ÚVOD

Tvůj první projekt

Pokud si chceš vyzkoušet nabité znalosti, nyní máš skvělou příležitost. Čeká na tebe **první praktický projekt**, kde si můžeš svoje dovednosti aplikovat.

Před tím, než začneš pracovat na projektu, čekají tě ještě 3 mezizastávky:

1. **Formátování zápisu** aneb jak psát čistý kód,
2. základy **verzování** v Git,
3. opakovací **kvíz**.

FORMÁTOVÁNÍ ZÁPISU

Pár pravidel na začátek

Aby mohli všichni programátoři pohodlně pracovat s **cizím** kódem a snadno jej **číst a pochopit**, musí se řídit nějakými **doporučeními**, nebo **vzory**.

Stejně jako u vaření dosáhneš často nejlepších výsledků tehdy, pokud dodržuješ kroky v receptu.

Soubor těchto **pravidel** pro *čistý kód* můžeš najít v oficiální dokumentaci zde. Nicméně pravidel je tam více, než v tento moment dovedeš uplatnit. Proto si nyní ukážeme ty nejpodstatnější.

Délka řádku

Maximální délka řádku by měla být 79 znaků. Pokud bude tvůj řádek delší, můžeš jej zalomit nebo rozdělit.

Níže napsaný `tuple` je příliš dlouhý a tudíž velice nepraktický pro čtení:

Ukázka kódu **1** ZKOPÍROVAT KÓD

```
cities = ("Prague", "Brno", "Ostrava", "Plzeň", "Liberec", "Olomouc", "Ces
```

V takovém případě je nejlepší hodnoty zapsat pod sebe:

Ukázka kódu **2** ZKOPÍROVAT KÓD

```
cities = (
    "Prague",
    "Brno",
    "Ostrava",
    "Plzeň",
    "Liberec",
    "Olomouc",
    "České Budějovice",
    "Hradec Králové",
    "Ústí nad Labem",
    "Pardubice"
)
```

Pokud je ovšem i výpis pod sebou příliš dlouhý, můžeš zvolit variantu několika hodnot na řádek:

Ukázka kódu **3** ZKOPÍROVAT KÓD

```
cities = (
```

```
    "Prague", "Brno", "Ostrava",
    "Plzeň", "Liberec", "Olomouc",
    "České Budějovice", "Hradec Králové", "Ústí nad Labem",
    "Pardubice"
)
```

Odsazování

Můžeš používat **4 mezery** nebo **1 tabulátor**, ale nikdy nekombinovat. Prostě si vyber jednu variantu a tu konzistentně používej:

Ukázka kódu **4** ZKOPÍROVAT KÓD

```
if name == "Matouš":
    print("Ahoj Matouši")
else:
    print("Ahoj všem!")
```

Mezery v zápise

Python ti dovolí napsat mezery prakticky kdekoliv, ale to neznamená, že je to správně. Ukážeme ti nyní několik variant.

Mezery **patří** za datový oddělovač čárku:

Ukázka kódu **5** ZKOPÍROVAT KÓD

```
# špatně
print(cities[1],cities[2],cities[3])

# správně
print(cities[1], cities[2], cities[3])
```

Mezery **nepatří** mezi závorky funkcí a jméno funkce:

Ukázka kódu **6** ZKOPÍROVAT KÓD

```
# špatně
print (cities[1])

# správně
print(cities[1])
```

Jedna mezera patří okolo přiřazovacího operátoru `=` :

Ukázka kódu **7** ZKOPÍROVAT KÓD

```
# špatně
jmeno = "Lukáš"
email = "lukas@gmail.com"
vek   = 30

# správně
jmeno = "Lukáš"
email = "lukas@gmail.com"
vek = 30
```

Správně zapsané podmínky

Pokud ověřuješ hodnoty typu `bool`, nebo potřebuješ zkontrolovat jestli není `set` prázdný, potom není vhodné použít srovnávací operátory `==` a `!=` :

Ukázka kódu **8** ZKOPÍROVAT KÓD

```
# špatně
if registered == True:
    # ...

# lepší řešení
if registered is True:
    # ...

# správně
if registered:
    # ...
```

Pro funkci `len()` :

Ukázka kódu **9** ZKOPÍROVAT KÓD

```
# špatně
if len(cities) != 0:
    # ...

# správně
```

```
6  if not len(cities):
7      # ...
```

OPAKOVACÍ KVÍZ

Kvíz

KVÍZOVÉ OTÁZKY

Opakování

... je matka moudrosti!
Pojď si proto naučenou látku znovu projít a otestovat, že všemu rozumíš.

