

**PYTHON DASTURINING O`ZIGA XOS JIHLARI**

*Samarqand Davlat Universiteti Urgut filiali*

*Biznesni boshqarish va tabiiy fanlar fakulteti*

*220-guruh talabasi:*

***Berdimurodov Ibrohim Bobomurod o'g'li***

*Pochta: [ibrohimberdimurodov5@gmail.com](mailto:ibrohimberdimurodov5@gmail.com)*

*O'qituvchi: assistant, Ahtamqulov Muhriddin*

*Pochta: [muhriddinahtamqulov@gmail.com](mailto:muhriddinahtamqulov@gmail.com)*

***Annotatsiya:*** *Ushbu maqolada pythonning standart kutubxonasi, dasturlarning o'ziga xos jihatlari muhokama etiladi.*

***Kalit so'zlar:*** *Python, kompyuter, metod, formula.*

**KIRISH**

XX asrning 80 - yillaridan oldin ishlab chiqilgan kompyuterlar uchun katta dasturiy sistemalarni ishlab chiqish juda ham mushkul vazifa edi. Buning eng katta sababi shu davrga xos bo'lgan kompyuterlarning imoniyatlarining chegaralanganidir. Dasturiy komplekslarni ishlab chiqishda asosiy cheklanishlar kompyuter tezkor xotirasining sig'imi, ma'lumotlarni ikkilamchi xotira qurilmalaridan (magnit lentalar, barabanlar va x.k.) o'qish tezligi, prostessorning ishlash tezligi (ularning takt chastotalari bir necha yuz mikrosekund bo'lgan) bilan bog'liq. Bu davrdagi kompyuterlar xalq ho'jaligining hisob-kitob bilan bog'liq bo'lgan masalalarini yechish uchun mo'ljallangan edi. Kompyuterlarda boshqa xarakterdagi masalalarning yechishning iloji yo'q edi.

**ADABIYOTLAR SHARHI**

Dasturchilar uchun shu kompyuterlarda yechilayotgan masala uchun berilgan ma'lumotlarni hisobga olgan holda masalaning yechish algoritmini

ishlab chiqish birinchi o'rindagi vazifa hisoblangan. Bu borada N. Vitrning mashhur ma'lumotlar+algoritm=dastur formulasini yodga olish yetarli. 80-yillardan keyin ishlab chiqilgan kompyuterlarning imkoniyatlari kengayib, ishlab chiqarish narxi keskin pasaygani tufayli ulardan foydalanish samarasi ortib bordi. Natijada, dastlab matn va grafiklar uchun muharrirlari ishlab chiqildi, keyinchalik multimedia qurilmasi yordamida tovushli ma'lumotlarni qayta ishlash imkoniyatlari ham paydo bo'ldi. Kompyuter tasviriy imkoniyatlarining kengayishi foydalanuvchilar uchun qulay bo'lgan grafik muloqot interfeysini yaratilishiga sabab bo'ldi. Natijada yuqori murakkablikdagi katta amaliy dasturlarni yaratish foydali hamda zarur bo'lib qoldi.

### **TADQIQOT METODOLOGIYASI VA EMPIRIK TAHLIL**

Pythonning standart kutubxonasi, Python tilida yaratilgan va kutubxona nomi bilan topilgan modullar to'plami hisoblanadi. Bu modullar, Pythonning o'zida ishlatiladigan asosiy funksiyalarni, obyektlarni va klasslarni o'z ichiga oladi. Pythonni o'rnatgandan so'ng, standart kutubxonaning modullari avtomatik ravishda o'rnatiladi va dasturlash jarayonida o'zgarishsiz ishlatiladi.

Quyidagi Pythonning standart kutubxona modullaridan ba'zilari:

1. **\*\*os\*\***: Operatsion tizim bilan ishlash uchun funksiyalar (fayllar, kataloglar, yo'llar bilan ishlash).
2. **\*\*sys\*\***: Python interpretatorining qo'llanilish parametrlari va funksiyalari.
3. **\*\*math\*\***: Matematik amallar (ildizlar, logarifmlar, trigonometriya funksiyalari).
4. **\*\*random\*\***: Tasodifiy sonlar va tasodifiy elementlarni qayta-tartiblash uchun funksiyalar.
5. **\*\*datetime\*\***: Sana va vaqt obyektlari va amallari.
6. **\*\*re\*\***: Regulyar ifodalar bilan ishlash.

7. **\*\*json\*\***: JSON (JavaScript Object Notation) ma'lumotlar formati bilan ishlash uchun funksiyalar.
8. **\*\*csv\*\***: CSV (Comma-Separated Values) formatdagi fayllar bilan ishlash.
9. **\*\*requests\*\***: HTTP so'rovlarini jo'natish uchun funksional.
10. **\*\*sqlite3\*\***: SQLite ma'lumotlar bazasi bilan ishlash.
11. **\*\*subprocess\*\***: Tizim boshqa dasturlarni chaqirish.
12. **\*\*pickle\*\***: Python obyektlarini serilizatsiya qilish va unserializatsiya qilish.
13. **\*\*argparse\*\***: Komand line argumentlarni qabul qilish.
14. **\*\*logging\*\***: Loglar yozish uchun funksional.
15. **\*\*unittest\*\***: Testlar yozish va bajarish uchun kerakli modul.

