**Задание 1: Парсинг JSON**

Вам дано 3 файла json. Каждый файл представляет из себя структуру: {“headers”: [], “values”: []}

Необходимо распарсить все json и записать данные в один файл excel в виде таблицы, где заголовки – headers, а значения – values (каждый файл на отдельном листе excel).

У каждого элемента из headers необходимо в качестве значения использовать свойство QuickInfo, а у элементов values в качестве значения использовать свойство Text.

Порядок следования элементов внутри headers может не всегда соответствовать порядку внутри values.

Для одного json порядок следования элементов внутри headers может не всегда соответствовать порядку внутри headers другого файла (аналогично и для values).

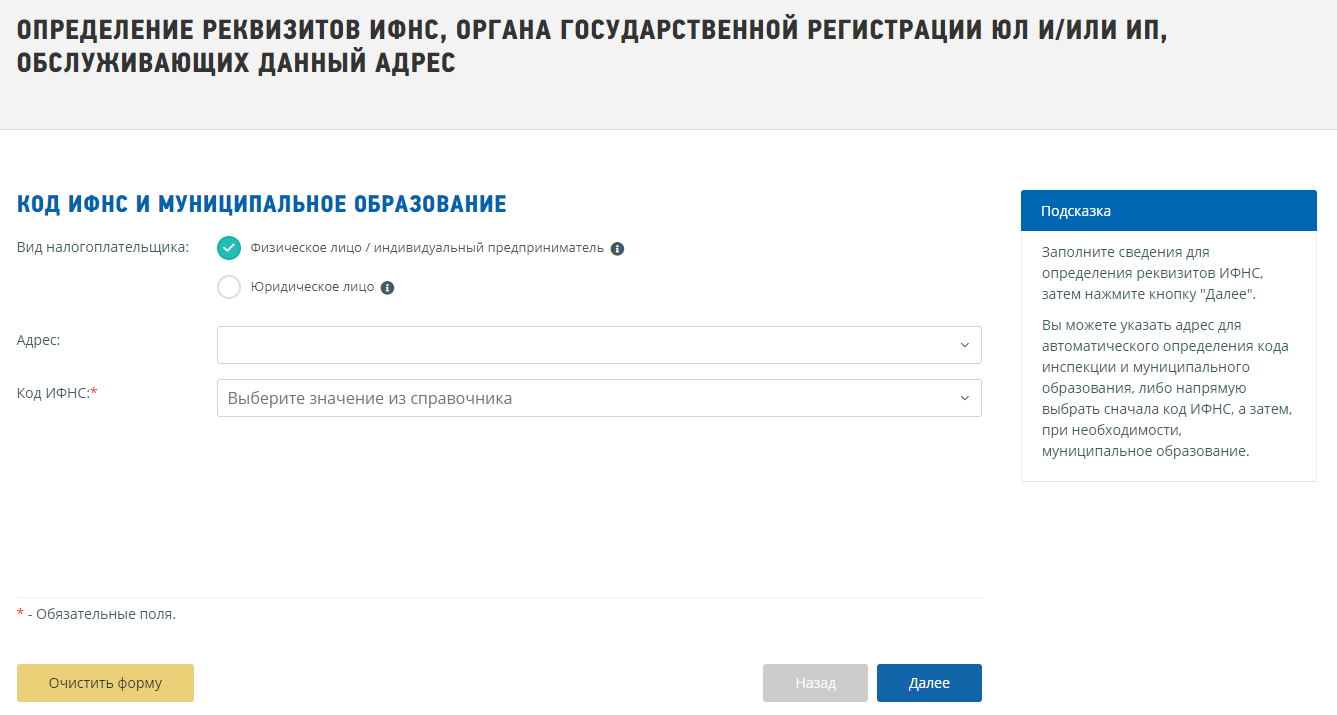
На выходе должен получиться один файл excel c 3 листами, на каждом листе должна быть таблица:

|  |  |  |  |
| --- | --- | --- | --- |
| **Сумма по ВВ** | **Внутренняя валюта** | **Код налога** | **Счет Главной книги** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

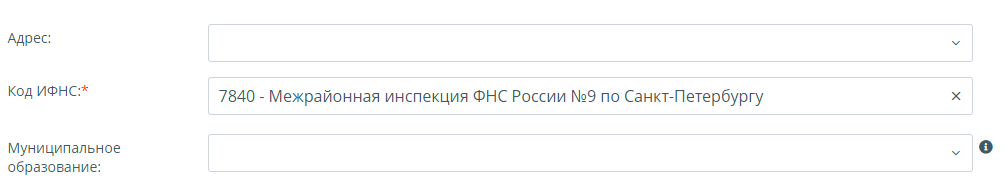
**Задание 2: REST**

Процесс «Проверка актуальных платежных реквизитов на сайте ИФНС»:

