

9

Blender dasturidagi transformatsiyalarni belgilang:

- a) Move, Rotate va Scale;
- b) Rotate, Copy va Cube;
- d) 3D Viewport, Scale va Timeline;
- c) Camera, Light va Cube.

V bob. PYTHONDA DASTURLASH

33-dars. RO'YXATLAR (LISTS) BILAN ISHLASH

Dasturda bir nechta ma'lumotni bir joyda saqlash uchun ularni ro'yxat shakliga keltirish lozim.

Ro'yxatlar kompyuterda katta hajmdagi bir xil ma'lumotlarni qayta ishlashda keng qo'llaniladi. Masalan, o'quvchilarning imtihonda olgan baholari, telefon tarmog'idagi abonentlar raqamlari va h. k.

Ro'yxatda bir vaqtda son, satr va boshqa turdagi ma'lumotlarni saqlash mumkin.

Odatda, har bir o'zgaruvchi biror nom bilan aniqlanib, tarkibida bitta ma'lumot (qiymat)ni saqlaydi.

Ro'yxatlar esa bir nechta ma'lumot (qiymat)larni o'z ichiga olishi mumkin. Masalan, 25 nafar o'quvchi

familiyasini saqlash uchun 25 ta o'zgaruvchi yoki bitta ro'yxatdan foydalanish mumkin.

Ro'yxat tartib bilan saqlangan elementlar tuzilmasi bo'lib, har bir elementga bitta indeks to'g'ri keladi va u orqali elementga murojaat etish mumkin.

Indeks raqamlari boshqa dasturlash tillaridagi kabi 0 dan boshlangan sonlar ketma-ketligidan iborat.

TAYANCH TUSHUNCHALAR

Massiv – bitta identifikator ostida *bir xil* obyektlar to'plamini saqlovchi ma'lumotlar turi.

Ro'yxat (list) – bitta identifikator ostida *har xil* obyektlar to'plamini saqlovchi ma'lumotlar turi.

Indeks – elementga murojaat qiluvchi raqamlar ketma-ketligi.

Ro'yxat nomi	mevalar			
Ro'yxat indeksi	0	1	2	3
	0	-3	-2	-1
Ro'yxat elementi	olma	banan	shaftoli	nok

Ro'yxatlarni e'lon qilish

Ro'yxatlar ham o'zgaruvchilar kabi e'lon qilinadi. Faqat ro'yxatlarni e'lon qilishda, ular o'lchami, ya'ni ro'yxatda saqlanuvchi elementlar sonini bilish lozim. Ro'yxatlar []qavs yoki list() konstruktori yordamida e'lon qilinadi.

Ro'yxatga elementlarni qo'shish, ya'ni ro'yxatlar hosil qilish uchun turli usullardan foydalanish mumkin.

1-usul. Elementlarni bevosita dastur kodiga kiritish orqali ro'yxat hosil qilish:

```
>>> mevalar = ['olma', 'banan', \
'shaftoli', 'nok', '12', True]
>>> mevalar1 = list('olma', 'banan', \
'shaftoli', 'nok', '12', True)
>>> print(mevalar[0])
olma
```

mevalar – ro'yxatni saqllovchi o'zgaruvchi nomi;
ro'yxatlar [] qavs yoki list() konstrukturi yordamida hosil qilinadi va ular ichida ro'yxat elementlari joylashadi;
ro'yxat elementlari vergul (,) bilan ajratib yoziladi;
\ (chap slesh) belgisi yordamida kodni keyingi qatordan davom ettirish mumkin.

2-usul. range() funksiyasi yordamida sonlar ketma-ketligidan iborat ro'yxat hosil qilish:

range(start, end, step) – start dan end gacha (end kirmaydi) bo'lgan sonlar ketma-ketligini step qadam bilan hosil qiladi.
step qadam ko'rsatilmasa, qadam sonini 1 deb oladi.

```
>>> sonlar=list(range(1, 5, 2))
>>> print(sonlar)

[1,3]
```

start=2, end=5, step=2. Sonlar o'zgaruvchisi 1 dan 5 gacha bo'lgan sonlar ketma-ketligini oladi.

