

MODUL TUSHUNCHASI, STANDART KUTUBXONALAR, SYS VA COPY MODULI

*Denov tuman maktabgacha va maktab
ta'limi tashkilotiga qarashli 51-umumiy
o'rta ta'lim maktabi Informatika fani o'qituvchisi*
XOLTURAYEV ZUHRIDDIN

Annotatsiya: Ushbu maqola Python dasturlash tilida modul tushunchasi, standart kutubxonalar, xususan, sys va copy modullarining ahamiyatini yoritadi. Modullarning turlari, ularni dasturga ulash usullari va standart kutubxonalarning roli batafsil tushuntirilgan. sys moduli orqali Python interpretatori muhitiga kirish va buyruq qatori argumentlari bilan ishlash, copy moduli orqali esa obyektlarni nusxalash usullari misollar yordamida ko'rsatilgan. Ushbu maqola Python dasturlash tilida modul tushunchasini o'zlashtirish va standart kutubxonalardan samarali foydalanishni istagan dasturchilar uchun mo'ljallangan.

Kalit so'zlar: python, modul, paket, standart kutubxona, sys, copy, import, sayoz nusxa, chuqur nusxa, dasturlash

Python dasturlash tili zamonaviy dasturlash olamida o'zining soddaligi, o'qilishi osonligi va keng qo'llanilishi bilan alohida o'ringa ega. Pythonning kuchli tomonlaridan biri uning boy kutubxonasi va ma'lumotlar tuzilmalaridir. Ushbu bob sizni Python dasturlash tilidagi ma'lumotlar tuzilmalari va standart modullar bilan tanishtiradi. Siz ro'yxatlar, kortejlar, lug'atlar va to'plamlar kabi asosiy ma'lumotlar tuzilmalarini yaratish, ular ustida amallar bajarish, shuningdek, Pythonning standart modullaridan samarali foydalanishni o'rganasiz. Ushbu bobni o'zlashtirish sizga Python dasturlash tilida murakkab masalalarni yechishda va professional darajada kod yozishda yordam beradi.

Standart kutubxona modullarini o'rganishdan oldin Pythonda modul tushunchasiga aniqlik kiritib olish lozim. Python tilida bir xil vazifani bajaruvchi modullar yig'indisini bitta paketga joylashtirish mumkin. Shunday paketlardan biri sifatida XML paketini misol qilib keltirish mumkin. Ushbu paket XML ning har xil aspektlarini qayta ishlashga mo'ljallangan modullardan tashkil topgan.

Python tilda dastur tuzishda modul atributlari modulda aniqlangan nomlar bo'lgan obyekt modul sifatida taqdim etiladi.

```
>>> import datetime
>>> d1=datetime.date(2017,11,20)
..
```

Bu misolda datetime moduli import qilinayapti. Import operatorining ishi natijasida mazkur nomlar kengligida datetime nomi bilan obyekt paydo bo'lyapti.

Python tilida modullar oddiy (Pythonda yozilgan) va kengaytiriladigan ya'ni boshqa tilda yoziladigan masalan Python interpretatori yozilgan C dasturlash tilida yozilgan modullarga bo'linadi. Foydalanuvchi nuqtai nazarida ular ishlash tezligi bilan farq qiladi. Standart kutubxonada modul 2 xil variantda bo'ladi: Pythonda yozilgan yoki C da. Bunga misol sifatida *pickle* va *cpickle* modullarini keltirish mumkin. Odatda Pythonda oddiy modullar kengaytirilgan modullarga nisbatan ishlatishda qulay hisoblanadi.

Modul funksiyalaridan foydalanish uchun uni boshqa dasturdan yuklash (импортировать) mumkin. Dastlab standart kutubxonalar modullarini qanday ishlatishni ko'raylik.

```

1  import sys
2
3  print("Buyruqlar qatori argumentlari:")
4
5  for i in sys.argv:
6
7      print(i)
8
9  print("\n\n O'zgaruvchi PYTHONPATH qiymati", sys.path, "\n")

```

Natija:

```

1. $ Python3 using_sys.py biz argumentlarmiz
2. Buyruqlar qatori argumentlari:
3. using_sys.py
4. biz
5. argumentlarmiz
6. O'zgaruvchi PYTHONPATH qiymati ['/home/user/python darslari/7-
dars', '/usr/lib/python3.4', '/usr/lib/python3.4/plat-x86_64-
linux-gnu', '/usr/lib/python3.4/lib-dynload',
'/usr/local/lib/python3.4/dist-packages', '/usr/lib/python3/dist-
packages']