SPUSTIT KVÍZ

U každé otázky musíte zvolit alespoň jednu, nebo více možností.
U některých otázek je potřeba zvolit správnou kombinaci více odpovědí.

AKADEMIE

Popis projektu

V tomto projektu bude tvým cílem vytvořit *textový analyzátor* - program, který se bude umět prokousat libovolně dlouhým textem a zjistit o něm různé informace.

Ještě než začneš, budeš pracovat se zadanými **předpřipravenými texty**. Kód se ti pak bude lépe kontrolovat. Tyto texty jsou dostupné **zde**.

Tvůj program bude obsahovat následující:

1. Na úvod si svůj soubor **popiš hlavičkou**, ať se s tebou můžeme snadněji spojit:

Ukázka kódu **10** ZKOPÍROVAT KÓD

```
"""
projekt_1.py: první projekt do Engeto Online Python Akademie

author: Petr Svetr
email: petr.svetr@gmail.com
discord: Petr Svetr#4490
"""
import ...
```

2. Vyžádá si od uživatele **přihlašovací jméno** a **heslo**,
3. zjistí, jestli zadané údaje odpovídají někomu z **registrovaných uživatelů**,
4. pokud **je registrovaný**, pozdrav jej a umožni mu **analyzovat texty**,
5. pokud **není registrovaný**, upozorni jej a ukonči program.**

Registrováni jsou následující uživatelé:

```
+------+-------------+
| user |   password  |
+------+-------------+
| bob  |     123     |
| ann  |   pass123   |
| mike | password123 |
| liz  |   pass123   |
+------+-------------+
```

5. Program nechá uživatele vybrat mezi třemi texty, uloženými v proměnné `TEXTS`:

1. Pokud uživatel vybere takové číslo textu, které **není v zadání**, program jej upozorní a skončí,
2. pokud uživatel zadá **jiný vstup** než číslo, program jej rovněž upozorní a skončí.

6. Pro vybraný text spočítá následující **statistiky**:
    1. počet slov,
    2. počet slov začínajících velkým písmenem,
    3. počet slov psaných velkými písmeny,
    4. počet slov psaných malými písmeny,
    5. počet čísel (ne cifer),
    6. sumu všech čísel (ne cifer) v textu.
7. Program zobrazí jednoduchý **sloupcový graf**, který bude reprezentovat četnost různých délek slov v textu. Například takto:

```
# ...
 7| * 1
 8| ********** 11
 9| *************** 15
10| ********* 9
11| ********** 10
```

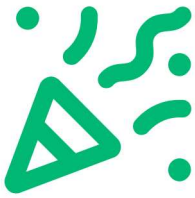
Po spuštění by měl průběh vypadat následovně:

```
$ python projekt1.py
username:bob
password:123
----------------------------------------
Welcome to the app, bob
We have 3 texts to be analyzed.
----------------------------------------
Enter a number btw. 1 and 3 to select: 1
----------------------------------------
There are 54 words in the selected text.
There are 12 titlecase words.
There are 1 uppercase words.
There are 38 lowercase words.
```

```
There are 3 numeric strings.
The sum of all the numbers 8510
----------------------------------------
LEN|  OCCURENCES   |NR.
----------------------------------------
  1|*              |1
  2|*********      |9
  3|******         |6
  4|***********    |11
  5|************   |12
  6|***            |3
  7|****           |4
  8|*****          |5
  9|*              |1
 10|*              |1
 11|*              |1
```

Pokud uživatel **není registrovaný**:

```
$ python projekt1.py
username:marek
password:123
unregistered user, terminating the program..
```



Gratulujeme! Máš hotový tento studijní materiál.

ZPĚT DO AKADEMIE