

## Лабораторная работа №4

### Работа с JSON-файлами

json-файл представляет собой следующую структуру:

```
{
  "Books": {
    "Book": [
      {
        "id":1,
        "Title": "Война и мир",
        "Author": "Л.Н. Толстой"
      },
      {
        "id":2,
        "Title": "Мертвые души",
        "Author": "Н.В. Гоголь"
      },
      {
        "id":3,
        "Title": "Евгений Онегин",
        "Author": "А.С. Пушкин"
      }
    ]
  }
}
```

Для работы с такими структурированными данными в Python используется библиотека

```
import json
```

Для считывания структуры и содержимого файла:

```
with open('d:/Python/book.json', 'r', encoding="utf-8") as j:
    json_data = json.load(j)
    print(json_data)
```

Кодировка может отличаться.

Фрагмент кода для вывода названия всех книг:

```
for book in json_data["Books"]["Book"]:
    print(str(book["Title"]))
```

Фрагмент кода для вывода названия книги с id=2:

```
for book in json_data["Books"]["Book"]:
    if book["id"]==2:
        print(str(book["Title"]))
```

### Задания для лабораторной работы

Для вывода данных использовать таблицы (PrettyTable).

1. Используя представленный json-файл:
  - 1.1. Вывести количество универсальных и общепрофессиональных компетенций
  - 1.2. Вывести на экран коды и названия профессиональных стандартов
  - 1.3. Вывести список универсальных компетенций с кодами
  - 1.4. Вывести индикаторы для указанной пользователем универсальной компетенции

2. Создайте json-файл, содержащий сведения об продаваемых квартирах: адрес, площадь, район, стоимость продажи, этаж, дата размещения объявления.

2.1. Определить адреса квартир со стоимостью более указанной суммы

2.2. Определите среднюю стоимость 1м<sup>2</sup> квартир в указанном районе

2.3. Определите адреса квартир, расположенных на 2 этаже с площадью более 40 м<sup>2</sup>

2.4. Определите адреса квартир, объявления о продаже которых были размещены в прошлом году. Для работы с датой используйте библиотеку

```
import datetime
```

Для извлечения года из даты, использовать строку:

```
datetime.datetime.strptime(realestate["date_of_publication"], "%d/%m/%Y").year
```

Для определения формата можно использовать следующие коды:

%d: день месяца в виде числа

%m: порядковый номер месяца

%y: год в виде 2-х чисел

%Y: год в виде 4-х чисел

2.5. Определить сколько времени в днях, секундах, микросекундах прошло с даты публикации до текущей даты

Нередко при работе с датами возникает необходимость добавить к какой-либо дате определенный промежуток времени или, наоборот, вычесть некоторый период. И специально для таких операций в модуле datetime определен класс timedelta. Фактически этот класс определяет некоторый период времени.

Класс timedelta имеет несколько свойств, с помощью которых мы можем получить временной промежуток:

days: возвращает количество дней

seconds: возвращает количество секунд

microseconds: возвращает количество микросекунд

Для получения текущей даты:

```
datetime.datetime.now()
```