

**Задание 1.** X красных машин и Y белых стоят в одном ряду. Напишите программу, которая выдаст, как нужно расположить красные и белые машины, чтобы рядом с каждой красной стояла хотя бы одна белая, а рядом с каждой белой — хотя бы одна красная.

На вход подаются два числа - кол-во красных машин X и кол-во белых машин Y. В ответе выведите какую-нибудь строку, в которой будет равно X символов "R" (красные машины) и Y символов "W" (белые машины), удовлетворяющую условию задачи. Пробелы между символами выводить не нужно. Если расставить машины согласно условию задачи невозможно, выведите строку "Нет решения".

**Задание 2.** Задача: спроектируйте API для онлайн-магазина породистых котят.

Используемые технологии: Flask, sqlalchemy + postgres, docker

Создайте простой портал для электронной библиотеки. Админка должна содержать две страницы:

1. список всех котиков с пагинацией (по 5 котиков) и поиском. При нажатии на котика попадаем на вторую страницу

2. страница отдельного котика (на ней должен отображаться порода, картинка, имя, краткое описание и возраст в месяцах)

Должен поддерживаться полнотекстовый поиск по всем полям информации о котиках (порода, имя, возраст, описание). Должна быть возможность выбрать сортировку отдаваемого результата: по породе, по возрасту, по релевантности (сортировка по умолчанию)

Решение оформите в виде проекта на Flask в обвязке Docker контейнера. Запуск решения должен происходить по команде "docker-compose up". В магазине должно

быть как минимум 20 котиков для проверки реализации.

Визуально проект не должен выглядеть отталкивающе. Мы любим котиков :-)

**Задание 3.** Дан ресурс в интернете, его адрес вводится с клавиатуры. Напишите функцию, которая будет одновременно отправлять 1 000 GET-запросов к нему. С помощью библиотеки tqdm отобразите progress-bar, отображающий количество уже отправленных запросов.

**Задание 4.** (Проверка Soft скилов)

Два студента выполнили домашнее задание и прислали его на проверку. Ниже есть текст заданий и результат работы студентов. Определите, стоит принять выполненную работу или вернуть ее на доработку. Составьте ответ студенту по каждой задаче в отдельности.

Студент 1.

Задание: Ежемесячная стипендия студента составляет educational\_grant руб., а расходы на проживание превышают стипендию и составляют expenses руб. в месяц. Рост цен ежемесячно увеличивает расходы на 3%, кроме первого месяца.

Составьте программу расчета суммы денег, которую необходимо единовременно попросить у родителей, чтобы можно было прожить учебный год (10 месяцев), используя только эти деньги и стипендию.

[Решение студента](#)

Комментарий: задание из модуля “Цикл while”

(Представьте, что студент прошел следующие темы: [Список](#))

Ответьте студенту, который неправильно решил задание.

Студент 2.

Задание: По номеру месяца вывести кол-во дней в нем (без указания названия месяца, в феврале 28 дней). Результат проверки вывести на консоль. Если номер месяца некорректен - сообщить об этом

[Решение студента](#)

Комментарий: задание из модуля “Начало работы со списками”

Ответьте студенту, который правильно решил задание.