```

Bu misolda dastlab **sys** moduli **import** buyrug'i yordamida yuklanyapti. **sys** moduli Python interpretatoriga va uning muxitiga ya'ni tizimiga (system) tegishli funksiyalardan tashkil topgan. Python **import sys** buyrug'ini bajarayotganda **sys** modulini qidiradi. Bu holatda **sys** standart modullardan biri bo'lganligi uchun, Python uni qayerdan izlash kerakligini biladi. Agar bu oddiy modul, ya'ni Pythonda yozilgan modul bo'lganida edi, u holda Python uni **sys.path** ko'rsatilgan kataloglardan izlagan bo'lar edi. Agar modul topilsa, undagi buyruqlar bajariladi va bu modul foydalanishga (доступным) shay holatga keladi. **sys** modulidagi **argv** o'zgaruvchisiga murojat qilish nuqta orqali amalga oshiriladi ya'ni **sys.argv**. Bunday ifodalashning afzalligi dasturda ishlatilishi mumkin bo'lgan **argv** o'zgaruvchisi bilan xatoliklar yuz bermaydi. **sys.argv** qatorlar ro'yxati hisoblanadi. U buyruqlar qatori argumentlaridan ya'ni buyruqlar qatoridan dasturga uzatilgan argumentlardan tashkil topgan.

Python dasturida modullarni ulash *import* operatori orqali amalga oshirilishi yuqoridagi misolda ko'rdik. Modullarni ulashini ham 2 xil shakli mavjud: birinchisi *import* operatori orqali bo'lsa, ikkinchisi *from-import* operatori orqalidir. **From... import ... operatori**- argv o'zgaruvchisini dasturga to'g'ridan-to'g'ri yuklash uchun hamda har doim sys.argv deb yozmaslik uchun, from sys import argv ifodasidan foydalanish mumkin. sys modulida ishlatiladigan hamma nomlarni yuklash uchun "from sys import *" buyrug'ini bajarish mumkin.

```
1. from math import *
2.
3. n = int(input('Biror son kiriting: '))
4.
5. print(sqrt(n))
```

Standart kutubxonalar

Python tili standart kutubxonasining modullarini shartli ravishda mavzular bo'yicha quyidagi guruhlariga ajratish mumkin:

1. Bajarish davri servislari. Modular: sys, atexit, copy, traceback, math, cmath, random, time, calendar, datetime, sets, array, struct, intertools, locale, gettext.
2. Siklni qayta ishlashni qo'llab-quvvatlovchi. Modular: pdb, hotshot, profile, unittest, pydoc. Paketlar: docutils, distutils.
3. OS (fayllar, protseslar) bilan ishlash. Modular: os, os.path, getopt, glob, popen2, shutil, select, signal, stat, tempfile.
4. Matnlarni qayta ishlovchi. Modular: string, re, StringIO, codecs, difflib, mmap, sgmlib, htmlib, htmlentitydefs. Paket: xml.
5. Ko'p oqimli hisoblashlar. Modular: threading, thread, Queue.
6. Ma'lumotlarni saqlash. Arxivlash. Modular: pickle, shelve, anydbm, gdbm, gzip, zlib, zipfile, bz2, csv, tarfile.
7. Platformaga tobe modullar. UNIX uchun: commands, pwd, grp, fcntl, resource, termios, readline, rlcompleter. Windows uchun: msvcrt, _winreg, winsound.
8. Tarmoqni qo'llab-quvvatlash. Internet protokollari. Modular: cgi, Cookie, urllib, urlparse, httplib, smtplib, poplib, telnetlib, socket, asyncore. Serverlarga misollar: SocketServer, BaseHTTPServer, xmlrpclib, asynchat.
9. Internetni qo'llab-quvvatlash. Ma'lumotlar formatlari. Modular: quopri, uu, base64, binhex, binascii, rfc822, mimetools, MimeWriter, multifile, mailbox. Paket: email.
10. Python uchun. Modular: parser, symbol, token, keyword, inspect, tokenize, pycbr, py_compile, compileall, dis, compiler.
11. Grafik interfeys. Modul: Tkinter.

Ko'pincha modullar o'zida bir yoki bir nechta sinflarni saqlaydilar. Bu sinflar yordamida kerakli tipdagi obyekt yaratiladi, lekin gap modulidagi nomlar haqida emas, aksincha shu obtekt atributi haqida boradi. Bir nechta modullar faqat erkin obyektlar ustida ishlash uchun umumiy bo'lgan funksiyalardan iborat bo'ladilar.

Sys moduli

Sys moduli Python interpretatorida dasturni bajaruvchi muhitdir. Quyida bu modulni eng ko'p qo'llaniladigan obyektlari keltirilgan:

Exit([c])- dasturdan chiqish. Tugatishning raqamli kodini yuborish mumkin: agarda dasturni tugatish muvaffaqiyatli amalga ohsa 0 ni yuboradi, aksincha bo'lsa ya'ni xatolik yuz bersa boshqa raqamlarni yuboradi.