3-usul. Klaviatura orqali kiritilgan elementlardan ro'yxat hosil qilish:

```
k=10
x=[0]*k

for i in range(0,k):
    x[i]=int(input())
print(x)
```

k – ro'yxatdagi elementlar soni;
x – ro'yxatning k(10) ta elementini joylashtirish uchun xotiradan joy zaxira qiladi. x = [0,0,0,0,0,0,0,0,0,0]

0 dan 9 gacha 10 ta sikl ishlaydi; klaviatura orqali kiritilgan elementlarni ro'yxatga o'zlashtiradi;

```
14 24 14 12 8 7 -5 4 -9 8
[14, 24, 14, 12, 8, 7, -5, 4, -9, 8]
```

klaviatura orqali ro'yxat elementlari kiritiladi.

4-usul. Tasodifiy sonlar operatori yordamida ro'yxat hosil qilish:

```
from random import randint
k=10
x=[0]*k
```

tasodifiy sonlar moduliga ulanadi;

```
for i in range(0,k):
    x[i]=randint(1,100)
print(x)
```

0 dan 9 gacha 10 ta sikl ishlaydi; 1 dan 100 gacha bo'lgan sonlar orasidan tasodifiy sonlarni ro'yxatga o'zlashtiradi.

```
[35, 36, 78, 6, 44, 70, 29, 40, 16, 14]
```

5-usul. Ro'yxat generatori yordamida ro'yxat hosil qilish:

<pre>>>> a=[0 for i in range(5)] >>> print(a)</pre>	0 qiymatga ega 5 ta elementdan tarkib topgan ro'yxat hosil bo'ladi;
<pre>>>> n=5 >>> a=[i**2 for i in range(n)] >>> print(a)</pre>	elementlari 0 dan 5 gacha bo'lgan sonlar kvadratlariga teng ro'yxat hosil bo'ladi;
<pre>>>> from random import randrange >>> n=10 >>> a=[randrange(1,10) for i in range(n)] >>> print(a)</pre>	random modulining randrange funksiyasini qo'llab, 1 dan 10 gacha bo'lgan sonlar orasidan tasodifiylari tanlab olingan ro'yxat hosil bo'ladi;
<pre>>>>a = [input() for i in range(int(input()))] >>> print(a)</pre>	foydalanuvchi tomonidan dastlab elementlar soni, so'ngra elementlarning o'zi kiritiladi. Ro'yxat shu tartibda hosil qilinadi.

Ro'yxat elementlari, asosan, sikllar bilan birga ishlatiladi:

<pre>>>> ranglar = ['ko'k', 'qizil', 'sariq'] >>> for i in ranglar: >>> print('Xona rangi: ', i)</pre>	ranglar o'zgaruvchisida ro'yxat saqlanadi; iteratsiyalar soni ro'yxat elementlari soniga teng.
Xona rangi: ko'k Xona rangi: qizil Xona rangi: sariq	<i>Ushbu dastur xona rangini har bir rang uchun chiqaradi.</i>

Ro'yxat elementlari, asosan, sikllar bilan birga ishlatiladi:

Ro'yxat uzunligi, eng kichik va eng katta elementini aniqlash uchun standart funksiyalardan foydalaniladi.

Funksiyalar	Tavsifi
join(list)	Ro'yxat elementlarini birlashtirish.
len(list)	Ro'yxat uzunligi, ya'ni elementlar sonini hisoblaydi.
sorted(list)	O'sish tartibida tartiblangan ro'yxatni chiqaradi.
sorted(list, reverse=True)	Kamayish tartibida tartiblangan ro'yxatni chiqaradi.
min(list)	Ro'yxatdagi eng kichik elementni aniqlaydi.
max(list)	Ro'yxatdagi eng katta elementni aniqlaydi.

Ro'yxatlar ustida amallar bajarish

1. Ikki ro'yxatni bir-biriga qo'shish mumkin, natijada ikki ro'yxatdan tarkib topgan yangi ro'yxat hosil bo'ladi.

```
>>> a=[1,2,3]
>>> b=[4,5]
>>> c=a+b
>>> print(c)
```

```
[1, 2, 3, 4, 5]
```

a ro'yxat e'lon qilindi.
b ro'yxat e'lon qilindi.
a ro'yxatga b ro'yxatni
qo'shib, c ro'yxat hosil qilindi.
c ro'yxat ekranga chiqarildi.

