

# Python Dasturlash Tili

## 1 Python dasturlash tilida berilganlar turlari va o'garuvchilar.

- Python o'zgaruvchilari
- Python o'zgaruvchilariga qiymat berish
- Bir nechta o'zgaruvchiga qiymat berish
- O'zgaruvchilarning qiymatini almashtirish
- O'zgaruvchilarni o'chirish
- Python berilganlar turlari
- O'zgaruvchi turlarni o'zgartirish



# Python Variables

<code>x = 45</code>	<p>Type = Integer</p> <p><code>45</code></p> <p>x</p>
<code>name = "DataFlair"</code>	<p>Type = String</p> <p><code>"DataFlair"</code></p> <p>name</p>
<code>nums = [ 1, 8.5, 9 ]</code>	<p>Type = Lists</p> <p><code>[1, 8.5, 9]</code></p> <p>nums</p>

O'zgaruvchi - bu qiymat uchun konteyner bo'lib, unga nom berish kerak undan keyinroq dasturda murojaat qilish uchun. Belgilangan qiymatga asoslanib, interpretator ma'lumotlar turini belgilaydi. Siz har doim o'zgaruvchida boshqa turni saqlashingiz mumkin. Misol uchun, agar siz 7 ni o'zgaruvchida saqlasangiz, keyinroq "Dinosaur" ni saqlashingiz mumkin.

## O'zgaruvchini (identifikator)

qanday nomlash mumkinligi haqida ma'lum qoidalar mavjud.

- 1 Python o'zgaruvchilari faqat harf (A-Z/a-z) yoki pastki chiziq (\_\_) bilan boshlanishi mumkin.

```
1 9lives=9 # Xato nom berish # SyntaxError: invalid syntax
2 flag=0 # To'g'iri
3 _9lives='cat' # To'g'iri
```

- 2 Identifikatorning qolgan qismida harflar (A-Z/a-z), pastki chiziq (\_\_) va raqamlar (0-9) bo'lishi mumkin.

```
1 year2='Sophomore' # To'g'ir
2 _$$=7 # Xato => SyntaxError: invalid syntax
3 name='Ayushi' # To'g'iri
```

```
1 print(Name)
```

Xatolik:

*Traceback (most recent call last): File "<pyshell#21>", line 1, in <module> Name  
NameError: name 'Name' is not defined*

- 4 Zaxiralangan so'zlarni (kalit so'zlarni) identifikator nomlari sifatida ishlatish mumkin emas.

Table:

and	def	False	import	not	True
as	del	finally	in	or	try
assert	elif	for	is	pass	while
break	else	from	lambda	print	with
class	except	global	None	raise	yield
continue	exec	if	nonlocal	return	

## Python o'zgaruvchilariga qiymat berish

Python o'zgaruvchilariga qiymat berish uchun siz uning turini e'lon qilishingiz shart emas. Siz uni **Python o'zgaruvchilar nomlash qoidalari** bo'limida ko'rsatilgan qoidalarga muvofiq nomlaysiz va tenglik belgisidan keyin qiymatni kiriting (=).

```
1 age=7
2 print(age)
```

Natija: 7

```
1 age='Dinosaur'
2 print(age)
```

Natija: *Dinosaur*

Biroq, `age=Dinosaur` yozish mumkin emas. Bundan tashqari, siz o'zgaruvchiga qiymat berishdan oldin o'zgaruvchidan foydalana olmaysiz.



```
1 print (Name)
```

Xatolik:

*Traceback (most recent call last): File "<pyshell#21>", line 1, in <module> Name  
NameError: name 'Name' is not defined*

Biroq, identifikatorni teng belgisining o'ng tomoniga qo'yib bo'lmaydi. Quyidagi kod xatolikka olib keladi.

```
1 7=age
```

Xatolik:

*SyntaxError: can't assign to literal*

Python o'zgaruvchilari kalit so'zga ham qiymat sifatida bera olmaysiz.

```
1 False=choice
```

Xatolik:

*SyntaxError: can't assign to keyword*

## Bir nechta o'zgaruvchiga qiymat berish

Bir vaqtning o'zida bir nechta o'zgaruvchiga qiymat berishingiz mumkin.

```
1 age, city=21, 'Indore'  
2 print(age, city)
```

Natija:

*21 Indore*

Yoki bir nechta Python o'zgaruvchilariga bir xil qiymatni belgilashingiz mumkin.

```
1 age=fav=7  
2 print(age, fav)
```

Natija:

*7 7*

Python o'zgaruvchilariga qiymatlarni shunday belgilashingiz mumkin.

## O'zgaruvchilarning qiymatini almashtirish

Python o'zgaruvchilarning qiymatini almashtirish juda oson.

```
1 a,b='red','blue'  
2 a,b=b,a  
3 print(a,b)
```

Natija:

*blue red*

Yoki bir nechta Python o'zgaruvchilariga bir xil qiymatni belgilashingiz mumkin.

```
1 age=fav=7  
2 print(age,fav)
```

Natija:

*7 7*

Python o'zgaruvchilariga qiymatlarni shunday belgilashingiz mumkin.

