgan bir davrda umumta’lim maktab o‘quvchilariga dasturlash tillarini o‘qitishda alohida uslub bilan yondashish, soddalashtirilgan usullarni ishlab chiqish, o‘quvchilarni yoshini inobatga olgan holda vazifalar tayinlash dolzarb masalalardan biri hisoblanadi. Umumiy o‘rta ta’lim maktablarida o‘quvchi yoshlarda mustaqil dastur tuzish ko‘nikmalarini shakllantirish, dasturlash tillariga qiziqtirish, ularni kelajakda o‘z o‘rinini topishiga va mamlakat rivoji uchun mukammal dasturlar yaratishiga turtki bo‘ladi.

Kalit so‘zlar: dastur, ta’lim metodlari, dasturlash muhiti, funksiyalar, protsedura, komponent, ilova, ma’lumotlar bazasi.

LEARNING THE PYTHON PROGRAMMING LANGUAGE METHODOLOGY OF USING FUNCTIONS

Abstract. Nowadays when modern information technologies are widely developed, one of the most actual problems is to issue a special approach for teaching programming languages to secondary school students and to pay attention to the development of simplified methods assigning tasks based on the age of students. This article develops self-programming skills among high school students, helps to develop an interest in programming and help them find their place in the future, as well as create better programs for the development of the country.

Keywords: program, educational methods, programming environment, functions, procedure, component, application, database.

ИЗУЧЕНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ PYTHON МЕТОДОЛОГИЯ ИСПОЛЬЗОВАНИЯ ФУНКЦИЙ

Аннотация. В настоящее время, когда современные информационные технологии широко развиты, одной из наиболее актуальных проблем является разработка специального подхода к обучению учащихся средних школ языкам программирования и уделение внимания разработке упрощенных методов постановки задач с учетом возраста учащихся. Данная статья развивает у

старшеклассников навыки самопрограммирования, помогает развить интерес к программированию и помогает им найти свое место в будущем, а также создавать более качественные программы для развития страны.

Ключевые слова: программа, методы обучения, среда программирования, функции, процедура, компонент, приложение, база данных.

***Kirish* (*Introduction*).**

Bugungi kunda deyarli har bir sohada axborot texnologiyalaridan samarali foydalanilayotganini ko‘rishimiz mumkin. Bu esa o‘quvchi yoshlarda dasturlash va axborot texnologiyalaridan foydalanish ko‘nikmasinini rivojlantirish naqadar muhim ekanligini ko‘rsatib bermoqda. Bugungi kunda rivojlangan mamlakatlarida texnologik bazada ta’lim olish mumkin. Aqlli maktablar virtual ta’lim sohasida sakrashga erishdilar. Onlayn o‘qitish va masofadan o‘qitish yangi asrdagi yangi ta’lim shakllari qatoriga kiradi. 21-asrning boshlarida o‘quv muhitini rivojlantirish orqali shaxslar va jamiyatlar ta’limga bo‘lgan ehtiyojning ortishi bilan ta’lim muassasalari va ularning an'anaviy tuzilmalari zimmasiga katta mas’uliyat yukladilar. Bugungi kunda turli xil axborot va kommunikatsiya texnologiyalari ta’lim va o‘quv jarayonini engillashtirish qobiliyatiga ega.

***Mavzuga oid adabiyotlar tahlili* (*Literature review*).**

DASTURLASH TILI - ma’lumotlar (axborotlar)ni va ularni elektron mashinalarda ishlash algoritmlari (dasturlari)ni bayon qilish uchun mo‘ljallangan rasmiy til; odamning elektron mashina bilan bog‘lanishi uchun ishlatiladigan shartli belgilar tizimi. Dasturlash tili tabiiy til bilan shartli mashina tili orasidagi o‘rinni egallaydi. Dasturlash tilining asosiy vazifasi dasturlar tuzish (Dasturlash) vositasi bo‘lishi, ya’ni ma’lumotlarni ifodalash va ularni EHM (kompyuter) da ishlashda qo‘llanishidan iborat.

