

Čitelný kód, if-else

E 3011

Jan Böhm

RECETOX

March 13, 2024

Co nás dnes čeká

1 jak psát čitelný kód

2 If-else

3 Mysteries & errors

4 Opakování matematiky

Jak pojmenovávat objekty

Vždy záleží na dohodě. Obvykle se tím dohodne na nějakých standardech, které je třeba dodržovat. Dnes takové standardy nastavíme:

- budeme používat angličtinu a nebude používat speciální znaky
- víceslovné názvy funkcí, proměnných – lowCamelCase
- jména musí vystihovat, co proměnná znamená / co funkce dělá, např. `dateOfBirth`, `dotProduct(vec1, vec2)`
- zkratky jsou ok, pokud jsou pochopitelné
- dlouhé názvy jsou lepší než nepochopitelné zkratky
- nepoužívat jednopísmenné proměnné s výjimkou indexů (např. `i`, `j` jako indexy matice jsou ok)
- konstanty – KAPITALKY (`UCD = 408849`)

Formátování kódu

Tady je několik pravidel, které pomohou, aby byl kód čitelný

- maximálně 80 znaků na řádku
- mezery za , ;
- mezera před i za binární operací +, -, /, %, ... nebo okolo přiřazení =
- v případě závorek nepoužívat mezery u nejvyšší priority
- nepoužívat zbytečně mezery
- zalamovat řádky tak, aby začínaly binárním operátorem
- více parametrů zalamovat tak, aby začínaly na stejném řádku
- před novou funkcí mít 2 volné řádky
- použít prázdný řádek pro oddělování větších kroků v programu
- pro odsazení používat **4 mezery** (hlídá IDE)

Ukázky dobře (a špatně) formátovaného kódu

```
1 # Comments can be written using hashtag
2 # Anything beyond hashtag Python doesn't see
3 name = "Jan" # comment can be inline
4
5 # Not recommended
6 x = "Jan Bohm" # Teacher
7
8 # Recommended
9 teacher = "Jan Bohm"
10
11 # Not recommended
12 x = 'Mgr. Jan Bohm'
13 a, b, c = x.split()
14 print(c, b, sep=", ")
15
16 # Recommended
17 person = "Mgr. Jan Bohm"
18 title, name, surname = person.split()
19 print(surname, name, sep = ", ")
```

Ukázky dobře (a špatně) formátovaného kódu

```
1 # Not recommended
2 def db(x):
3     return(2*x)
4
5
6 # Recommended
7 def double(x):
8     return(2*x)
9
10
11 # Not recommended
12 def dummyFunction(argument1, argument2, argument3, argument4):
13     pass
14
15
16 # Recommended
17 def dummyFunction(argument1, argument2,
18                   argument3, argument4):
19     pass
```

Ukázky dobře (a špatně) formátovaného kódu

```
1 # Not recommended
2 total = (number1 + number2 +
3         number3 + number4)
4
5 # Recommended
6 total = (number1 + number2
7         + number3 + number4)
8
9 # Not recommended
10 name="Jan"
11 importantNumber=importantNumber*3
12 x=(-b+D**(1/2))/2a
13 x=( -b + D**(1 / 2) ) / ( 2 * a)
14
15 # Recommended
16 name = "Jan"
17 importantNumber = importantNumber * 3
18 x = (-b + D**(1/2)) / (2*a)
```

Ukázky dobře (a špatně) formátovaného kódu

```
1 # Not recommended
2 if x * y > 0 or y * z > 0 or x * z > 0:
3     pass
4
5 # Better
6 if x*y > 0 or y*z > 0 or x*z > 0:
7     pass
8
9 # Recommended
10 cond1 = x*y > 0
11 cond2 = y*z > 0
12 cond3 = x*z > 0
13 if cond1 or cond2 or cond3:
14     pass
```


Ukázky dobře (a špatně) formátovaného kódu

```
1 # Not recommended
2 def solveQuadraticEquation(a,b,c):
3     pass
4
5 # Recommended
6 def solveQuadraticEquation(a, b, c):
7     pass
```

Struktura skriptu

Skripty strukturujeme následovně

- 1 Úplně v hlavičce v komentáři popis, o co se jedná
- 2 Následují importy z knihoven
- 3 Poté definice vlastních funkcí
- 4 Samotný skript

V případě, že vlastních funkcí je hodně, můžeme je mít v jednom skriptu, např. `functions.py` a naimportovat je jako knihovnu `import functions` (oba skripty musí být ve stejném adresáři). Takto to budeme dělat...

