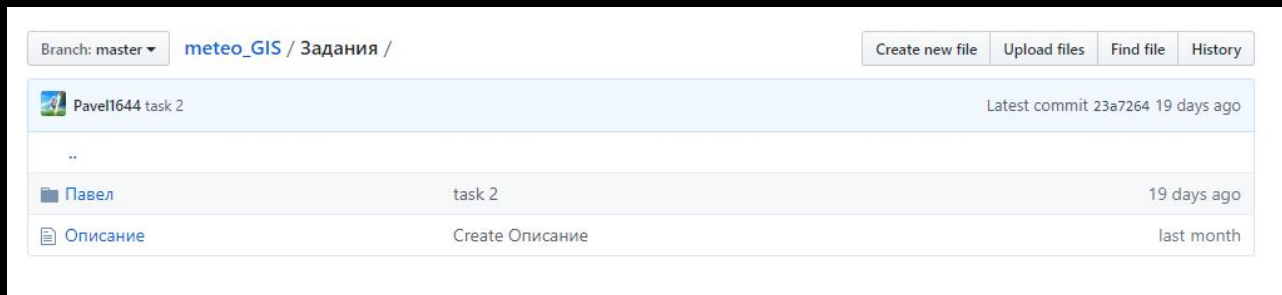


Привет, данная шпаргалка окажется одновременно и вашим вторым обязательным заданием к выполнению. Для того, чтобы освежить в вашей памяти то, какое задание будет считаться выполненным, я напишу необходимые критерии ниже:

- Все файлы, необходимые для выполнения задания, должны быть загружены в папку с вашим именем в репозиторий GitHub



- В зависимости от задания от вас требуется разная итоговая информация. Так например: для **первого** задания вам необходимо предоставить небольшой отчёт, по полученным результатам и файл модели из **QGIS**

Если требования выполнены не будут - задание считается не засчитанным. Без засчитанных заданий (их будет три) перейти к выполнению зачетного задания у вас не получится. А без зачетного задания - нет зачета.

При выполнении этого задания вам необходимо загрузить файл **Jupyter Notebook (.ipynb)** в папку, которую вы создадите в вашей папке в вышеуказанной директории репозитория.

Предварительно откомпилировал ячейки (**Shift + Enter**), чтобы я мог просто посмотреть результат.

Задание заключается в следующем:

Написать две функции, которые принимают на вход строку написанную буквами латинского алфавита, которая вводится пользователем:

```
n = str(input())  
hey guys u menya vso nice
```

И возвращают лист, который представляет из себя набор номером букв в алфавите. Пробелы остаются пробелами. Никаких других знаков деления обрабатывать не нужно.

```
'8525 7212519 21 13514251 221915 14935'
```

Так же напишите функцию, которую из полученного вами списка цифр возвращает первичную фразу.

```
'hey guys u menya vso nice'
```

Фразу придумайте сами. Для того, чтобы вам было понятно, какие методы использовать для выполнения задания, я оставлю чуточку ссылок, которые позволят вам разобраться достаточно, чтобы одолеть данное задание. В случае, если будут возникать конкретные вопросы - пишите мне тут

А теперь ссылки:

- Для ввода строки пользователем используйте метод [input\(\)](#)
- Для того, чтобы получить английский алфавит, воспользуйтесь методом, который приведён в библиотеке [string](#)
- Чтобы получить возможность возвращать индекс буквы, я рекомендую воспользоваться методом [zip\(\)](#). Если возникнут альтернативы - с радостью посмотрю на ваши решения !
- Функцию можете задать как классическим способом, так и с помощью лямбда выражения
- Для того, чтобы перебирать величины внутри листа, в который вы превратите вашу входную строку, воспользуйтесь циклом [for](#) или конструкцией [list comprehension](#)
- Для того, чтобы сопоставить букву с её индексом, советую воспользоваться условием [if](#)
- Чтобы не заблудиться в индексации внутри листа, ознакомьтесь с этим материалом
- Чтобы ваши разделённые символы в листе, которые вы присоедините с помощью метода [.append\(\)](#) стали элегантно строчкой, воспользуйтесь методом [.join\(\)](#)

Успехов !