Tizimli va amaliy dasturlarni yaratish uchun dasturlash tillaridan foydalaniladi. Dastur yaratish jarayoni katta hajmga ega bólib, dasturlash tillarida Tizimli va amaliy dasturlarni yaratish uchun dasturlash tillaridan foydalaniladi. Dastur yaratish jarayoni katta hajmga ega bo‘lib, dasturlash tillarida dastur yozish shu jarayonning bir qismi sanaladi.

Python ([ˈpʌɪθ(ə)n] – *payton*, *piton*) – turli sohalar uchun yuqori darajadagi umumiy maqsadli dasturlash tili. Uning til konstruksiyalari va obyektga yo‘naltirilgan yondashuvi dasturchilarga kichik va yirik loyihalar uchun aniq, mantiqiy kod yozishda yordam berishga qaratilgan. Shuningdek Python sun’iy intellekt hamda ma’lumotlar tahlili sohalarining tili hisoblanadi. Python deyarli barcha platformalarda ishlay oladi, xususan Windows, Linux, Mac OS X, Palm OS, Mac OS va boshqalar shular jumlasidandir.

***Tadqiqot metodologiyasi* (*Research Methodology*).**

Ma’lum vazifani bajaruvchi tegishli buyruqlar guruhiga **funksiya** deyiladi. Funksiyalar dasturimizni kichik va modulli qismlarga arjratishga yordam beradi. Dasturimiz kattalashgani sari, funksiyalar uni shuncha boshqariluvchan va tartibli

qiladi. Bundan tashqari, u kodni takrorlanishini oldini oladi hamda kodni qayta ishlatish imkonini beradi.

Aslida har qanday funksiyaning ortida ham bir necha qatordan iborat kod bo‘ladi, lekin biz funksiyaga murojat qilganda uning nomini yozamiz xolos. Funksiya ortidagi kod esa biz uchun yashirin bo‘lib qolaveradi. Funksiyalarning qulayligi ham shunda. Dastur davomida ma’lum bir kodlarni qayta-qayta yozmaslik uchun biz ularni jamlab, bitta funksiya ichiga joylashimiz va dastur davomida bu kodlarga funksiya nomi orqali murojat qilishimiz mumkin.

Funksiyalar turlicha bo‘ladi, ba’zi funksiyalar sizdan qiymat qabul qilib, konsolga biror ma’umot chiqaradi, ba’zilari esa sizdan qabul qilgan qiymat ustida turli amallar bajarib yangi qiymat qaytaradi. Foydalanuvchidan mutlaqo qiymat qabul qilmaydigan funksiyalar ham mavjud.

Ushbu tadqiqot metodologiyasida siz qanday qilib Pythonda yangi funksiya yaratish, unga murojaat qilish, tekshirish va to‘g‘rilashni o‘rganasiz. Shuningdek tadqiqot yakunida dasturimizni bir nechta fayllarga ajratishni va funksiylarni alohida, module deb ataluvchi fayllarga joylashni ham o‘rganamiz.

***Tahlil va natijalar* (*Analysis and results*).**

Odatda, funksiyalar ikki turga bo‘linadi, ular:

1. Standart funksiyalar – Python dasturlash tilidagi oldindan mavjud bo‘lgan funksiyalar.
2. Qo‘lbola funksiyalar – foydalanuvchilarning o‘zlari tomonidan belgilangan funksiyalar.

