

PYTHON DASTURLASH TILIDA RO'YXATLAR, KORTEJLAR VA LUG'ATLAR BILAN ISHLASH

Onarqulov Maksadjon Karimberdiyevich

Farg'ona davlat universiteti Amaliy matematika

va informatika kafedrası dotsenti

maxmaqsad@gmail.com

Yusupov Mirsaid Abdulaziz o'g'li

Farg'ona davlat universiteti Amaliy matematika

va informatika kafedrası o'qituvchisi

mirsaidbeky@gmail.com

Nuritdinova Nargizaxon Muxtorali qizi

Farg'ona davlat universiteti talabasi

aikasd32ew@gmail.com

Annotatsiya. Python dasturlash tili juda qulay va oson o'rganiladigan til sifatida keng tarqalgan. Uning ba'zi asosiy ma'lumot tuzilmalari — ro'yxatlar, kortejlar va lug'atlardir. Bu tuzilmalar yordamida ma'lumotlarni samarali saqlash va boshqarish mumkin. Ushbu maqolada biz ro'yxatlar, kortejlar va lug'atlarga oid tushunchalar, ularning xususiyatlari va qanday ishlatilishi haqida batafsil ma'lumot beramiz.

Kalit so'zlar: Python, ro'yxatlar, kortejlar, lug'atlar, ma'lumot tuzilmalari, dasturlash, o'zgaruvchi, indeks, kalit-qiymat, immutable, mutable, Python list, tuple, dictionary, ma'lumotlarni saqlash.

Annotation. Python is a widely used programming language known for its simplicity and ease of learning. Some of its fundamental data structures include lists, tuples, and dictionaries. These structures allow for efficient storage and management of data. In this article, we provide a detailed explanation of lists, tuples, and dictionaries, their characteristics, and how they are used in Python programming

Keywords: Python, lists, tuples, dictionaries, data structures, programming, mutable, index, key-value pair, immutable, Python list, tuple, dictionary, data storage

Аннотация. Язык программирования Питон широко используется благодаря своей простоте и лёгкости в освоении. Одними из его основных структур данных являются списки, кортежи и словари. Эти структуры позволяют эффективно сохранять и обрабатывать данные. В данной статье подробно рассматриваются понятия списков, кортежей и словарей, их особенности и способы использования в программировании на языке Питон.

Ключевые слова: Питон списки, кортежи, словари, структуры данных, программирование, изменяемый, индекс, пара "ключ-значение", неизменяемый, изменяемый, список Питон, кортеж, словарь, хранение данных

1. Ro'yxatlar — Python'dagi eng ko'p ishlatiladigan ma'lumot tuzilmasidir. Ro'yxat bir nechta elementlardan iborat bo'lib, ular o'zaro ajralgan vergullar bilan bir qatorda joylashadi. Ro'yxatni yaratish uchun kvadrat qavslar [] ishlatiladi. Ro'yxatdagi elementlar turli ma'lumot turlarida bo'lishi mumkin: sonlar, satrlar, booleans, hatto boshqa ro'yxatlar.

- **Ro'yxatni yaratish:** `my_list = [1, 2, 3, 4, 5]`

- Ro'yxatdagi elementlarga indekslar orqali murojaat qilinadi. Indeks 0 dan boshlanadi.

```
print(my_list[0])
```

 Natija: 1

```
print(my_list[-1])
```

 Natija: 5 (oxirgi element)

- `append()` metodi ro'yxatning oxiriga yangi element qo'shadi.

```
my_list.append(6)
```

```
print(my_list)
```

 Natija: [1, 2, 3, 4, 5, 6]

- `remove()` metodi ro'yxatdan ma'lum bir elementni o'chiradi.

```
my_list.remove(3)
```

```
print(my_list)
```

 Natija: [1, 2, 4, 5, 6]

- Indeks orqali ro'yxatdagi elementni o'zgartirish mumkin.

```
my_list[2] = 10
```

```
print(my_list)
```

 Natija: [1, 2, 10, 5, 6]

Ro'yxatlar dinamik bo'lib, ular o'zgarishi mumkin .

2. Kortejlar - ro'yxatlarga o'xshash, ammo ular o'zgarmaydigan ma'lumot tuzilmasidir. Kortejlar, asosan, ma'lumotlarni o'zgarimas holatda saqlash kerak bo'lganda ishlatiladi. Kortejni yaratish uchun qavslar () ishlatiladi.

