

# Задание на комиссию

## Базовая часть

### 1.1. Создание объектов

В приложении должна быть кнопка с текстом "Создать пользователей". По нажатию на эту кнопку должно создаваться **5 объектов класса User** со следующими полями, заполненными с помощью библиотеки **Data Faker**:

- id (UUID)
- name (String)
- email (String)
- userType (Enum: REGULAR, PREMIUM) // Тип должен определяться случайным образом

Процесс создания объектов должен быть инкапсулирован с использованием **порождающего паттерна «Фабричный метод» (Factory Method)**. Конкретный тип создаваемого объекта (например, если бы была иерархия классов) определялся бы на основе переданных параметров.

### 1.2. Подписка на событие и вывод информации

В приложении должна быть вторая кнопка с текстом "Показать PREMIUM".

- В момент создания каждый объект User должен **подписаться на событие нажатия этой кнопки**, но только если его userType == UserType.PREMIUM.
- При нажатии кнопки "Показать PREMIUM" в графический интерфейс (например, в многострочное текстовое поле JTextArea) должен выводиться список всех PREMIUM-пользователей в формате: {name} | {email}.
- После однократного срабатывания события **подписка данного объекта на это событие должна быть отменена**. Повторное нажатие кнопки не должно выводить информацию о пользователях, созданных в предыдущую сессию.

### 1.3. Реализация событий

Механизм подписки и оповещения (управление подписчиками и рассылка уведомлений) должен быть реализован с использованием паттерна **«Издатель-Подписчик» (Publisher-**

**Subscriber)**. Класс кнопки "Показать PREMIUM" должен выступать в роли Издателя (Publisher).

### 1.4. Применение лямбда-выражений

**Подписка на событие** (регистрация обработчика) должна быть реализована с использованием **лямбда-выражения**. Лямбда-выражение должно определять действие, которое необходимо выполнить объекту User при наступлении события (нажатии кнопки).