5. INPUT va SONLAR



Shu paytgacha biz o'zgaruvchilarning qiymatini dasturning ichida berayotgan edik. Keling endi qiymatni o'zimiz emas, balki dastur foydalanuvchilariga kiritish imkonini beramiz.

Buning uchun input() funktsyasidan foydalanamiz.

```
ism = input("Ismingiz nima?")

print("Assalom alaykum, " + ism)
```

Yuqoridagi dastur, avval 1-qatorda foydalanuvchining ismini so'raydi. Foydalanuvchi ismini kiritib, **Enter** tugmasini bosgach, foydalanuvchi kiritgan matnism degan o'zgaruvchiga yuklanadi va dasturining 2-qatori bajaradi:

Natija:

Ismingiz nima? anvar

Assalom alaykum, anvar

SONLAR

Dasturlash davomida turli sonlar bilan ishlash tabiiy hol. Pythonda sonlarning bir necha turlari bor. Keling ular bilan yaqindan tanishaylik.

INTEGERS — BUTUN SONLAR

Butun sonlarni ham o'zgaruvchida saqlash, ularning ustida qo'shish (+), ayirish (-), ko'paytirish(*), bo'lish (/) kabi arifmetik amalarni bajarish mumkin:

```
a = 20  # Sonlar musbat yoko

b = -30 # manfiy bo'lishi mumkin

c = a + b

print(c)
```

Natija: -10

Python - operatorlar orasidagi bo'shliqlarni inobatga olmaydi. O'qishga qulay bo'lishi uchun yuqoridagi kabi (bo'shliqlar bilan) yozishingiz mumkin.

```
# Kvadratning yuzini hisoblaymiz

kvdrt_tmni = 20 # Kavdratning tomoni 20 ga teng

kvdrt_yuzi = kvdrt_tmni**2 # Kvadrat yuzini hisoblaymiz

print(kvdrt_yuzi)
```

Natija: 400

FLOATS — HAQIQIY SONLAR

Pythonda haqiqiy sonlar **floating point numbers** yoki qisqa qilib **floats** deyiladi. "*Floating point numbers*" atamasini o'zbek tiliga "*suzuvchi nuqtali*

sonlar" deb tarjima qilish mumkin. Ingliz tilida o'nlik sonlarni yozishda vergul (,) emas nuqta (.) belgisi ishlatiladi va bu nuqta sonning katta kichikligiga qarab joyi o'zgargani uchun "*floating*" *(suzuvchi)* deyiladi.

```
pi = 3.14159 # o'nlik son (float)

radius = 10  # butun son (integer)

diametr = 2*radius

print("Aylana uzunligi ", pi*diametr, " ga teng.")
```

Natija: Aylana uzunligi 4.712384999999999 ga teng.

BUTUN SONDAN O'NLIK SONGA

Avval aytganimizdek ikki butun sonni bo'lish (/) natijasida o'nlik son hosil bo'ladi (natija butun bo'lsa ham).

```
a = -20

b = 40

c = b/a

print(c) # natija o'nlik son bo'ladi
```

Natija: -2.0

Shuningdek butun va o'nlik sonlar o'rtasidagi har qanday arifmetik amallarning natijasi ham o'nlik son bo'ladi.

```
a = 2

b = 3.0

# Quyidagi arifmetik amallarning natijasi o'nlik son hosil qiladi

print(a+b)

print(a*b)

print(a**b)
```

```
print(2*(a+b))
```

Natijalar:

2.0

5.0

6.0

8.0

10.0

UZUN SONLARNI KIRITISH

Uzun sonlarni kiritishda, qulaylik uchun, raqamlarni pastki chiziq (_) yordamida guruhlash mumkin. Python - son tarkibidagi pastki chiziqlarni (_) inobatga olmasdan, uzun sonligicha qabul qiladi.

```
aholi_soni = 7_594_000_000 # o'zmizga qulay bo'lishi uchun shinday yozdik

print("Yer kurrasida", aholi_soni, " ga yaqin odam yashaydi")
```

Natija: Yer kurrasida 7594000000 ga yaqin odam yashaydi

KONSTANTA

Aksar dasturlash tillarida konstant qiymatlar tushunchasi bor. Konstantlar o'zgarmas bo'ladi (misol uchun \pi π ning qiymati konstant, o'zgarmas qiymat). Pythonda konstant tushunchasi yo'q, shuning uchun dasturchilar bunday o'zgaruvchilarning nomini katta harflar bilan yozadilar (*ogohlantirish sifatida*). Bu albatta qat'iy qonun emas, lekin kelajakda o'zgaruvchilar orasida konstant qiymatlarni ajratish uchun yaxshi usul.

```
PI = 3.14159

raduis = 21.2
```

BIR NECHTA O'ZGARUVCHIGA QIYMAT BERISH

Birdaniga bir nechta o'zgaruvchiga qiymat berish uchun o'zgaruvchilar va ularga mos qiymatlar vergul (,) bilan ajratiladi:

x, y, z = 10, -7.25, -30

Yuqoridagi kod x ga 10, y ga -7.25, va z ga -30 qiymatini yuklaydi.

