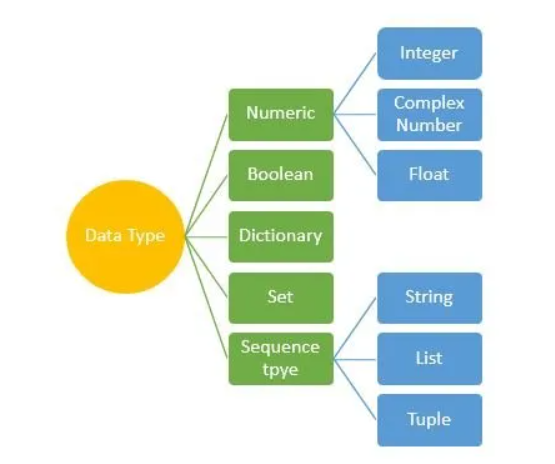
Типове данни

22.01.2023

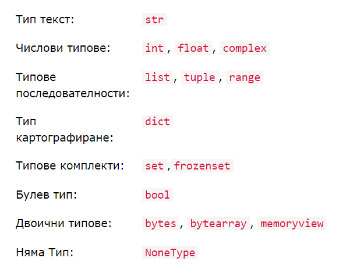
4 часа

В този урок ще се запознаете с типовете данни в Python и как да създавате и използвате променливи.

Променлива в език за програмиране е името на запазени места в паметта на компютъра за съхранение на ценности. Това означава, че при създаване на променлива, която действително запазваме определено място в паметта на компютъра.В Python променливите са от динамичен тип, т.е. може на една променлива да присвоим целочислена стойност, а след това да ѝ присвоим някакъв низ от символи. В зависимост от типа на променливи данни, преводачът разпределя необходимото количество памет и решава какво може да е в зоната на резервирания памет.За разбиране, можете да мислите за променлива като кутия, в която можете да сложите нещо, но само с определен размер.

# Присвояване на стойност на променлива:

Операндът от лявата страна на знака за равенство (=) е името на една променлива, операнда от дясната страна - определената стойност на тази променлива.

Стандартните типове данни в Python са:   
**1.Number  
2.String  
3.List  
4.Тuple  
5.Dictionary**

|  |  |
| --- | --- |
| 1  2  3 | var1 = 15 # integer  var2 = 1.13 # float  var3 = 3.14j # complex |

1.Number – биват няколко вида:   
a) int – целочислен тип  
б) long – целочислен тип с по-голям размер  
в) float – число с плаваща запетая  
г) complex – комплексно число

Примери за числа:

Числата представляват числовите стойности.

* Цяло число (int): Включва положителни, както и отрицателни числа (2, 555, -80). Тя може да бъде с всякаква дължина.
* Комплексно число: Съдържа подредени двойки числа, т.е. във вид x + iy, където x е реално число, а y е въображаемо число. Например, 2 + 3i, 6i и т.н.
* Float: Съхранява числа с плаваща запетая и дължината е до 15 знака след десетичната запетая. Пример, 2.568 и т.н.

ЗАБЕЛЕЖКА: За да знаете вида на данните, **Тип()** се използва функция. За да проверите дали даден обект принадлежи към определен клас **isinstance()** се използва функция.

Както обикновено, за присвояване на стойност използваме оператора ‘=’.

2.String – низ, или масив от символи.  
Пример: name = "Martin"

Тъй като низът представлява масив от символи, можем да достъпваме негови елементи. Например: str = "programming"  
str[2] ще бъде ‘o’

3.Булева- Булева проверка на изявлението дали е вярно или невярно. Предоставя две вградени стойности, true и false и принадлежи към клас bool. Истината е представена с ненула или 'T', а false е представена с нула или 'F'.

Пример, print(“Type:”, type(True))

List – в Python няма масиви, а има листове.Разликата е, че листът може да съдържа елементи от различен тип.

|  |  |
| --- | --- |
| 1  2 | list = [ 'asd', 'dog', '1337']  list.append('cat') # добавя 'cat' в края на листа |

list[3] ще бъде ‘cat’

4.Tuple – или комплект, същото като лист, но е read-only, т.е. можем само да четем от него, без да променяме стойностите му.

## 5.Dictionary – имат ключове (keys) и стойности (values), като на всеки ключ може да съответстват много стойности.Използват се къдрави скоби, за да се създаде. Ключът е примитивен тип данни и стойността съдържа обекта на Python. Подобно на хеш таблицата, където ключът съдържа стойността, посочена за него.

