



Programování 1 (NPRG030)

Cvičení 10: Práce se soubory

Shrnutí přednášky: Práce se soubory

```
1 # Práce se soubory
2 soubor = open(nazev_souboru, rezim) # Otevře soubor v zadaném režimu (r, w, a, x, b, t, +)
3 soubor = open(nazev_souboru, rezim, encoding) # Otevře soubor navíc se určeným kódováním znaků
4 soubor.read() # Přečte celý obsah souboru jako jediný řetězec
5 soubor.read(pocet) # Přečte určitý počet znaků (nebo bytů) ze souboru
6 soubor.readline() # Přečte jediný řádek ze souboru
7 soubor.write(text) # Zapiše text do souboru
8 soubor.close() # Uzavře soubor, aby byl dostupný pro další operace
9 for radek in soubor: # Iteruje přes soubor a postupně čte jeho řádky v cyklu for
```

```
10 # Správce kontextu
11 with open("soubor.txt", "r", encoding="utf-8") as soubor:
12     nactena_data = soubor.read()
```

```
13 # Čtení a zápis do JSON souboru (balíček json)
14 json.loads(retezec) # Převede JSON text na slovník obsahující dvojice (klíč, hodnota)
15 json.dumps(slovník) # Převede slovník na JSON text
```

```
16 # Čtení a zápis do CSV souboru (balíček csv)
17 ctecka = csv.DictReader(soubor) # Inicializuje čtení CSV souboru
18 for radek in ctecka: # Postupné čtení CSV souboru po řádcích
19     zapisovac = csv.DictWriter(soubor, nazvy_sloupcu) # Inicializuje zápis s určenými sloupci
20     zapisovac.writerow(object) # Zapiše řádek do CSV souboru
```

Příklad 10.3: Kontrola existence souboru

- ❖ Vytvořte *funkci* v jazyce Python, která *vypíše obsah souboru* na standardní výstup, pokud soubor existuje
 - ❖ Parametrem funkce bude (relativní) cesta k souboru
 - ❖ Pro ověření existence souboru můžete využít třídu `path` z knihovny `os`
 - ❖ Nebo vhodně ošetřete výjimku `FileNotFoundError`

Cvičení 10.4: Deserializace z JSON souboru

- ❖ Vytvořte *program* v jazyce Python, který *deserializuje obsah JSON souboru* obsahující seznam osob
 - ❖ Pro jednoduchost uvažujte, že každá osoba vždy obsahuje jméno, příjmení, email a telefon
 - ❖ Každou osobu reprezentujte jako slovník s klíči odpovídající názvům atributů
- ❖ Vypište načtené osoby na standardní výstup ve tvaru

```
Osoba: [Jmeno] [Prijmeni]  
Email: [Email]  
Telefon: [Telefon]
```

❖ Těžší varianta

- ❖ Každou osobu reprezentujte jako objekt třídy `OsobaDTO` s odpovídajícími atributy
- ❖ Zvolte vhodný datový typ u každého atributu a vytvořte metodu `__str__`
- ❖ Pro *zjednodušení deserializace* můžete použít např. *knihovnu* `json`



Cvičení 10.5: Serializace objektů do JSON souboru

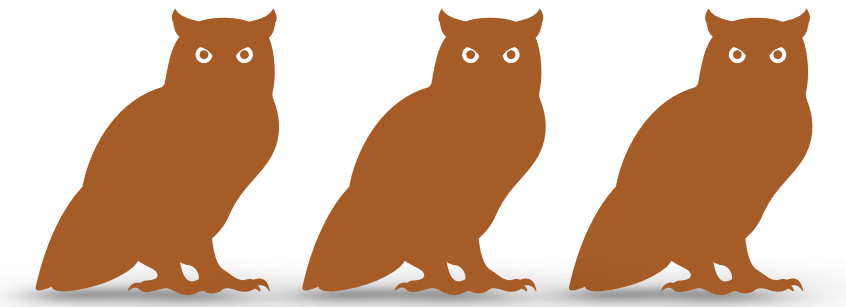
- ❖ Vytvořte *program* v jazyce Python, který *serializuje seznam objektů* reprezentujících Osoby *do JSON souboru*
- ❖ Pro jednoduchost uvažujte, že každá osoba vždy obsahuje atributy jméno, příjmení, email a telefon

Cvičení 10.6: Serializace a Deserializace CSV

- ❖ Vytvořte *program* v jazyce Python, který:
- ❖ *Deserializuje obsah CSV* souboru obsahující seznam osob
- ❖ *Serializuje seznam osob do CSV souboru*
- ❖ Pro jednoduchost uvažujte, že každá osoba vždy obsahuje atributy jmeno, prijmeni, email a telefon
 - ❖ Každou osobu reprezentujte jako slovník s klíči odpovídající názvům atributů
- ❖ Vypište načtené osoby na standardní výstup ve tvaru

```
Osoba: [Jmeno] [Prijmeni]  
       Email: [Email]  
       Telefon: [Telefon]
```

Příklad 10.7: Práce s databází PostgreSQL



- ❖ Vytvořte *program* v jazyce Python, který umožňuje *vytvářet*, *ukládat a načítat* informace o osobách do/z *databáze PostgreSQL*
- ❖ Pro jednoduchost u každé osoby předpokládejte pouze tyto atributy:
 - ❖ jmeno, prijmeni, email, telefon
- ❖ Každou osobu reprezentujte pomocí třídy `OsobaDT0`

Jak psát kód: Relativní cesty k souborům

- ❖ Vyhněte se pevnému určení cest k souborům
 - ❖ Pro ukládání cest k souborům *používejte relativní cesty* nebo konfigurační soubory

```
1 # Pevně určená cesta
2 file = open('/path/to/file.txt', 'r')
```

```
1 # Relativní cesta
2 file = open('config/file.txt', 'r')
```

Jak psát kód: Správce kontextu

- ❖ Použití správců kontextu (angl. *context managers*) pro správu prostředků
 - ❖ Pro správnou správu prostředků používejte správce kontextu (příkaz `with`)

```
1 # Bez použití správce kontextu
2 file = open('data.txt', 'r')
3 data = file.read()
4 file.close()
```

```
1 # S použitím with
2 with open('data.txt', 'r') as file:
3     data = file.read()
```

Reference

Práce se soubory

- ❖ https://www.w3schools.com/python/python_file_handling.asp
- ❖ https://www.w3schools.com/python/python_file_open.asp
- ❖ https://www.w3schools.com/python/python_file_write.asp
- ❖ https://www.w3schools.com/python/python_file_remove.asp

CSV, JSON

- ❖ https://www.w3schools.com/python/python_json.asp