# Библиотеки Python. Часть 5. Документы

# Простая документация в дорогу

Kлассная работа

макс. 2 балл.

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Константин уезжает на каникулы к бабушке в деревню, Интернет там не работает. Однако в свободное время он хотел почитать документацию на некоторые библиотеки Python. Проблема в том, что документация к библиотекам написана в формате **Markdown** (<https://ru.wikipedia.org/wiki/Markdown>) и в обычном офисном пакете не открывается нормально.

Помогите Константину и напишите функцию **markdown\_to\_docx(text)**, которая принимает текст, содержащий на первой строке название документа, а потом – несколько строк с содержимым документа в формате Markdown, создаёт документ и сохраняет его в файле **res.docx**.

В этот раз нам немного повезло, и документация может содержать только следующие элементы:

* заголовки уровней 1-6;
* одноуровневые (то есть не содержащие вложенных) нумерованные и ненумерованные списки;
* абзацы обычного, наклонного, жирного и наклонно-жирного текста. Стиль распространяется на весь абзац. Вложенного оформления не бывает.

На пустые строки Markdown создаются пустые строки в документе.

## Формат ввода

Строка с названием документа, а затем его содержание на нескольких строках в формате **Markdown** в параметре **text**.

## Формат вывода

Сохранённый на диске в директории с программой файл **res.docx** со сконвертированным содержимым.

## Примечания

Данная задача дополнительно проверяется преподавателем.

# Документация в дорогу

Kлассная работа

макс. 3 балл.

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Константин снова уезжает на каникулы и хочет взять с собой документацию. Только в этот раз структура документации значительно сложнее, и поэтому необходимо модифицировать конвертер, который вы написали в задаче «Простая документация в дорогу».

Помогите Константину и напишите новый конвертер, который принимает на вход название документа, а потом несколько строк с содержимым документа в формате **Markdown**. Ваш конвертер должен создавать документ Word и сохранять его в файле <название документа>.docx.

Помимо элементов, с которыми уже умеет работать предыдущая программа, она должна также корректно обрабатывать: обычный, наклонный, жирный и наклонно-жирный текст внутри обычных абзацев, нумерованных и ненумерованных списков. Обратите внимание, что внутри жирного текста может быть кусочек наклонного текста, и тогда он должен быть написан наклонно-жирным стилем (такое верно и для жирного текста внутри наклонного).

На пустые строки Markdown создаются пустые строки в документе только в том случае, если эта строка **не первая** после строки с текстом.

## Формат ввода

Строка с названием документа, а затем его содержимое на нескольких строках в формате **Markdown**.

## Формат вывода

Сохранённый на диске в директории с программой файл <название документа>.docx со сконвертированным содержимым.

## Примечания

Данная задача дополнительно проверяется преподавателем.

**Чек**

Kлассная работа

макс. 3 балл.

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Напишите функцию **export\_check(text)**, которая принимает на вход текст, где каждая строка состоит из:

1. **названия товара (строка)**
2. **цены за единицу товара (вещественное число)**
3. **количества единиц товара (целое число)**

(Значения внутри строки разделены символом табуляции).  
Ваша программа должна обработать полученные данные и создать документ Excel **res.xlsx**, в котором будут следующие столбцы:

**«Товар», «Цена за единицу товара», «Количество товара», «Общая стоимость товара»** (названия столбцов выводить не надо)

В столбце **«Общая стоимость товара»** должна содержаться формула, которая рассчитывает общую стоимость товара как произведение цены и количества товара (с использованием простого перемножения ячеек)

Последней строкой должно идти **«Итого»**, c пустыми значениями в столбцах **«Цена за единицу товара»** и **«Количество товара»**.  
В столбце **«Общая стоимость товара»** должна содержаться формула, которая считает общую сумму покупок. (с использованием функции Excel СУММ)

**Формат ввода**

Текст из нескольких строк в параметре **text**.

**Формат вывода**

Документ **res.xlsx** с преобразованными данными.

**Примечания**

Данная задача дополнительно проверяется преподавателем.

# Чеки

Kлассная работа

макс. 4 балл.

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Модифицируйте функцию из задачи **«Чек»** таким образом, чтобы она могла принимать на вход информацию о нескольких чеках.

Для разделения товаров разных чеков используется строка «---». В документе **res.xlsx** информация о каждом чеке должна располагаться на отдельном листе. При дублировании товара (совпадении названия и цены за единицу) в чеке в **res.xlsx** должна создаваться только одна строка с общим количеством купленного товара.

Товары в чеке должны быть отсортированы по алфавиту.

## Формат ввода

Многострочный текст в параметре **text**.

## Формат вывода

Файл **res.xlsx**, содержащий обработанную информацию.

## Примечания

Данная задача дополнительно проверяется преподавателем.

# Учебная ведомость

Kлассная работа

макс. 4 балл.

|  |  |
| --- | --- |
| Ограничение времени | 1 секунда |
| Ограничение памяти | 64Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Напишите функцию create\_training\_sheet, которая с использованием библиотеки **docxtpl**генерирует учебную ведомость.  
Функция должна принимать на вход:

1. строку **class\_name**- номер класса,
2. строку **subject\_name**- название предмета,
3. строку **tpl\_name**- имя шаблона документа, по умолчанию "tpl.docx"
4. неограниченное количество кортежей строк, которые содержат ФИО ученика в нулевом элементе и оценку за год для этого ученика.

Функция должна генерировать документ по шаблону **tpl\_name**и сохранять его под именем "res.docx".  
В качестве параметра шаблон принимает:

1. строку **class\_name**- номер класса,
2. строку **subject\_name**- название предмета,
3. список словарей **marks**- где каждый словарь содержит ключи **num**, **ﬁo**и **mark**которые содержат номер ученика по алфавиту, ФИО ученика и его оценку за год соответственно.

Ученики должны быть упорядочены по алфавиту.

## Формат ввода

Пример вызова теста:

create\_training\_sheet("3И", "Математика", "tpl.docx",    
                      ("Петров Петр", "5"),    
                      ("Иванов Иван", "5"),    
                      ("Сергеев Сергей", "3"),    
                      ("Никитин Никита", "4"))

Пример шаблона [tpl.docx](https://lyceum.s3.yandex.net/content/problems/tpl.docx)

## Формат вывода

Пример результата [res.docx](https://lyceum.s3.yandex.net/content/problems/res.docx)