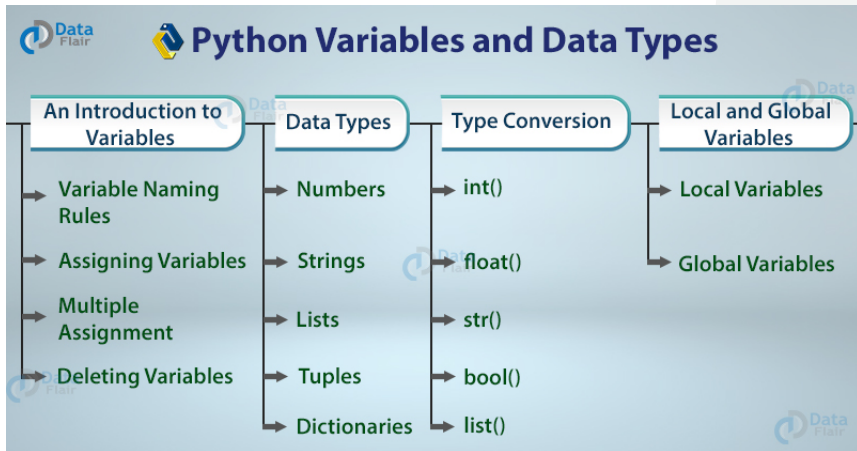
## O'zgaruvchilarni o'chirish

Python o'zgaruvchilarini "del" kalit so'zidan foydalanib ham o'chirishingiz mumkin.

```
1 a = 'red'  
2 del a  
3 print(a)
```

Xatolik:

*Traceback (most recent call last): File "<pyshell#39>", line 1, in <module> a NameError: name 'a' is not defined*



Python berilganlarning to'rtta sonli turi mavjud.

**int** – butun sonni bildiradi. Ushbu Python berilganlar turi ishorasiz butun sonlarni saqlaydi. O'zgaruvchining turini aniqlash uchun `type()` funksiyasidan foydalanishimiz mumkin.

Natija:

```
<class 'int'>
```

- Butun son har qanday uzunlikda bo'lishi mumkin, yagona cheklov mavjud xotiradir.

```
1 a=-7
2 print(type(a))
```

Natija:

`<class 'int'>`

**float** – ushbu Python berilganlar turi suzuvchi nuqtali haqiqiy qiymatlarga ega. **int** faqat 3 raqamini saqlashi mumkin, lekin agar xohlasangiz **float** 3.25 ni saqlashi mumkin.

```
1 a=3.0
2 print(type(a))
```

Natija:

`<class 'float'>`



- **long** – Python berilganlar turi cheksiz uzunlikdagi uzun butun songa ega. Lekin bu konstruktsiya Python 3.x da mavjud emas.
- **complex** – Python berilganlar turi kompleks raqamga ega. Kompleks sonning ko'rinishi quyidagicha :  $a+bj$  Bu yerda  $a$  va  $b$  sonning haqiqiy va  $j$  mavhum qismlari.

```
1 a=2+3j  
2 type(a)
```

Natija:

`<class 'complex'>`

- 2 **Strings** – bu belgilar ketma-ketligi. Python-da C++ yoki Java-dan farqli o'laroq char berilganlar turi mavjud emas. Siz bitta qo'shtirnoq yoki ikkita qo'shtirnoq yordamida satrni chegaralashingiz mumkin.
- 3 **Lists** – bu qiymatlar to'plami. Esda tutingki, u turli xil qiymatlarni o'z ichiga olishi mumkin.
- 4 **Tuples** – ro'yxatga o'xshaydi. Buning o'rniga siz uni qavslar yordamida e'lon qilasiz.
- 5 **Dictionaries** – kalit-qiymat juftligini o'z ichiga oladi. Uni jingalak qavslar ichida, vergul bilan ajratiladi va juftliklar bilan e'lon qiling. Kalit va qiymatlar ikki nuqta (:) bilan ajratiladi.
- 6 **Sets** – qiymatlar ro'yxatiga ega bo'lishi mumkin. Buni qavslar yordamida aniqlang.

## O'zgaruvchi turlarni o'zgartirish

Python dinamik turli dastrulash tili bo'lsada, dasturchi o'zgaruvchi turlarini boshqa turga keltirishi mumkin. Buning uchun quyidagi funksiyalardan foydalanamiz:

① `int()` – qiymatni int ga aylantiradi.

```
1 a = 3.7 # a ning turi float
2 print(a, type(a))
3 b = int(a) # b ning turi int
4 print(b, type(b))
```

Natija:

3.7 <class 'float'>

3 <class 'int'>

Savollar

E'tiboringiz uchun rahmat  
Thanks for attention