

Python – lekce 7

Řetězce

Práce s řetězci je jednou z velmi silných stránek jazyka Python. Řetězce jsou věci, které se uzavírají do uvozovek. Jednoduchých, dvojitých nebo trojitých. Je to celkem jedno, jen na začátku a konci musí být stejné. Trojitě se používají u dlouhých textů, které přesahují přes několik řádků.

Program 01

```
print("Superstar", 'SuperStar' , ""SuperStar"", """"SuperStar""")
print("""Žít
a
nechat žít""")
```

Řetězce jsou neměnné. Pokud je dáme do proměnné, tak to může vypadat, že se mění, ale ve skutečnosti vždy vzniká nová proměnná se stejným jménem, ale jiným obsahem.

Existují tzv. escape znaky, sekvence, které slouží hlavně ke zkracování kódu a zpřehledňování tisku. Jsou to hlavně:

- \n – Enter, nový řádek
- \t – tabulátor

Program 02

```
print("Dny v týdnu \nPondělí \nÚterý \n \na tak dále....\n \tNeděle")
```

Formátovací operátor %

- slouží ke vkládání a formátování různých datových typů do řetězce

Obecně:

```
"...%zkratka typu ..." % (N-tice hodnot)
```

Zkratky typů:

%i celé číslo

%f desetinné

%s řetězec

%x hexadecimální tvar

%o osmičkový tvar

%5.2f číslo bude složeno z 5 znaků, z toho 2 budou za desetinnou tečkou, chybějící znaky budou nahrazeny mezerami zleva

%-8.2f číslo bude složeno z 8 znaků, z toho 2 budou za desetinnou tečkou, chybějící znaky budou nahrazeny mezerami zprava

%4i, %-4i obdobně pro celá čísla

Program 03 (po částech, math až u posledního příkladu)

```

from math import *

z=24
x="Celé %4i desetinné %-8.2f text" % (z,z)
print(x)

w="Řetězec %s řetězec" % 'vložený text'
print(w)

y="Hexadecimální %x oktalové %o" % (z,z)
print(y)

x="Zobrazení Pí na tři desetinná místa: %7.3f ***" %(pi)
y="Zobrazení Pí na tři desetinná místa: %-7.3f ***" %(pi)
print(x)
print(y)

```

Indexování

K jednotlivým částem řetězce lze přistoupit pomocí [].
Kladné hodnoty číslují znaky zleva, záporné zprava.

Program 04 (první část)

```

ret="pátek"
print(ret[-1])

```

Operátor : (slice)

- vrací podřetězec
[:n] - vrátí prvních n znaků
[n:] - vrátí podřetězec od pozice n do konce
[m:n] - vrátí podřetězec od pozice m do n

Program 04 (druhá část)

```

print(ret[:3])
print(ret[-3:])
print(ret[1:3])

```

Funkce str()

- převede libovolný typ na řetězec

```

x=int(input("zadej číslo "))
print( "Zadali jste "+str(x))

```

Základní operace

Řetězce můžeme:

- spojovat = sčítat = +
- klonovat = násobit celým číslem = *
- můžeme také zjišťovat jejich délku = len(retezec) = kolik obsahuje znaků

Program 05

```
pozdrav="ahoj"
pozdrav2= pozdrav+" Eliško"
print(pozdrav2)
print(pozdrav*10)

print(len(pozdrav))                # kolik je v řetězci znaků?
```

Procházení větou a provádění operace nad znaky

Pro procházení jednotlivými znaky v řetězci se používá cyklus FOR.

Program 06

```
veta="Kobyla má malý bok"
for znak in veta:
    print(znak)
    if znak=="a":
        print("Našel jsem písmeno a")
```

Cyklus prochází jednotlivými znaky, postupně je pod sebe tiskne, a když (IF) narazí na znak "a", tak něco vykoná.

Na jednoduché zjišťování přítomnosti písmene stačí IF. Pokud bychom ale již chtěli vědět, kolikrát tam je, musíme použít výše uvedený cyklus FOR nebo nějakou funkci.

Program 07

```
veta="Kobyla má malý bok"
if "m" in veta:
    print("Věta obsahuje alespoň jedno písmeno 'm'")
```