

## Домашнее задание

### Упражнение

1. Вам дан исходный код класса `PaintFrame`, однако там присутствует баг - когда вы меняете цвет пера, все линии на панели меняют цвет. Вам нужно придумать способ, как сохранять информацию о цвете отдельной линии.
2. Начинаем в домашних заданиях новый раздел - Алгоритмы! При-сылать все алгоритмы в одном файле `.java`. Очень желательно на алгоритмы написать `unit`-тесты.
  - найти максимальный элемент массива (в массиве только положительные целые числа)
  - написать функцию разворота строки (`abcd -> dcba`)
  - на вход функции подается строка, посчитать частоту вхождения каждой буквы в строке и вернуть `Map<String, Integer>` - список букв и их частот

### Финансовый менеджер

Мы уже написали логику работы с данными (`DataSource`, `User`, `Account`, `Record`) и сделали авторизацию. Теперь задача - реализовать графический интерфейс. Минимальный интерфейс должен содержать окна:

- окно авторизации (позволяет зарегистрироваться, есть окно ввода для логина, пароля, кнопка `Login/Create User`). Если что-то пошло не так, вывести в окно строку с причиной ошибки. Если пароль верный, то скрыть окно авторизации и создать основное окно приложения.
- основное окно приложения содержит в себе следующие элементы:
  - имя пользователя и текущий остаток на счете
  - по умолчанию, работаем с одним счетом. (\*) выпадающий список счетов
  - пролистывающийся список транзакций

- элемент списка транзакций содержит текстовое описание транзакции
  - кнопка "Добавить запись/транзакцию". По щелчку появляется новое модальное окно (то есть родительское окно становится недоступным), где можно добавить транзакцию. Реализуется с помощью диалоговых окон JDialog
  - (\*) строка меню с пунктами "сменить пользователя" "создать аккаунт" "выход"
- окно создания транзакции. Дает возможность заполнить все поля транзакции. Для выбора пополнение/снятие использовать JRadioButton. Для выбора категории - выпадающий список.
  - (\*) элемент списка транзакций должен иметь свой дизайн. Изучите класс ListCellRenderer - он умеет в качестве элемента списка возвращать объект класса Component, который также может содержать элементы swing - кнопки, надписи, и т.д. Идеи дизайна - добавить иконку снятие/пополнение, раскрашивать, в зависимости от категории.
  - (\*) по клику на элемент транзакции открыть окно редактирования, добавить возможность удалить транзакцию