Пример, dict = { 1:'Pyhton' , 2:'Java', 3:'C++' } “Създаване на речник

print(dict) „Отпечатване на стойности от речника

print(dict.keys()) „Отпечатване на ключови стойности

print(dict.values()) „Отпечатване на стойности, свързани с ключове

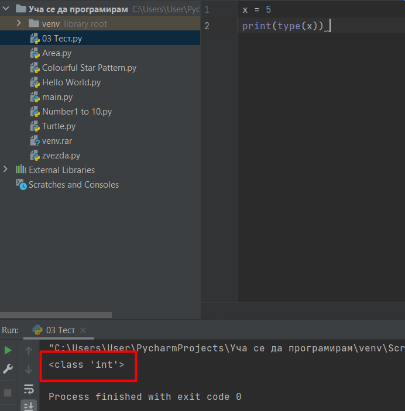
ИЗХОД:

1:'Pyhton' , 2:'Java', 3:'C++'

dict\_keys([1, 2, 3])

dict\_values([ 'Python' , 'Java' , 'C++'])

Например: dict = { 'key1':'value1', 'key2':'value2' }



# Получаване на типа данни

Можете да получите типа данни на всеки обект, като използвате type()функцията:  
Пример: Отпечатайте типа данни на променливата x:

x = 5  
print(type(x))

# Задаване на типа данни

В Python типът данни се задава, когато присвоите стойност на променлива:

|  |  |
| --- | --- |
| **Пример** | **Data Type** |
| x = "Уча се да програмирам" | str |
| x = 20 | int |
| x = 20.5 | float |
| x = 1j | complex |
| x = ["apple", "banana", "cherry"] | list |
| x = ("apple", "banana", "cherry") | tuple |
| x = range(6) | range |
| x = {"name" : "John", "age" : 36} | dict |
| x = {"apple", "banana", "cherry"} | set |
| x = frozenset({"apple", "banana", "cherry"}) | frozenset |
| x = True | bool |
| x = b"Hello" | bytes |
| x = bytearray(5) | bytearray |
| x = memoryview(bytes(5)) | memoryview |
| x = None | NoneType |

# Упражнение:

Следният примерен код ще отпечата типа данни на x, какъв тип данни ще бъде това?

|  |  |
| --- | --- |
| x = ["apple", "banana", "cherry"]  print(type(x)) | Click or tap here to enter text. |
| x = ("apple", "banana", "cherry")  print(type(x)) | Click or tap here to enter text. |
| x = 20.5  print(type(x)) | Click or tap here to enter text. |
| x = "„Уча се да програмирам“"  print(type(x)) | Click or tap here to enter text. |
| х = 5 print (type (x)) | Click or tap here to enter text. |
| x = {"name" : "John", "age" : 36}  print(type(x)) | Click or tap here to enter text. |
| x = True  print(type(x)) | Click or tap here to enter text. |
| Въведете правилния синтаксис, за да преобразувате x в цяло число.  *x = 5.5*  *x =*Click or tap here to enter text. *(x)* | Въведете правилния синтаксис, за да преобразувате x в число с плаваща запетая.  *x = 5*  *x =* Click or tap here to enter text.*(x)* |
| Въведете правилния синтаксис, за да преобразувате x в комплексно число.  *x = 5*  *x =* Click or tap here to enter text.*(x)* | [!](https://www.w3schools.com/python/exercise.asp?filename=exercise_strings1) Използвайте **len** метода, за да отпечатате дължината на низа.  x = "„Уча се да програмирам“"  print()Click or tap here to enter text. |
| Вземете първия знак от низа txt.  txt = "Уча се да програмирам"  x =Click or tap here to enter text. | Вземете знаците от индекс 2 до индекс 4 ( llo).  *txt = "Уча се да програмирам"*  *x =*Click or tap here to enter text. |
| Връща низа без интервали в началото или края.  *txt = " Уча се да програмирам "*  *x =*Click or tap here to enter text. | Преобразувайте стойността на txt в главни букви. *txt = " Уча се да програмирам " x =*Click or tap here to enter text. |
| Преобразувайте стойността на txtв малки букви.  *txt = " Уча се да програмирам " x =*Click or tap here to enter text. | txt = " Уча се да програмирам " x =Click or tap here to enter text. |
| Заменете знака Hс J.  *txt = "Hello World"*  *txt = txt.*Click or tap here to enter text. *(*Click or tap here to enter text.*,*Click or tap here to enter text. *)* | Вмъкнете правилния синтаксис, за да добавите контейнер за параметъра възраст. *age = 36*  *txt = "My name is John, and I am* Click or tap here to enter text.*"*  *print(txt.form at(age))* |
| [!](https://www.w3schools.com/python/exercise.asp?filename=exercise_booleans1) Коя булева стойност ще се отпечата? *print(10 > 9)*Click or tap here to enter text. | print(10 == 9)Click or tap here to enter text. |
| print(10 < 9)Click or tap here to enter text. | print(bool("abc"))Click or tap here to enter text. |
| print(bool(0))Click or tap here to enter text. | Умножете 15 с 2 и отпечатайте резултата.  print(15 Click or tap here to enter text.2) |
| Разделете 10 на 5 и отпечатайте резултата.  print(10Click or tap here to enter text. 5) | Използвайте правилния оператор за членство, за да проверите дали „ябълка“ присъства в fruitsобекта.  fruits = ["apple", "banana"]  if "apple" Click or tap here to enter text. fruits:  print("Yes, apple is a fruit!") |
| Използвайте правилния оператор за сравнение, за да проверите дали 5 не е **равно** на 10.  if 5 10Click or tap here to enter text.:  print("5 and 10 is not equal") | Използвайте правилния логически оператор, за да проверите дали поне един от двата израза е True. if 5 == 10 Click or tap here to enter text.4 == 4:  print("At least one of the statements is true") |

<https://www.w3schools.com/python/python_datatypes.asp>

<https://python-book.softuni.bg/chapter-02-simple-calculations.html>