ESLAB QOLING



Ro'yxatlar o'zgartirish, ya'ni element qo'shish va o'chirish hamda ular tartibini almashtirish mumkin bo'lgan obyektlar hisoblanadi. Lekin shunday obyektlar ham borki, ularni yaratgandan keyin o'zgartirib bo'lmaydi. Masalan, kortejlar.

2. Ro'yxatni biror songa ko'paytirish mumkin, natijada takrorlangan ro'yxat hosil bo'ladi.

```
>>> a=[1,2,3]
>>> b=[4,5]
>>> d=b*2
>>> print(d)
```

```
[4, 5, 4, 5]
```

a ro'yxat e'lon qilindi.
b ro'yxat e'lon qilindi.
b ro'yxatni 3 ga ko'paytirib, d
ro'yxat hosil qilindi.
d ro'yxat ekranga chiqarildi.

3. Ro'yxat qismini qirqib olish mumkin, natijada qism ro'yxat hosil bo'ladi.

list(start:end:step) – qism ro'yxat hosil qilish;

list – ro'yxat nomi;

start – qirqib olinishi kerak bo'lgan qism ro'yxat 1-elementining indeks raqami;

end (end kirmaydi) – qirqib olinishi kerak bo'lgan qism ro'yxat oxirgi elementining indeks raqami;

step – qadamlar soni.

```
>>> yoshi=[14, 28, 39, 75, 55, 41, 32, 25]
>>> yoshi[2:8:2]
```

2-element 39 dan boshlab, 8-element yo'q. Shu sababli oxirigacha 2 qadam bilan qism ro'yxatni qirqib oladi.

```
>>> yoshi[2:4]=[10,20,30]
>>> print(yoshi)
```

```
[14, 28, 10, 20, 30, 55, 41, 32, 25]
```

2-element 39 dan boshlab 4-element 55 gacha (55 kirmaydi) bo'lgan qism ro'yxat o'rniga yangi [10,20,30] ro'yxatni qo'yadi.

Ro'yxatlarni o'zgartirish

Xohlagan vaqtda ro'yxat elementini almashtirish, o'chirish yoki ro'yxatga yangi element qo'shish mumkin. Buning uchun metodlardan foydalaniladi.

Metodlar	Tavsifi
list.insert(index, item)	Ro'yxatga index indeksi bo'yicha item elementini qo'shish.
list.append(item)	Ro'yxat oxiriga item elementini qo'shish.
list.remove(item)	Ro'yxatdan item elementini o'chirish. Ushbu metod ro'yxatdan birinchi uchragan item elementini o'chiradi, agar bu element ro'yxatda mavjud bo'lmasa, ValueError holati yuzaga keladi.

list.pop([index])	Ro'yxatdan index indeksi bo'yicha elementni o'chiradi. Indeks ko'rsatilmasa, oxirgi elementni o'chiradi.
list.clear()	Ro'yxatdagi barcha elementlarni o'chiradi.
list.index(item,[start [, end]])	Ro'yxatdagi item elementining indeksini qaytaradi (bu metodda start va end dan ham foydalanish mumkin).
list.count(item)	Ro'yxatdagi item elementlar sonini hisoblaydi.
list.sort()	Ro'yxat elementlarini o'sish tartibida tartiblaydi.
list.sort(reverse=True)	Ro'yxat elementlarini kamayish tartibida tartiblaydi.
list.copy()	Ro'yxatdan nusxa oladi.

Ro'yxatdagi elementlar o'rnini almashtirish

Axborotlarni qayta ishlash jarayonida ro'yxatdagi elementlarni yoki ular o'rnini almashtirishga to'g'ri keladi. Masalan, dars jadvaliga yangi fan qo'shilganda, fanning o'rnini yoki ikkita fan o'rnini almashtirish.