Funksiya Yaratamiz

Keling oddiy, **`salom_ber`** deb nomlangan funksiya yaratamiz. Bu funksiya murojat qilganimizda konsolga **"Assalom alaykum!"** degan xabarni chiqarsin.

```
def salom_ber():
"""Salom beruvchi funksiya"""
print(“Assalomu alaykum!”)
```

Kodni qatroma-qator tahlil qilaylik:

1. Avvalo **`def`** operatori yordamida Pythonga funksiya yaratayotganimizni bildirdik. **`def`** dan so‘ng esa funksiyamizga nom berdik va qavslarni ochib, yopdik. Bizning funksiyamiz foydalanuvchidan hech qanday qiymat qabul qilmaydi, shuning uchun ham qavs ichi bo‘sh. Keyingi misollarda foydalanuvchidan qiymat qabul qiluvchi funksiyalarni ham ko‘ramiz.
2. **`def`** qatoridan keyin o‘ngga surib yozilgan har qanday kod funksiyaning badani hisoblanadi. 2-qatorda biz uchta ketma-ket qo‘shtirnoq ichida funksiya haqida ma’lumot berdik. Python mana shu ma’lumotni o‘qib, dasturchi funksiya haqida bilmoqchi bo‘lganda aynan shu matnni ko‘rsatadi.
3. Oxirgi qatorimizda esa **`"Assalomu alaykum!"`** matnini konsolga chiqarishni buyurdik. Bizning sodda funksiyamizning asosiy vazifasi ham shu.

Mana funksiya tayyor. Endi bu funksiyadan foydalanish uchun uni *chaqiramiz*. Buning uchun funksiya nomini yozamiz va qavslarni ochib, yopamiz (esingizda

bo‘lsa bizning funksiyamiz qiymat qabul qilmaydi, shuning uchun qavslar ichi bo‘sh).

salom_ber()

Funksiyaga Qiymat Uzatish

Avvalgi sodda funksiyamiz foydalanivchidan hech qanday qiymat olmaydi va barchaga birday “Assalomu aleykum” deb javob qiladi. Keling funksiyaga o‘zgartirish kiritamiz, funksiya foydalanuvchi ismini qabul qilib, unga ismi bilan murojat qilsin. Buning uchun funksiya nomidan keyin, qavs ichida foydalanuvchi berishi kerak bo‘lgan qiymatni ko‘rsatamiz.

```
def salom_ber(ism):
"""Foydalanuvchi ismini qabul qilib, unga salom beruvchi funksiya"""
print(f "Assalomu alaykum. Hurmatli{ism.title( )}!")
salom_ber(Ali)
```

Mana endi funksiyamiz foydalanuvchidan ism degan qiymatni ham kutadi.

Agar funksiyaga murojaat qilishda, unga qiymat bermasak xatolik vujudga keladi.

Argument va Parametr

Funksiya yaratishda, qavs ichida berilgan, funksiya to‘g‘ri ishlashi uchun uzatiladigan qiymat *parameter* deb ataladi. Yuqoridagi misolda ism bu **salom_ber** funksiyasining parametri.

Foydalanuvchi funksiyaga murojat qilishda funksiyaga uzatgan qiymat esa **argument** deb atalad.Ba’zi manbalarda yoki darslarda argument va parametr so‘zlari almashtirib ishlatilishi ham kuzatiladi.

Funksiyaga bir nechta argument uzatish

Shunday funksiyalar bor, bir emas bir nechta parameter talab qilishi mumkin, foydalanuvchi esa o‘z navbatida bir nechta argumentlar taqdim qilishi kerak.

Yuqoridagi funksiya to‘g‘ri natija chiqarishi uchun argumentlarni ism va familiya ketma-ketligida kiritishimiz lozim.

```
def toliq_ism(ism, familiya):
    print(f"Foydalanuvchi ismi:{ism.title()}! \n"
    f"Foydalanuvchi familiyasi:{familiya.title()}!")
toliq_ism('Olim','Hakimov')
```

