1. Вам необходимо выстроить и написать архитектуру для банковских продуктов. Банковскими продуктами являются – кредитная карта, дебетовая карта, валютная дебетовая карта и вклад.

- Карты и вклад имеют параметры: валюта, баланс и название

- Карты содержат методы: пополнение, списание, запрос баланса

- Кредитная карта имеет дополнительный метод запрос задолженности и дополнительный параметр процентная ставка

- Вклад содержит методы: пополнение, запрос баланса, закрытие

Опишите вашу структуру/архитектуру в классах. Учтите, что продуктов в будущем станет больше - то есть предусмотрите возможность изменения вашей архитектуры без различных проблем и лишних правок.

Напишите unit тесты для проверки реализованной архитектуры.

2.

Зарегистрироваться на сервисе передачи данных о погоде - <https://www.weatherapi.com/>. Для этого нужно указать адрес электронной почты и получить токен (API Key), который используется для запросов. Бесплатный ключ действует две недели.

Документация по API - <https://www.weatherapi.com/docs/>.

Используя любой BDD фреймворк реализовать:

1. **Позитивный тест:** 
   * Запросить текущую погоду (метод /current.json) минимум по четырем городам на свой выбор.
   * Распарсить результат, сравнить с ожидаемыми значениями из тестового набора. Ожидаемые тестовые данные можно определить или задать для каждого города корректные, либо можно задать\сгенерировать случайным образом с соблюдением формата.
   * Вывести расхождения по результату сравнения по каждому значению в лог.
2. **Негативный тест:**
   * Получить 4 варианта ошибок из списка API Errors (на выбор), сравнить с ожидаемым результатом.

Результат выполнения тестов должен быть в отчете Allure.