<pre>>>> a=[1, 2, 3] >>> b = a >>> print ('a=', a , 'b=' , b)</pre>	a nomli ro'yxat hosil qilindi. b nomli ro'yxat a ning qiymatiga teng. O'zgaruvchi nomi va qiymatini chiqaradi.
a=[1, 2, 3] b=[1, 2, 3]	a va b qiymatlari bir xil elementlar.
<pre>>>> a[1]=100 >>> print ('a=', a , 'b=' , b)</pre>	a ro'yxatning 2-elementi o'zgartirildi. O'zgaruvchi nomi va qiymatini chiqaradi.
a=[1, 100, 3] b=[1, 100, 3]	Ikkala ro'yxatdagi 2-element o'zgardi.
<pre>>>> b[2]=25 >>> print ('a=', a , 'b=' , b)</pre>	b ro'yxatning 3-elementi o'zgartirildi. O'zgaruvchi nomi va qiymatini chiqaradi.
a=[1, 100, 25] b=[1, 100, 25]	Ikkala ro'yxatdagi 3 - element o'zgardi.

Ichma-ich joylashgan ro'yxatlar

Ro'yxat elementlarining o'zi ham, o'z navbatida, ro'yxat bo'lishi mumkin. avto nomli ro'yxat uchta turli (raqami, nomi va rangi) ro'yxatdan tarkib topgan.

avto	Avto raqam [0]	Avto nomi [1]	Avto rangi [2]
[0]	1001 [0][0]	nexia [1][0]	red [2][0]
[1]	1002 [0][1]	spark [1][1]	white [2][1]
[2]	1003 [0][2]	tracker [1][2]	black [2][2]

<pre>>>> avto=[[1001,1002,1003], ['nexia', 'spark', 'tracker'], ['red', 'white', 'black']] >>> print (avto) # ro'yxatning barcha elementlarini chiqaradi</pre>
[[1001,1002,1003], ['nexia', 'spark', 'tracker'], ['red', 'white', 'black']]
<pre>>>> print(avto[2][-1]) # 2-ro'yxatning oxiridan 1-elementini chiqaradi # indeks sifatida manfiy son kiritilsa, ro'yxatning eng oxirgi elementi -1 # oxiridan oldingi elementi -2, va h. k.</pre>
black

Savol va topshiriqlar



1. Ro'yxat nima va u qanday e'lon qilinadi?
2. Indekslar manfiy son bo'lishi mumkinmi?
3. range() funksiyasi yordamida qanday massivlar hosil qilinadi?
4. randint() funksiyasi yordamida qanday massivlar hosil qilinadi?
5. Ichma-ich joylashgan ro'yxatlar qanday hosil qilinadi?
6. Ro'yxatdagi elementlar o'rnini qanday almashtirish mumkin?

34-dars. AMALIY MASHG'ULOT

Masala. Sinfidagi n nafar o'quvchining imtixon natijalari $x[n]$ ko'rinishidagi ro'yxatda berilgan. Ro'yxat elementlari har bir o'quvchining imtixon baholari (foizlarda)dan tarkib topgan. O'quvchilarning eng yuqori, eng past va o'rtacha o'zlashtirish ko'rsatkichini hisoblovchi dastur tuzing. O'quvchilar soni va ularning imtixon natijalari (100 foizda) klaviatura orqali kiritiladi. Matematika fanidan sizga ma'lumki, n nafar o'quvchining o'rtacha o'zlashtirish ko'rsatkichi quyidagiga teng:

$$R = \frac{x_1 + x_2 + \dots + x_n}{n}$$

print('O'quvchilar soni:') n=int(input())	n – ro'yxat elementlari soni, ya'ni o'quvchilar soni kiritiladi.
x=[0]*n	x ro'yxatning n ta elementini joylashtirish uchun xotiradan joy zaxira qiladi. $x = [0, 0, \dots, 0]$
R=0	Yig'indi hisoblanayotganligi uchun boshlang'ich qiymati 0 ga teng deb olinadi.
print('Imtixon natijalari:') for i in range(0,n): x[i]=int(input()) min_res=min(x) max_res=max(x) for i in range(0,n): R+=x[i] R=R/n	0 dan n gacha n ta sikl ishlaydi. Klaviatura orqali kiritilgan elementlarni ro'yxatga o'zlashtiradi.
	Ro'yxatdagi eng kichik elementni topadi.
	Ro'yxatdagi eng katta elementni topadi.
	Dastlab baholar yig'indisini hisoblab, keyin uni n ga bo'ladi va o'rtacha ko'rsatkichni chiqaradi.