O'ZGARUVCHI TURINI ALMASHTIRISH

Keling quyidagi misolni ko'raylik, maqsadimiz ism va yosh degan ikki o'zgaruvchini yangi xabar degan o'zgaruvchiga yuklab, "Akobir 36 yoshda" degan matnni konsolga chiqarish:

ism = 'Akobir'

yosh = 36

xabar = ism + ' ' + yosh + ' yoshda'

print(xabar)

Natija: **TypeError: can only concatenate str (not "int") to str**

Afsuski, kutilgan natija o'rniga xatolik chiqdi. Agar xatoni ingliz tilidan tarjima qilsak, matn (str) va son (int) ni jamlab bo'lmaydi degan ma'no chiqadi.

Demak Pythonda matn (string) va son (int, float) turidagi o'zgaruvchilarni jamlab bo'lmas ekan. Xo'sh, bunga yechim bormi? **Albatta**.

Pythonda bir turdagi o'zgaruvchini boshqa turga o'tkazish mumkin, bu ingliz tilida ***typecasting*** detiladi. Buning uchun Pythonda mahsus funktsiyalar bor, keling ular bilan tanishamiz:

- str()— int yoki float turidagi sonlarni matnga o'zgartiradi.
- int()— matn yoki float ko'rinishidagi qiymatlarni butun songa o'zgartiradi. Bunda matn butun son ko'rinishida bo'lishi kerak.
- float()— matn yoki int ko'rinishidagi qiymatlarni o'nlik songa o'zgartiradi.

Demak, yuqoridagi kod to'g'ri ishlashi uchun 3-qatorni quyidagicha o'zgartiramiz:

```
ism = 'Akobir'

yosh = 36

xabar = ism + ' ' + str(yosh) + ' yoshda'

print(xabar)
```

Natija: Akobir 36 yoshda

str(yosh) kodi yosh degan o'zgaruvchining qiymatini matn ko'rinishida ko'rsatdi xolos. Asl o'zgaruvchining qiymati sonligicha qoladi. int() va float()ham huddi shunday ishlaydi.

O'ZGARUVCHI TURINI TEKSHIRISH

Kodimizda o'zgaruvchilar ko'payib ketdi. Yuqoridagi kabi xatolar qilmaslik uchun ba'zida o'zgaruvchinig turini tekshrish talab qilinadi. Buning uchun type() funktsiyasidan foydalanamiz:

```
ism = 'Akobir'

yosh = 36

print(type(ism))  # ism degan o'zgaruvchining turini konsolga chiqaramiz

print(type(yosh)) # ismyosh degan o'zgaruvchining turini konsolga chiqaramiz
```

Natija:

```
<class 'str'>

<class 'int'>
```

Kurib turganingizdek, ism nomli o'zgaruvchi 'str' ya'ni matn, yosh esa 'int' son turida ekan.

INPUT() VA SONLAR

Avvalgi mavzuimizda foydalanuvchidan ma'lumot olish uchun input() funktsyasidan foydalanishni o'rgandik. Kelin endi shu funktsiya yordamida foydalanuvchidan son olishni ko'raylik. Quyidagi kod foydalanuvchining tug'ilgan yilini so'raydi va uning yoshini hisoblab beradi:



Natija:

Tug'ilgan yilingizni kiriting:1982

Siz 39 yoshda ekansiz

Biz 2 va 6- qatorga o’zgartirish kiritdik, gap shundaki, input() funktsiyasi har qanday kiritilgan qiymatni matn (string) ko'rinishida qabul qiladi (garchi biz son kiritgan bo'lsak ham). 2021 sondan matnni ayirib bo’lmaydi. Shuning uchun 2 va 6-qatorlarni o’zgatirdik.

Yuqoridagi kodning 2-qatoriga e'tibor bersangiz, biz ikki funktsiyani bir biriga joylab yozdik (int(input()). Aslida, ajratib ham yozishimiz mumkin edi:

Image not found or type unknown

