

## Задача 1 “Файлики”.

Реализуйте простенький помощник.

Пользователь вводит строку, и число.

Программа пишет в текстовый файл эту строку ровно столько раз, сколько пользователь ввел.

Ниже вы видите режимы для функции `open()`, которые помогут вам при решении.

Режим	Описание
'r'	Открытие файла для чтения. Режим используется по умолчанию.
'w'	Открытие файла для записи. Режим создаёт новый файл, если он не существует, или стирает содержимое существующего.
'x'	Открытие файла для записи. Если файл существует, операция заканчивается неудачей (исключением).
'a'	Открытие файла для добавления данных в конец файла без очистки его содержимого. Этот режим создаёт новый файл, если он не существует.
't'	Открытие файла в текстовом формате. Этот режим используется по умолчанию.
'b'	Открытие файла в бинарном формате.
'+'	Открытие файла для обновления (чтения и записи).

## Задача 2 “Адский бот”.

Напишите бота, который может работать с тремя командами: `/start`, `/adduser`, `/deluser`

Он делает очень простую вещь: при старте здоровается с юзером, далее после команды `adduser` добавляет его имя в текстовый файл, а потом удаляет, при команды `deluser`.

Внимание! Обратите внимание на таблицу выше. Нам надо, чтобы данные не затирались, а хранились там вечно. Подсказка -- используйте режим “a”.

## Задача 3 “Райский бот” \*.

Переведите бота выше на три разных языка и добавьте новые команды для переключения языков. Реализуйте это именно через словари, а не через `if-else`.

Бот должен выводить информацию (логи) в командную строку.

## **Задача 4 “Советчик”. (задача с полужирной звездочкой^)**

Все то же самое, бот с хранением пользователей, но при этом он с вами может общаться с заранее заготовленными фразами и отвечает на ваши сообщения ответами из готового словаря. На разных языках.

Любые наработки по поиску коллизий по типу “ппривееет -> привет” (по типу тех, которые мы разбирали на занятии про словари) приветствуются. Любые усложнения и улучшения тоже на ваш вкус.

**Лучшие работы разберем на занятии!!!**