Co nás dnes čeká

1 jak psát čitelný kód

2 If-else

3 Mysteries & errors

4 Opakování matematiky

If-else

If else je základní řídicí struktura kódu. Funguje takto:

if *condition* **then**

```
| doSomething();      /* Runs only if condition is True */
```

else

```
| doSomethingElse(); /* Runs only if condition is False */
```

If-else

```
1 if x < y:
2     print("x is less than y")
3 else:
4     if x > y:
5         print("x is greater than y")
6     else:
7         print("x and y are equal")
```

If-else

```
1 if x < y:  
2     print("x is less than y")  
3 else:  
4     if x > y:  
5         print("x is greater than y")  
6     else:  
7         print("x and y are equal")
```

```
1 if x < y:  
2     print("x is less than y")  
3 elif x > y:  
4     print("x is greater than y")  
5 else:  
6     print("x and y are equal")
```

Co nás dnes čeká

1 jak psát čitelný kód

2 If-else

3 Mysteries & errors

4 Opakování matematiky

Zadání

Následuje několik ukázek kódu v pythonu. Pokud je v kódu chyba, poznamenejte si kde. Pokud je kód správně, запиšte si, co vypíše.

Mysteries & errors #1

```
1 number1 = 1
2 number2 = -1
3
4 if number1 = number2:
5     print("Equal")
6 else:
7     print("Not equal")
```

Mysteries & errors #2

```
1 n = 1
2 while n > 0:
3     n /= 10
4
5 print("DONE")
```

Mysteries & errors #3

```
1 for n in range(6):  
2     if n % 2 == 0:  
3         print("n is even")
```

Mysteries & errors #4

```
1 for n in range(6):  
2     if 12 % n == 0:  
3         print("12 is divisible by ", n)
```

Mysteries & errors #5

```
1 number1 = "4"  
2 number2 = "-3"  
3 number3 = number1 + number2  
4  
5 print(number3)
```

Mysteries & errors #6

```
1 print("Red")  
2     print("Green")
```

Mysteries & errors #7

```
1 number = "408849"  
2 result = 0  
3  
4 for n in number:  
5     result += number  
6  
7 print(result)
```

Mysteries & errors #8

```
1 cond1 = "True"
2 cond2 = "False"
3
4 if cond1 and cond2:
5     print("Red")
6 else:
7     print("Green")
```


Mysteries & errors #9

```
1 x = 10
2
3 def plusOne():
4     x += 1
5     print(x)
6
7 plusOne()
```

Mysteries & errors #10

```
1 x = 10
2
3 def dummy():
4     x = 5
5     return(x)
6
7 dummy()
8 print(x)
```

Mysteries & errors #11

```
1 def half(x):  
2     print(x/2)  
3  
4 x = 32  
5 for i in range(5):  
6     x = half(x)  
7     print(x)
```

Mysteries & errors #12

```
1 animal = cat
2
3 if animal == "dog":
4     print("Good boi")
```

Co nás dnes čeká

- 1 jak psát čitelný kód
- 2 If-else
- 3 Mysteries & errors
- 4 Opakování matematiky

Definice prvočísla

Prvočíslo je číslo, které má právě 2 dělitele - sebe samo a jedničku.

Definice prvočísla

Prvočíslo je číslo, které má právě 2 dělitele - sebe samo a jedničku.

- Prvočíselnost řešíme pouze u celých kladných čísel.
- Jediné sudé prvočíslo je 2.
- 1 není prvočíslo.

Desítková soustava

Zápis 408849 znamená

$$408849 = 4 \cdot 10^5 + 0 \cdot 10^4 + 8 \cdot 10^3 + 8 \cdot 10^2 + 4 \cdot 10^1 + 9 \cdot 10^0$$

Dvojková soustava

Zápis 11010_2 znamená

$$11010_2 = 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 26_{10}$$

Jak převádět mezi soustavami?

Převod mezi soustavami

Pokud chci číslo x (např. 26) vyjádřit ve dvojkové soustavě, musím:

- Najít největší K takové, že $2^K \leq x$ (tj. $K = 4$)
- Pro všechna $k \in \{K, K - 1, \dots, 1, 0\}$:
 - Spočítat $c = x \div 2^k$ (to je nová cifra, $c = 26 \div 2^4 = 1$)
 - Zmenšit $x \leftarrow x \bmod c$ (v tomto příkladu $x \leftarrow 10$)

Pozn: \div značí celočíselné dělení.

k	4	3	2	1	0
2^k	16	8	4	2	1
x	26	10	2	2	0
c	1	1	0	1	0