

---

## DICTIONARIES IN THE PYTHON PROGRAMMING LANGUAGE (DICTINORAY) AND ACTIONS ON THEM, LINKING BACK TO DICTINORAY

**Djumoboyeva Yanglish Egamnazarovna**

Senior Lecturer of the Department of Methods of exact and natural sciences at Syrdarya regional center for retraining and advanced training of public educators

---

### ABSTRACT

In Python, the Dictionary type is important when working with datasets. This type acts like an associative array in other languages and is used to read data in JSON files.

*Keywords: dictionary, JSON, update, pop, internal dictionary, clear, keys, items, dictionary loops.*

### INTRODUCTION

Hozirgi kunda ta'lim-tarbiya jarayoniga zamonaviy axborot texnologiyalarini joriy etish, ta'limni kompyuterlashtirish muammolarini hal qilish muhim ahamiyat kasb etadi.

Python bu umumiy maqsadli dasturlash uchun keng tarzda foydalaniladigan yuqori darajali dasturlash tili, chunki o'rganish oson va qulay sintaksisga ega. Undan tashqari skriptli dasturlash tillariga kiradi. Python dinamik tipizatsiyaga ega, obyektga yo'naltirilgan dasturlash, funksional dasturlash, strukturali, avtomatik xotirani boshqarish va albatta ko'p patokli dasturlash tillaridan biri. Python har xil platformalar uchun yozilgan masalan Windows, Linux, Mac OSX, Palm OS, Mac OS va hokazo. Python Microsoft.NET platformasi uchun yozilgan realizatsiyasi ham bor uni nomi IronPython. Bugungi kunda dunyoga mashhur ko'plab kompaniyalar NASA, Google, Yandex, CERN, Apple computer, Dream Works, kosmik teleskop institutlari Pythonni ishlatishadi. Dunyoning rivojlangan mamlakatlari AQSH (Koliforniya Universiteti, Florida Universiteti, Lova Universiteti, Massachushta Texnologiya Universiteti), Kanada (Toronto Universiteti, Alberto Universiteti), Buyuk Britaniya (Oksford Universiteti), Fransiya, Rossiya, Avstraliya, Ispaniyaning universitet va kollejlari o'qitishda Python dasturlash tili qo'llaniladi.

Pythondagi lug'atlar kalit bo'yicha kirishga ruxsat etuvchi erkin obyektlarning tartiblangan jamlanmasi. Ularni yana assotsiativli massivlar yoki hesh jadvallar deb nomlaydilar. Sodaroq qilib aytadigan bo'lsak lug'at xuddi manzillar kitobiga o'xshaydi, ya'ni biror insonning ismini bilgan holda uning manzili yoki u bilan bo'g'lanish ma'lumotlarini olish mumkin. Lug'atlar bilan ishlash uchun ham albatta uni yaratish kerak. Lug'atni yaratishni esa bir necha usullari bor. [1]

#### Lug'at metodlari

**Dict.clear()**- lug'atni tozalaydi.

**Dict.copy()**-lug'at nusxasini qaytaradi.

Classmethod **dict.fromkeys(seq[, value])**- Seq dan kalitni va Value qiymatlariga ega bo'lgan lug'atni yaratadi.

**Dict.get(key[, default])**-kalit qiymatini qaytaradi, lekin u bo'lmasa xatolik beradi, default (jimlikda None) qaytaradi.

**Dict.items()**-juftliklarni qaytaradi(kalit, qiymat)

**Dict.keys()**- lug'atdagi kalitlarni qaytaradi

**Dict.pop(key[default])**-kalitni yo'qotib qiymatni qaytaradi. Agarda kalit bo'lmasa defaultni qaytaradi.

**Dict.popitem()**- juftlikni o'chirib qaytaradi (kalit, qiymat). Agarda lug'at bo'sh bo'lsa KeyError istisnoni chaqiradi. Esingizda tursin lug'atlar tartibli emas.

**Dict.setdefault(key [, default])**-kalit qiymatni qaytaradi, lekin u bo'lmasa xatolik bermaydi, default qiymatga ega kalitni yaratadi (jimlikda None).

**Dict.update([other])**- other dan juftliklarni (kalit, qiymat) kiritib lug'atni to'ldiradi.

Mavjud bo'lgan kalitlar qaytadan yoziladilar. None (eski lug'at) qaytaradi.

