# Зачетное задание по курсу PY110

Задание необходимо выполнить в dev ветке!!! И прислать в виде Pull Request вашему преподавателю. Для работы можно использовать [заготовку](https://docs.google.com/document/d/1ZEzcXG25OgBPvnkRljd6Q_RKUUjarG1EnMcFOvNSK1A/edit#heading=h.2ilwvjpy7w8k), выполненную дома.

Чтобы получить зачет необходимо в рамках зачетного занятия ~ 3 часа выполнить “Основное задание”. Если в указанное время не удалось получить зачет, то его можно выполнить после, но уже с дополнительными заданиями на усмотрение преподавателя.

## Основное задание

Зачетное задание должно представлять из себя модуль main.py, который генерирует случайные книги.

1. Модуль должен содержать в себе функцию-генератор, которая возвращает словари следующей структуры:

| {  "model": "shop\_final.book",  "pk": 1,  "fields": {  "title": "test\_book",  "year": 2020,  "pages": 123,  "isbn13": "978-1-60487-647-5",  "rating": 5,  "price": 123456.0,  "author": [  "test\_author\_1",  "test\_author\_2"  ]  } } |
| --- |
|  |

Где поле

1. **“model”** содержится в конфигурационном файле conf.py. Его значение хранится в переменной MODEL, переменная попадает в main.py путем импорта.
2. **“pk”** является автоинкрементом, то есть счётчиком, который увеличивается на единицу при генерации нового объекта. По умолчанию значение поля **“pk”** = 1. Также должна быть реализована возможность устанавливать начальное значение через **аргумент функции-генератора (!!!)** на момент инициализации.

Fields

* 1. **“Title”** Содержит в себе название книги. Список возможных названий хранится в файле books.txt (достаточно 5 различных названий книг, но на русском языке). Каждая книга указана на отдельной строке. При генерации случайных словарей книг названия книг могут повторяться.
  2. **“year”** является натуральным числом и генерируется случайным образом
  3. **“pages”** является натуральным числом и генерируется случайным образом
  4. **“isbn13”** международный стандартный книжный номер, генерируется случайным образом с помощью модуля Faker <https://faker.readthedocs.io/en/master/providers/faker.providers.isbn.html#faker.providers.isbn.Provider.isbn13>
  5. **“rating”** - число с плавающей запятой в диапазоне от 0 до 5 обе границы включительно. Генерируется случайным образом
  6. **“price”** - число с плавающей запятой, генерируется случайным образом
  7. **“author”** - список авторов. Содержит от 1 до 3 авторов. Имя и фамилия автора выбираются случайным образом с помощью модуля Faker

1. Получение значений для полей из **fields** (см. предыдущий пункт) должно быть реализовано в виде отдельных функций. Одно поле - одна функции.
2. В блоке if \_\_name\_\_ == "\_\_main\_\_": должна быть только одна функция main(), которая запускает функцию генератор, формирует список из 100 книг (список словарей) и записывает его в json файл. Json файл должен быть читабельным (отступы, кодировки)
3. Модуль должен быть написан с учетом всех требований PEP8, а также в коде должна документация по модулю. Каждая функция должна иметь документацию, описание параметров. Все функции и их параметры должны иметь аннотацию типов.

## Дополнительные задания

1. Сделать функцию, которая возвращает название случайной книги, эффективной. Эффективной считается функция, которая не считывает сразу весь файл и выбирает случайную книгу, а считывает только одну случайную строку с названием книги.
2. Написать декоратор, который будет проводить валидацию названия книги, а именно проверять максимальную длину книги, которая записана внутри декоратора. В случае, если длина названия книги больше заданного значения, то генерируется ошибка ValueError с соответствующим сообщением.
3. Написать модуль validate\_isbn\_13.py, в котором будет содержаться регулярное выражение, проверяющее правильность структуры значения ISBN13, генерируемого модулем Faker. Проверить на миллионе случайных значений. Структура ISBN13:

Каждый ISBN состоит из 5 элементов, каждый раздел разделен пробелами или дефисами. Три из пяти элементов могут иметь разную длину:

* Элемент префикса - в настоящее время это может быть только 978 или 979. Он всегда состоит из трех цифр.
* Элемент группы регистрации - определяет конкретную страну, географический регион или языковую зону, участвующую в системе ISBN. Этот элемент может иметь длину от 1 до 5 цифр.
* Элемент регистранта - идентифицирует конкретного издателя или выходных данных. Длина может составлять от 1 до 7 цифр.
* Элемент публикации - определяет конкретное издание и формат определенного заголовка. Длина может составлять от 1 до 6 цифр.
* Контрольная цифра - это всегда последняя отдельная цифра (арабская от 0 до 9 или римская X)

1. Переписать декоратор для валидации максимальной длины названия книги под фабрику декораторов, чтобы принимать максимальную длину как параметр.

## Вспомогательный материал:

1. Для генерации случайных значений, можно использовать модуль [random](https://docs.python.org/3/library/random.html) из стандартной библиотеки. См. документацию и примеры
2. Faker:

<https://faker.readthedocs.io/en/master/#basic-usage> Установка

<https://faker.readthedocs.io/en/master/fakerclass.html#upgrade-guide> Инициализация

<https://faker.readthedocs.io/en/master/providers/faker.providers.isbn.html#faker.providers.isbn.Provider.isbn13> Генерация ISBN13

1. Структура ISBN13 [тут](https://www.isbn-international.org/content/what-isbn) и [тут](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D0%B6%D0%B4%D1%83%D0%BD%D0%B0%D1%80%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BD%D0%B8%D0%B6%D0%BD%D1%8B%D0%B9_%D0%BD%D0%BE%D0%BC%D0%B5%D1%80)