Qiymat qaytarish

Avvalgi dasturlarda yaratgan barcha funksiyalarimiz konsolga ma’lumot chiqarayotgan edi. Aslida, aksar holatlarda bu g‘ayritabiiy. Sababi, dasturchi sifatida biz konsolga chiqqan ma’lumotdan unumli foydalana olmaymiz. Konsoldagi qiymatni o‘zgaruvchiga yuklab, undan kelajakda foydalanib ham bo‘lmaydi. Mana shunday holatlarda, funksiyadan unumli foydalanish uchun undan biror qiymatni qaytarish maqsadga muvofiq bo‘ladi.

```
def tubson(son1,son2):
    al=[]
```

```
    s=0
    for i in range(son1,son2+1):
        for j in range(2,son2+1):
            if i%j==0:
                s+=1
        if s<2:
            al.append(i)
        s=0
    return al

b=int(input('1 chi sonni kiriting‘))
c=int(input('2 chi sonni kiriting‘))

h=tubson(b,c)
print(f"{b} va {c} sonlari orasidagi tub sonlar:{h}")
```

Hujjat satrlari (docstring)

Funksiyaning bosh qismidan keyin yoziladigan birinchi satr hujjat satri deyiladi, uning qisqartmasi docstring. U funksiya nima vazifa bajarishi haqida qisqacha axborot berish uchun ishlatiladi. Ixtiyoriy bo‘lsa ham lekin hujjat qilib yozib qo‘yish dasturlashda yaxshi amaliyot hisoblanadi.

Odatda hujjat satrlari uchtalik apostroflar bilan yoziladi, sababi hujjat satri 2 yoki undan ortiq qatorli ma’lumotdan iborat bo‘lishi mumkin. Bu holatda esa uchtalik apostrofni ishlatish eng yaxshi yo‘l hisoblanadi. Funksiyamizga yozgan hujjat satrimizni funksiyamizning __doc__ atributi orqali olishimiz mumkin

```
def square(x):
  """Bu funksiya bizga berilgan sonning kvadratini hisoblab beradi"""
      return x*x
print(square.__doc__)
```

***Xulosa va takliflar* (*Conclusion/Recommendations*).**

Ushbu mavzu tinglovchilarda dasturlashga bo‘lgan ishtiyoqni talab qilib, funksiyalardan to‘g‘ri foydalanish ko‘nikmalarini shakllantiradi. Egallagan bilimlarini mustaqil baholash va o‘tilgan mavzuga oid tushuncalarni amalda qo‘llashga imkon yaratadi. Turli funksiyalarni qo‘llanilishi bo‘yicha g‘oyalarni ifodalash va ular orasidagi bog‘liqlikni topish malakasini shakllantiradi.

Shu o‘rinda o‘quvchilarning dasturlashga bo‘lgan ishtiyoqlarini oshirish, bugungi kunning dolzarb masalasi bo‘lib qolmoqda. Umumiy o‘rta maktabda zamonaviy dasturlash tillarini o‘qitilishi o‘quvchilarda dasturlash muhitida ishlash va mustaqil dastur tuzish, mamlakatimizning turli sohalari uchun ilovalar yaratish, zamonaviy axborot texnologiyalari vositalaridan foydalanish ko‘nikmalarini shakllantirishga va ularning amaliy ishlarida yangi qirralarning ochilishiga zamin hozirlaydi.

Foydalanilgan adabiyotlar

1. https://www.codecademy.com – Python tilida programmalash bo‘yicha namunalar izoxlari bilan keltirilgan.
2. https://python.sariq.dev – Python tilida programmalash bo‘yicha namunalar izoxlari bilan keltirilgan.
3. https://www.w3schools.com – Python tilida programmalash bo‘yicha namunalar izoxlari bilan keltirilgan.
4. https://www.learnpython.org/ – Python tilida programmalash bo‘yicha namunalar izoxlari bilan keltirilgan.
5. https://cyberleninka.ru/ – Python dasturlash tillari, Django texnologiyasi bo‘yicha ma’ruzalar, dasturlar namunasini olish mumkin.
6. И. Ш. Садуллаев, Н. Н. Зарипов “Персональная учебная среда учащегося в режиме дистанционного обучения” - Международна научна школа «Парадигма». Лято, 2015.
7. Н. Н. Зарипов. “Kompyuter grafikasi yordamida talabalarning fazoviy tafakkurini rivojlantirish” // Образование и инновационные исследования https://doi.org/10.53885/edinres.2022.10.10.015b.