- **Kortejni yaratish:**

```
my_tuple = (1, 2, 3, 4)
```

- Kortejdagi elementlarga ro'yxat kabi indeks orqali murojaat qilinadi.

```
print(my_tuple[2])
```

 Natija: 3

- Kortejlar o'zgarimas bo'lgani uchun ularning elementlarini o'zgartirish mumkin emas.

- Quyidagi kod xato beradi:

```
my_tuple[1] = 5
```

- Kortejlar birlashtirilishi mumkin: `new_tuple = my_tuple + (5, 6)`

```
print(new_tuple)
```

 Natija: (1, 2, 3, 4, 5, 6)

Kortejlar asosan ma'lumotlar o'zgarimas bo'lishi kerak bo'lgan holatlarda ishlatiladi va ularning ishlash tezligi ro'yxatlarga qaraganda yuqoriroq bo'lishi mumkin.

3. Lug'atlar — kalit va qiymat juftliklaridan tashkil topgan ma'lumot tuzilmasidir. Bu tuzilma orqali ma'lumotlar kalitlar yordamida qidiriladi va ularga murojaat qilinadi. Lug'atni yaratish uchun qavslar {} ishlatiladi.

• **Lug'atni yaratish:**

```
my_dict = {'name': 'John', 'age': 30, 'city': 'New York'}
```

• **Kalit orqali qiymatga murojaat qilish:**

```
print(my_dict['name'])
```

Natija: John

• Lug'atga yangi kalit-qiymat juftligini qo'shish mumkin.

```
my_dict['job'] = 'Developer'
```

```
print(my_dict)
```

Natija: {'name': 'John', 'age': 30, 'city': 'New York', 'job': 'Developer'}

• del operatori yordamida lug'atdagi kalit-qiymat juftligini o'chirish mumkin. del my_dict['age']

```
print(my_dict)
```

Natija: {'name': 'John', 'city': 'New York', 'job': 'Developer'}

• Lug'atdagi barcha kalitlar va qiymatlar alohida ro'yxatlarda olinishi mumkin.

```
print(my_dict.keys()) # Kalitlar: dict_keys(['name', 'city', 'job'])
```

```
print(my_dict.values()) # Qiymatlar: dict_values(['John', 'New York', 'Developer'])
```

Lug'atlar ko'plab turli vazifalarni bajarishda, ayniqsa ma'lumotlarni tezkor qidirish va saqlashda juda foydalidir.

4. Ro'yxatlar, Kortejlar va Lug'atlarga Qarshi Solishtirish

Xususiyat	Ro'yxatlar	Kortejlar	Lug'atlar
Tuzilma	O'zgarmas (Mutable)	O'zgarmas (Immutable)	O'zgarmas
Elementlar	Indeks orqali murojaat	Indeks orqali murojaat	Kalit-qiymat juftligi
Qurilishi	[]	()	{}
Foydalanish	O'zgaruvchi ma'lumotlar	O'zgarmas ma'lumotlar	Ma'lumotlarga kalit orqali murojaat qilish
Ishlash tezligi	Tez (Lekin o'zgarmas emas)	Tez va samarali (O'zgarmas)	Juda tez qidirish (Kalit yordamida)

Xulosa. Python'dagi ro'yxatlar, kortejlar va lug'atlarga oid tushunchalar ma'lumotlar tuzilmalari va ularning imkoniyatlari haqida kengroq tasavvur hosil qilishga yordam beradi. Ro'yxatlar dinamik bo'lib, o'zgarishi mumkin bo'lsa, kortejlar o'zgarmas va tez ishlaydi. Lug'atlar esa kalit-qiymat asosida ma'lumotlarni tezkor saqlash va boshqarish imkonini beradi. Har bir ma'lumot tuzilmasi o'zining

afzalliklari va o'ziga xos ishlash xususiyatlari bilan dasturchilarga qulaylik yaratadi.

Foydalanilgan adabiyotlar:

1. Lutz, M. (2013). *Learning Python* (5th ed.). O'Reilly Media.
2. Sweigart, A. (2015). *Automate the Boring Stuff with Python*. No Starch Press.
3. Beazley, D., & Jones, B. K. (2013). *Python Cookbook* (3rd ed.). O'Reilly Media.
4. Python Software Foundation. (2024). *Python Documentation* — <https://docs.python.org/3/>
5. Zelle, J. M. (2010). *Python Programming: An Introduction to Computer Science* (2nd ed.). Franklin, Beedle & Associates Inc.
6. Downey, A. (2015). *Think Python: How to Think Like a Computer Scientist* (2nd ed.). Green Tea Press.
7. McKinney, W. (2017). *Python for Data Analysis* (2nd ed.). O'Reilly Media.