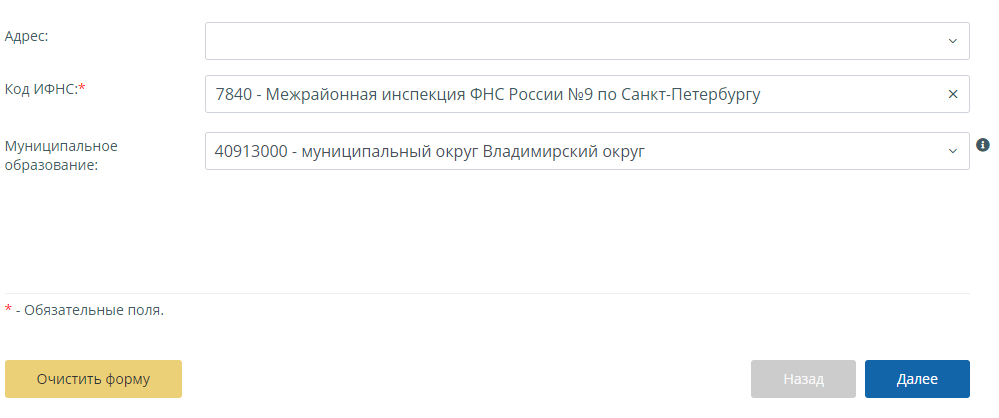
1. Пройти по ссылке <https://service.nalog.ru/addrno.do>



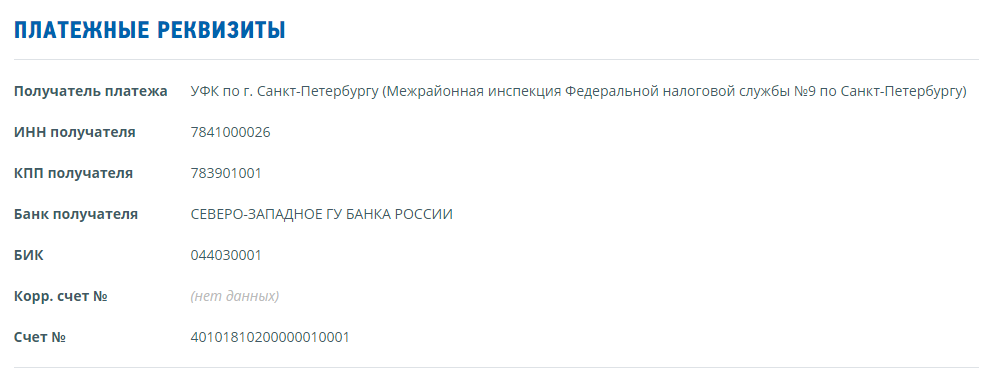
1. Раскрыть список «Код ИФНС» и в фильтре указать номер 7840, затем выбрать позицию



1. В списке «Муниципальное образование» выбрать позицию с номером 40913000



1. Нажать кнопку «Далее»
2. Запись платежных реквизитов в документ



Необходимо написать функцию с REST запросом, которая будет получать платежные реквизиты по введенным аргументам.

Входные данные: Код ИФНС (7840), ОКТМО (40913000)

Выходные данные: Массив с набором данных (Получатель платежа, ИНН получателя, КПП получателя, Банк получателя, БИК, Счет №)

**Задание 2: SQL**

Набор данных:

1. Таблица «Clients»

|  |  |
| --- | --- |
| **id\_users** | **user\_name** |
| 1 | Иван |
| 2 | Константин |
| 3 | Дмитрий |
| 4 | Александр |

1. Таблица «Products»

|  |  |  |
| --- | --- | --- |
| **id\_product** | **product\_name** | **price** |
| 1 | Мяч | 299,99 |
| 2 | Ручка | 18 |
| 3 | Кружка | 159,87 |
| 4 | Монитор | 18000 |
| 5 | Телефон | 9999,9 |
| 6 | Кофе | 159 |

1. Таблица «Orders»

|  |  |  |  |
| --- | --- | --- | --- |
| **id\_Order** | **id\_users** | **id\_product** | **Order\_name** |
| 1 | 2 | 2 | Закупка 1 |
| 1 | 2 | 5 | Закупка 1 |
| 1 | 2 | 1 | Закупка 1 |
| 2 | 1 | 1 | Закупка 2 |
| 2 | 1 | 3 | Закупка 2 |
| 2 | 1 | 6 | Закупка 2 |
| 2 | 1 | 2 | Закупка 2 |
| 3 | 4 | 5 | Закупка 3 |
| 4 | 3 | 6 | Закупка 4 |
| 4 | 3 | 3 | Закупка 4 |
| 2 | 1 | 5 | Закупка 2 |

Необходимо написать три запроса:

1. Список клиентов с общей суммой их покупок

Пример результата:

|  |  |
| --- | --- |
| **Клиент** | **Общая сумма покупок** |
| Александр | 5505,55 |
| Дмитрий | 890 |
| Иван | 3650,48 |

1. Список клиентов, которые купили телефон

Пример результата:

|  |
| --- |
| **Клиент** |
| Петр |
| Вячеслав |
| Дмитрий |

1. Список товаров с количеством их заказов

Пример результата:

|  |  |
| --- | --- |
| **Товар** | **Количество заказов** |
| Кофе | 1 |
| Кружка | 5 |
| Мяч | 3 |
| Ручка | 1 |
| Телефон | 8 |

**Требования к результату:**

* Реализация на Python 3.7 и выше.
* Допускается использование сторонних библиотек из pip.
* Публичный репозиторий на GitHub с инструкцией по запуску и указанием списка зависимостей.