Argv- buyruqlar qatori argumentlari ro'yxati. Oddiy holatda sys.argv[0] buyruqlar qatoriga ishga tushirilgan dastur nomini va boshqa parametrlar yuboriladi.

Platform- interpretator ishlaydigan platforma.

Stdin, stdout, stderr- standart kiritish, chiqarish, xalolarni chiqarish. Ochiq faylli obyektlar.

Version- interpretator versiyasi.

Sereursionlimit(limit)- rekursiv chaqirishlarni maksimal kiritish darajasini o'rnatadi.

Exc_info()-kiritish-chiqarish istisnosi haqida ma'lumot.

Copy moduli

Bu modul obyektlarni nusxalashga mo'ljallangan funksiyalarga ega. Boshida Pyhtonda sal sarosimaga solish uchun "paradoks" ni ko'rib chiqish tavsiya etiladi.

```
lst1 = [0, 0, 0]
lst = [lst1] * 3
print(lst)
lst[0][1] = 1
print (lst)
```

Va biz kutmagan natija paydo bo'ladi:

```
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
[[0, 1, 0], [0, 1, 0], [0, 1, 0]]
```

Gap shundaki bu yerda lst ro'yxati shu ro'yxatning izohiga ega. Agarda rostdan ham ro'yxatni ko'paytirmoqchi bo'lsak, copy modulidagi copy() funksiyasini qo'llash kerak.

```
from copy import copy
lst1 = [0, 0, 0]
lst = [copy(lst1) for i in range(3)]
print (lst)
lst[0][1] = 1
print (lst)
```

Endi kutilgan natija paydo bo'ladi:

```
[[0, 0, 0], [0, 0, 0], [0, 0, 0]]
[[0, 1, 0], [0, 0, 0], [0, 0, 0]]
```

Copy modulida yuqori aniqlikda nusxalash uchun deepcopy() funksiyasi bor bu funksiya yordamida obektlar butun imkoniyati bilan rekursiv nusxalanadi.

Math, cmath, random va os moduli.

Math va cmath modullarida haqiqiy va kompleksli argumentlar uchun matematik funksiyalar to'plangan. Bu C tilida foydalaniladigan funksiyalar. Quyida math modulining funksiyalari keltirilgan. Qayerda z harfi bilan argumentga belgilash kiritilgan bo'lsa, u cmath modulidagi analogik funksiya ham shunday belgilanishini bildiradi.

Acos(z)- arkkosinus z.

Asin(z)- arksinus z.

Atan(z)- arktangens z.

Atan2(y, x)- atan(y/x).

Ceil(x)- x ga teng yoki katta eng kichik butun son.

Cos(z)- kosinus z.

Cosh(x)- giperbolik x kosinusi.

e- e konstantasi.

Exp(z)- eksponenta (bu degani e^{**z})

Fabs(x)- x absolute raqami.

Floor(x)- x ga teng yoki kichik eng katta butun son

Fmod(x,y)- x ni y ga bo'lgandagi qoldiq qism.

Frexp(x)- mantisa va tartibni (m, i) juftligi kabi qaytaradi, m- o'zgaruvchan nuqtali son, i esa- $x=m*2^{**i}$ ga teng butun son bo'ladi. Agarda $0-(0,0)$ qaytarsa boshqa paytda $0.5 \leq \text{abs}(m) < 1.0$ bo'ladi.

Factorial(x)- x ning faktoriali. $N!=1*2*3*\dots*n$

Hypot(x,y)- $\text{sqrt}(x*x+y*y)$

Ldexp(m,i)- $m*(2^{**i})$.

Log(z)- natural logarifm z.

Log10(z)- o'nlik logarifm z.

Log2(z)-logarifm ikki asosga ko'ra z.

Modf(x)- (y,q) juftlikda x ning butun va kasr qismini qaytaradi.

p-pi konstantasi.

Pow(x,y)- x^{**y} .

Sin(z)- z ning sinusi.

Sinh(z)- z ning giperbolik sinusi.

Sqrt(z)- z ning kvadrat ildizi.

Tan(z)- z ning tangensi.

Tanh(z)- z ning giperbolik tangensi.

Trunc(x)- x haqiqiy sonning butun qismini qaytaradi.

degrees(x)-x ni radiandan gradusga o'tkazish.

radians(x)- x ni gradusdan radianga o'tkazish.

```
>>> import math
>>> math.factorial(10)
3628800
>>> math.hypot(3,4)
5.0
>>> math.ceil(10.8)
11
>>> math.ceil(10.2)
11
>>> math.log10(100)
2.0
>>> math.log2(64)
6.0
>>> math.trunc(10.6)
10
```

Random moduli

Bu modul har xil taqsimotlar uchun tasodifiy raqamlarni generatsiya qiladi. Eng ko'p qo'llaniladigan funksiyalari:

Random()-[0.0, 1.0) yarim ochiq diapozondagi tasodifiy sonlarni generatsiya qiladi.