Bu modullar va yana ko'plab boshqa modullar, Pythonning standart kutubxonasi hisoblanadi va ularga foydalanish orqali keng qamrovli dasturlar yozish mumkin. Modullarni qanday qilib chaqirishni o'rganish va ulardan foydalanish, Python dasturlashda juda muhimdir.

Python dasturlash tili yildan-yilga ommalashib bormoqda. Bunga birinchi navbatda Pythonning sodda va tushunarli sintaksi sabab bo'lsa, ikkinchi va ehtimol eng ko'zga ko'ringan sabab bu Pythonning keng qamrovli kutubxonalar to'plamidir. Ushbu darsimizda Python kutubxonasidagi ba'zi muhim modullar bilan tanishamiz. Standart kutubxonaning to'liq tarkibi bilan Python rasmiy sahifasida tanishishingiz mumkin.

### **datetime** — SANA VA VAQT

Ushbu modul yordamida Pythonda sanalar bilan ishlashimiz mumkin. Moduldan foydalanishdan avval uni import qilamiz. Har gal moduldan foydalanishda `datetime` deb qayta yozmaslik uchun, import qilishda modulga `dt` nomini beramiz.

```
import datetime as dt
```

Hozirgi vaqt va sanani ko'rish uchun `datetime.now()` moduliga murojat qilamiz:

```
hozir = dt.datetime.now()
```

```
print(hozir)
```

Natija: 2024-26-04 12:12:19.894899

Ko`rib turganingizdek, natija yil, oy, kun soat, minut, sekund va millisekund ko`rinishida chiqdi. Biz bu qiymatlardan istaganimzni maxsus metodlar yordamida ajratib olishimiz mumkin:

```
# sanani ajratib olish
```

```
print(hozir.date())
```

```
# vaqtni ajratib olish
```

```
print(hozir.time())
```

```
# soatni ajratib olish
```

```
print(hozir.hour)
```

```
# minutni ajratib olish
```

```
print(hozir.minute)
```

```
# sekundni ajratib olish
```

```
print(hozir.second)
```

Natija:

2024-04-26

12:15:35.367013

12 # soat

15 # minut

35 # sekund

Agar bugungi kunning sanasi talab qilinsa `datetime` moduli ichidagi `date.today()` moduliga murojat qilamiz.

```
bugun = dt.date.today()
```

```
print(f"Bugungi sana: {bugun}")
```

Natija: Bugungi sana: 2024-04-26

Agar biror sanani qo`lda kiritish talab qilinsa `.date()` metodiga kerakli sanani (yil, oy, kun) ko`rinishida kiritamiz.

```
ertaga = dt.date(2024, 04, 27)
```

```
print(f"Ertangi sana: {ertaga}")
```

Natija: Ertangi sana: 2024-04-27

Faqatgina vaqt bilan ishlash uchun `.datetime.now().time()` metodiga murojat qilishimiz mumkin:

```
hozir = dt.datetime.now()
```

```
vaqtHozir = hozir.time()
```

```
print(f"Hozir soat: {vaqtHozir}")
```

Natija: Hozir soat: 12:21:54.529788

Istalgan vaqtni qo'lda kiritish uchun esa `.time()` metodiga kerakli vaqtni (soat, minut, sekund) ko'rinishida beramiz:

```
vaqtKeyin = dt.time(23,45,00)
```

Ayirish operatori yordamida sanalalar va vaqtlar orasidagi farqni hisoblashimiz mumkin:

```
bugun = dt.date.today()
```

```
ramazon = dt.date(2021, 4, 13)
```

```
farq = ramazon-bugun
```

```
print(farq)
```

```
print(f"Ramazonga {farq.days} kun qoldi")
```

Natija: Ramazonga 35 kun qoldi.

### XULOSA VA MUNOZARA

Standart kutubxona (yoki kitobxona) funksiyalari, bir dastur yozishda tekrorlanan amallarni qayta-qayta yozishga to'g'ri keladi. Ular bilan dasturni yozish davomida kichik bo'lgan va umuman qo'llaniladigan kod qismlarini o'zlashtirishingiz mumkin. Bu funksiyalar keng qamrovli va amaliy imkoniyatlarga ega bo'lgan oddiy va umumiy asosiy amallarni o'z ichiga oladi.

Python dasturlash tili keng imkoniyatlarga ega bo'lgani uchun, standart kutubxona o'z ichiga ko'plab modullarni (yozuvlar to'plamlarini) o'z ichiga oladi. Ular orqali fayllarni boshqarish, matematik amallarini bajarish, internet bilan aloqani o'rnatish, va h.k. amallar osonlik bilan bajariladi.

**ADABIYOTLAR RO`YXATI**

1. Kulakov A.G., Lando S.K., Semyonov A.L., Shen A.X.. Algoritmika. V-VII sinflar. Moskva: Drofa, 1997.
2. Boltayev B . Abduqodirov A., Taylaqov N., Mahkamov M., Azamalov A., Xafizov S. Informatika va hisoblash texnikasi asoslari. 9-sinf. T.: Cho`lpon, 2006.
3. Boltayev B., Mahkamov M., Azamatov A. Informaiikadan olimpiada masalalarini yechish. Metodik qo'llanma, T : 2004.
4. Boltayev B., Mahkamov M., Azamatov A. Informalikadan olimpiada masalalarini yechish-2. Melodik qo'llanma, Toshkent, 2004.