**Dict.values()**-lug'atdagi qiymatni qaytaradi.

Python dasturlash tilida mavjud bo'lgan optimal qulayliklardan biri bu – Dictionary. Dictionary bu ma'lumotlar to'plamining key:value shaklidagi tip.[2]

```
thisdict = {
    "brand": "Ford",
    "model": "Mustang",
    "year": 1964
}
```

Bu yerda "brand", "model", "year" dictionary kalitlar, "Ford", "Mustang", 1964 elementlar qiymatlar. Dictionary tipi o'zgaruvchan va element nusxasi mavjud emas. Dictionary elementlari qiymatlari turli hil tipga ega bo'lishi mumkin.

```
thisdict = {
    "brand": "Ford",
    "electric": False,
    "year": 1964,
    "colors": ["red", "white", "blue"]
}
```

Dictionaryning bir oz murakkablik jihati Dictionaryda **key** va **value** qiymat parametrlari mavjud. Elementdan key yoki value qiymatlarini olish uchun **get()**, **key()** va **value()** funksiyalari mavjud. Har bir elementlarini chaqirish uchun **item()** funksiyasi ishlatiladi. [2]

<pre>thisdict = {     "brand": "Ford",     "model": "Mustang",     "year": 1964 } x = thisdict["model"] print(x) x = thisdict.get("model") print(x) x = thisdict.keys() print(x) thisdict['model'] = 'Lasetty' print(thisdict) x = thisdict.values() print(x) x = thisdict.items()</pre>	<p><b>Dastur natijasi:</b></p> <p>Mustang</p> <p>Mustang</p> <p>dict_keys(['brand', 'model', 'year'])</p> <p>{'brand': 'Ford', 'model': 'Lasetty', 'year': 1964}</p> <p>dict_values(['Ford', 'Lasetty', 1964])</p> <p>dict_items([('brand', 'Ford'), ('model', 'Lasetty'), ('year', 1964)])</p>
--	---

INTERNATIONAL SCIENTIFIC AND PRACTICE CONFERENCE ON "  
INTERNATIONAL EXPERIENCE IN INCREASING THE EFFECTIVENESS OF  
DISTANCE EDUCATION: PROBLEMS AND SOLUTIONS"

print(x)	
----------	--

Dictionary uchun yana bir qancha metodlar mavjud. Bular element qo‘shish, o‘zgartirish, elementni o‘chirish, Dictionaryni o‘chirish va tozalash.

<pre> thisdict = {     "brand": "Ford",     "model": "Mustang",     "year": 1964 } print(thisdict) thisdict["color"] = "red" print(thisdict) thisdict.update({"color": "blue"}) thisdict.pop("model") print(thisdict) thisdict.popitem() print(thisdict) del thisdict["year"] print(thisdict) thisdict.clear() print(thisdict) </pre>	<p><b>Dastur natijasi:</b></p> <pre> {'brand': 'Ford', 'model': 'Mustang', 'year': 1964} {'brand': 'Ford', 'model': 'Mustang', 'year': 1964, 'color': 'red'} {'brand': 'Ford', 'year': 1964, 'color': 'blue'} {'brand': 'Ford', 'year': 1964} {'brand': 'Ford'} {} </pre>
---	---

JSON - ma'lumotlarni saqlash va almashtirish uchun sintaksis. JSON (JavaScript object notation) - bu matn, JavaScript-ni ob'ekt belgisi bilan yozilgan.

JSON ma'lumotlar bazasi yozuvlarini python tiliga to'g'ridan to'g'ri dictionaryga o'tkaziladi. JSON bilan ishlash uchun json kutubxonasini elon qilish zarur.

<pre> (import json). import json # some JSON: x = '{ "name":"John", "age":30, "city":"New York"}' # parse x: y = json.loads(x) # the result is a Python dictionary: print(y["age"]) </pre>	<p><b>Dastur natijasi:</b></p> <p><b>30</b></p>
--	---

Bu misolda **y** yangi dictionary **x** o'zgaruvchisida JSON ma'lumotlari joylashtirilgan. Bu yerda JSON oddiy matn shaklida kiritilgan, lekin JSON internet tarmog'ida yoki boshqa manbadan kelgan holda dictionary tipi juda katta avzallik beradi. Albatta Dictionary o'zgaruvchisi qiymatini JSONga ham o'zgartirib biror manzilga jo'natish mumkin. [2]

import json	<b>Dastur natijasi:</b>
-------------	-------------------------

<pre># a Python object (dict): x = {     "name": "John",     "age": 30,     "city": "New York" } # convert into JSON: y = json.dumps(x) # the result is a JSON string: print(y)</pre>	<pre>{"name": "John", "age": 30, "city": "New York"}</pre>
---	--

## CONCLUSION

Xulosa qilib aytganda o'quvchilar Python darsturlash tiliga bo'lgan qiziqishini oshirish darsda qanday zamonaviy axborot kommunikatsion texnologiyalardan foydalanish bu jarayonda o'quvchining faolligi va qiziquvchanligini ta'minlab borish kerak. Aks holda ko'zlangan natija samarasiz yakunlanishi mumkin. O'qituvchi, yangilikka va islohotlarga tayyor turishi, unda faol ishtirok etishi uchun o'z ustida tinimsiz ishlashi, kasbiy mahoratini oshirib borishi, zamonaviy axborot va kompyuter texnologiyalaridan xabardor bo'lishi lozim.

## REFERENCES

1. Дмитрий Мусин. Самоучитель Python. 2015 г
2. [www.python.org](http://www.python.org).