

Dátové hodnoty

Boolovské hodnoty

Boolovské hodnoty True a False sa používajú hlavne na testovanie a môžete ich kombinovať spojkami or, and alebo not.

Čísla

+, -, *, / - základné operácie, % zvyšok po delení, ** mocnina

Reťazce

Ohraničené ' ', " alebo "" (pre viac riadkov)

Spájanie a opakovanie `s + a *`: `"Python is c" + "o" * 3 + "l" = "Python is cool"`

Prevod hodnôt na reťazec: `str(3.14)`

Prevod reťazca na číslo: `int(10) + float(3.14)`

Dĺžka reťazca: `len(s)`

Prevod na malé/veľké písmena: `s.lower()` `s.upper()`

Odstránenie prázdnych znakov zo začiatku a konca: `s.strip()`

Nahradenie podreťazca: `s.replace(old, new)`

Formátovanie správ

"Ahoj {0}, dnes je {1}".format("Peter", "pondelok")

Dosadí hodnoty argumentov do polí formátovacieho reťazca, výsledný reťazec je „Ahoj Peter, dnes je pondelok“.

Zarovnanie textu na 20 znakov vpravo/vľavo/na stred: `{0:>20}` `{0:<20}` `{0:^20}`

Formátovanie čísel na 3 desatinné miesta: `{0:.3f}`

Zoznamy

Vytvorenie zoznamu: `cisla = ["nula", 1, 2, 3]`

Počet prvkov: `len(cisla)`

Indexovanie od 0, priradenie hodnoty: `zoznam[index] = hodnota`

Mazanie prvkov: `del zoznam[index]`

Pridávanie prvkov na koniec/danú pozíciu: `zoznam.append(prvok)`

`zoznam.insert(index, prvok)`

Usporiadanie prvkov: `zoznam.sort()`

Prehodenie poradia prvkov: `zoznam.reverse()`

Vnorené zoznamy: `zoznam = [{"jedna", "dva"}, [1, 2, 3]]`

`zoznam[0][1] == "jedna"`

Rozsahy hodnôt

`zoznam[od (vrátane): do (bez)]`, napr.:

`cisla[1:3]` – od druhého po tretí prvok

`cisla[2:]` – od tretieho do konca zoznamu

Záporné čísla pre indexovanie od konca, napr:

`cisla[1:-1]` – od druhého po predposledný prvok

`cisla[-2:]` – posledné dva prvky

Príkazy

Vetvenia

príkaz if s voliteľnou vetvou else, alebo elif pre zretáženie, napr:

```
if cislo < 0:
    print("negativne")
elif cislo == 0:
    print("nula")
elif cislo == 1:
    print("jeden")
else:
    print("viac")
```

Cykly

Príkaz while:

```
i = 0
while i < 10:
    print(i)
    i += 1
```

Príkaz for:

```
for x in range(10):
    print(x)
```

Funkcia range generuje postupnosť čísel od (vrátane) do (bez) s daným krokom, napr.:

range(10) – od 0 do 9

range(1, 5) – od 1 do 4

range(5, 0, -1) - vygeneruje postupnosť v opačnom poradí od 5 do 1 (bez 0)

Príkaz break pre prerušenie cyklu

Funkcie

```
def mocnina(x, y=2):
    return x ** y
```

Príkaz return pre návrat z funkcie a vrátenie hodnoty

y=2 prednastavená hodnota argumentu

Volanie funkcie

Pozičný zoznam argumentov alebo podľa názvu, argumenty s prednastavenou hodnotou je možné vynechať

Napr.:

cislo = mocnina(4, 3) – argument x = 4, y = 3

cislo = mocnina(y=2, x=3) – argumenty podľa názvu, nemusia byť v takom poradí ako sú definované vo funkcii

cislo = mocnina(4) – argument x = 4, y = 2 (prednastavená hodnota)