Choice(s)- s ketma- ketlikdan tasodifiy elementni tanlab oladi.

Shuffle(s)- s o'zgaruvchan ketma-ketlik elementlarini joyiga joylashtiradi.

Randrange([start], stop, [step])- renga(start, stop, step) diapozondagi tasodifiy butun raqamni chiqaradi. Choice(range(start, stop, step)) ga analogik holatda.

Normalvariate(mu, sigma)- normal holatda taqsimlangan ketma-ketlikdan raqamni chiqaradi. Bu yerda mu- o'rtacha, sigma-o'rta kvadratli ($\sigma > 0$) sonlar.

Boshqa funksiyalar va uning parametrlarini hujjatlashdan aniqlab olish mumkin. Modulda qandaydir holatga tasodifiy raqamlar generatorini joylashtirishga imkon beruvchi seed(n) funksiyasi ham mavjud. Masalan: agarda bitta tasodifiy raqamlar ketma-ketligidan ko'p marta foydalanishga ehtiyoj sezilsa.

Time va Sets moduli

Time moduli joriy vaqtni olish uchun va vaqt formatlarini o'zgartirish uchun fuksiyalarni taqdim etadi.

Sets moduli to'plamlar uchun ko'rsatgichlar tipini amalga oshiradi. Quyidagi misol bu moduldan qanday foydalanish mumkinligini ko'rsatadi. Su rinda bilishimiz kerakki, Python 2.4 va undan yuqori versiyalarda set tipi sets o'rniga kiritilgan.

```
import sets
A = sets.Set([1, 2, 3])
B = sets.Set([2, 3, 4])
print (A | B, A & B, A - B, A ^ B)
```



```
for i in A:
    if I in B: print (i,)
```

Natijada:

```
Set([1, 2, 3, 4]) Set([2, 3]) Set([1]) Set([1, 4])
2 3
```

Array va struct modullari

Bu modullar past darajali massiv va korsatgichlar tuzilmasini amalgam oshiradi. Ularning asosiy vazifasi- ko'rsatgichlarning ikkilamchi formatlarini ko'rib chiqish.

Xulosa

Modullar Python-da kodni tashkillashtirish va qayta ishlatish uchun muhim vositadir. Python-ning standart kutubxonasi turli xil vazifalarni bajarish uchun mo'ljallangan ko'plab modullarni o'z ichiga oladi. sys moduli Python interpretatori bilan o'zaro aloqa qilish uchun, copy moduli esa obyektlarni nusxalash uchun ishlatiladi. Ushbu modullarni tushunish va ulardan foydalanish Python dasturlash ko'nikmalarini oshirishga yordam beradi. "Pythonda ma'lumotlar tuzilmasi va tilning standart modullari" deb atalib bunda ro'yxat, kortej, lug'at va to'plam tushunchalari va ularni qanday yaratish mumkinligi ular ustida amallar bajarish haqida ma'lumotlar keltirilib misollar yordamida tushuntirilgan. Shuningdek, bu bobda modul tushunchasi, Python dasturining juda boy kutubxonaga ega ekanligi haqidagi ma'lumotlar berilgan bo'lib, ko'plab modullar shu jumladan sys, copy, time, math, cmath, random, os modullari ularning funksiyalari va qo'llanilishi haqida yozilib, misollar keltirish yordamida tushuntirilgan.

Foydalanilgan adabiyotlar:

1. Федеральный государственный образовательный стандарт основного общего образования [Текст] / М-во образования и науки Рос.Федерации. —М.: Просвещение, 2011. - 48 с.
2. Семакин И.Г. Информатика. Базовый курс. 7-9 классы / И.Г.Семакин и др.— 2-е изд., испр. и доп. — М.: БИНОМ. Лаборатория знаний, 2004.
3. Бауэр Ф.Л., Гооз Г. Информатика. Вводный курс: Пер. с нем. / Под ред.
4. А. П. Ершова. — М.: Мир, 1976.
5. Дородницын А. А. Информатика: предмет и задачи // Кибернетика. Становление информатики. — М.: Наука, 1986.
6. "Экономика и социум" №2(93)-2 2022 www.iupr.ru 449
7. Ершов А. П., Звенигородский Г. А., Первин Ю.А. Школьная информатика (концепции, состояние, перспективы). — Новосибирск, ВЦ СО АН СССР, 1979.