

## LIST BILAN TANISHAMIZ

Avvalgi darsimizda biz o'zgaruvchi yaratish, va uning ichida biror qiymatni (matn yoki son) saqlashni o'rgandik. Bunda biz bitta o'zgaruvchiga bitta qiymat berdik xolos.

Bugun o'rganadigan navbatdagi mal'umot turi List (ro'yxat) deb ataladi. Ro'yxat o'z nomi bilan, bitta o'zgaruvchida bir nechta qiymatlarni saqlash imkonini beradi. Bu qiymatlar List elementlari deyiladi. Ro'yxatda son, matn yoki aralash turdagi elementlarni saqlash mumkin.

List quyidagicha yaratiladi:

```
mevalar = ['olma', 'anjir', 'shaftoli', 'o'rik'] # mevalar ro'yxati (matnlar)  
narhlar = [12000, 18000, 10900, 22000] # narhlar ro'yxati (sonlar)  
sonlar = ['bir', 'ikki', 3, 4, 5] # sonlar va matnlar aralash ro'yxat  
ismlar = [] # bo'sh ro'yxat
```

## LIST ELEMENTLARI

Ro'yxatdagi har bir element tartib bilan joylashgani sababli, biz istalgan elementga uning tartib raqami (indeksi) bo'yicha murojat qilishimiz mumkin.

Dasturlash olamida indeks 0 dan boshlanadi! Ya'ni Listning birinchi elementing tartib raqami (indeksi) 0 ga, ikkinchi elementning indeksi 1 ga teng va hokazo.

```
mevalar = ['olma', 'anjir', 'shaftoli', 'o'rik'] # mevalar ro'yxati (matnlar)  
print("Birinchi meva: ", mevalar[0])  
print("Ikkinchi meva: ", mevalar[1])
```

Agar list ichidagi elementlar matn ko'rinishid bo'lsa, ularga string metodlarni qo'llashimiz mumkin:

```
mevalar = ['olma', 'anjir', 'shaftoli', 'o'rik'] # mevalar ro'yxati (matnlar)  
print("Birinchi meva: ", mevalar[0].title())  
print("Ikkinchi meva: ", mevalar[1].upper())
```

List elementlari ustida arifmetik amallar bajarish:

```
narhlar = [12000, 18000, 10900, 22000]  
print(narhlar[2] + narhlar[3])
```

Pythonida Listning eng oxirgi elementiga -1 indeksi orqali ham murojat qilish mumkin. Bu usul Listning uzunligini bilmaganda juda asqotadi.

```
car_models = ['Toyota', 'GM', 'Volvo', 'BMW', 'Hyundai', 'Kia', 'Volkswagen']  
print(car_models[-1]) # Listning eng oxirgi elementiga -1 bilan murojat qilamiz  
ELEMENTLARNI QO'SHISH, O'CHIRISH VA O'ZGARTIRISH
```

Dastur davomida listning tarkibi o'zgarishi, yangi elementlar qo'shilishi, ba'zi elementlar o'chirilishi tabiiy hol. Misol uchun "Bozorlik ro'yxati" degan dasturni tasavvur qilaylik, foydalanuvchi ro'yxatga yangi mahsulotlar qo'shishi, sotib olganlarini esa o'chrishi mumkin.

Elementni o'zgartirish

Ro'yxatdagi biror elementning qiymatini o'zgartirish uchun, o'sha elementga indeksi bo'yicha murojat qilamiz va yangi qiymat yuklaymiz

```
narhlar = [12000, 18000, 10900, 22000]  
narhlar[0] = 13000 # 1-qiymatni 13000 ga o'zgartiramiz  
narhlar[2] = 11000 # 3-qiymatni 11000 ga o'zgartiramiz  
narhlar[3] = narhlar[3]+2000 # 4-qiymatga 2000 qo'shamiz  
print(narhlar)
```

.append() metodi

Ro'yxatga yangi element qo'shishning oson usuli bu .append() metodi yordamida ro'yxatning oxiriga qiymat qo'shish:

```
mevalar = ['olma', 'anjir', 'shaftoli', 'o'rik']  
mevalar.append("tarvuz") # mevalar ga tarvuz qo'shamiz  
print(mevalar)
```

.append() metodi bo'sh ro'yxatni to'ldirishda juda qulay usul. Odatda dastur boshida bo'sh ro'yxat yaratilib, dastur davomida ro'yxat foydalanuvchi tomonidan to'ldirib borilishi odatiy hol.

```
cars = [] # bo'sh ro'yxat yaratamiz  
cars.append('Lacetti') # ro'yxatga Lacetti mashinasini qo'shamiz  
cars.append('Nexia 3') # ro'yxatga Nexia 3 mashinasini qo'shamiz  
cars.append('Cobalt') # ro'yxatga Cobalt mashinasini qo'shamiz
```

**print(cars)**

.insert() metodi

Ro'yxatning istalgan joyiga yangi element qo'shish uchun .insert() metodidan foydalanamiz. .insert() metodi ichida yangi elementning indeksi va qiymati beriladi:

**cars = ['Lacetti', 'Nexia 3', 'Cobalt']**

**cars.insert(0, 'Malibu') # 1-o'ringa yangi qiymat qo'shamiz**

**print(cars)**

**cars.insert(2, 'Damas') # 3-o'ringa yangi qiymat qo'shamiz**

**print(cars)**

Elementni o'chirish

Ro'yxatdan biror elementni olib tashlash uchun uning indeksini yoki qiymatini bilishimiz lozim.

Inedks yordamida olib tashlash uchun del operatoridan foydalanamiz:

**mevalar = ['olma', 'anjir', 'shaftoli', "o'rik", 'anor']**

**del mevalar[1] # 2-element (anjir) ni o'chirib tashlaymiz**

**print(mevalar)**

Element qiymati bo'yichi o'chirish uchun esa .remove(qiymat) metodidan foydalanamiz. Buning uchun qavs ichida o'chirib tashlash kerak bo'lgan qiymatni yozamiz

**mevalar = ['olma', 'anjir', 'shaftoli', "o'rik", 'anor']**

**mevalar.remove('shaftoli') # Ro'yxatdan shaftolini o'chirdik**

**print(mevalar)**

.remove(qiymat) metodi ro'yxatda uchragan birinchi mos keluvchi qiymatni o'chiradi. Agar ro'yxatning ichida 2 va undan ko'p bir hil qiymatli elementlar bo'lsa, ulardan eng birinchisi o'chadi.

**hayvonlar = ['it', 'mushuk', 'sigir', 'qo'y', 'quyon', 'mushuk']**

**hayvonlar.remove("mushuk") # Ro'yxatda 2 ta mushuk bor, ulardan birinchisi o'chadi**

**print(hayvonlar)**

Elementni sug'urib olish

Ba'zida biror elementni butunlay o'chirib tashlash emas, balki uni ro'yxatdan sug'urib olish va undan foydalanish talab qilinishi mumkin. Buning uchun Pythonda .pop(indeks) metodidan foydalanmiz.

**bozorlik = ["yog", 'un', 'piyoz', 'banan', "go'sht"]**

**mahsulot = bozorlik.pop(3) # Ro'yxatdan banan ni sug'urib olamiz**

**print("Men " + mahsulot + " sotib oldim")**

**print("Olinmagan mahsulotlar: ", bozorlik)**

Men banan sotib oldim

Olinmagan mahsulotlar: ["yog", 'un', 'piyoz', "go'sht"]

Agar .pop() metodida indeks berilmasa, ro'yxatdan o'xirgi qiymat sug'urib olinadi