|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Отчет по выполнению практической работы 4

**Тема. СОЗДАНИЕ GUI. СОБЫТИЙНОЕ ПРОГРАММИРОВАНИЕ В JAVA.**

Дисциплина Программирование на языке Джава

Выполнил

|  |  |
| --- | --- |
| студент | Болотов Михаил |
|  | Фамилия И.О. |
| группа | ИКБО-06-19 |
|  | Номер группы |

Москва 2020

**Содержание**

[​ Теория 2](#__RefHeading___Toc4128_811265792)

[​ Задание 2](#__RefHeading___Toc4136_811265792)

[​ Код 3](#__RefHeading___Toc35859_811265792)

[​ Counter.java 3](#__RefHeading___Toc68417_811265792)

[​ Скриншот 5](#__RefHeading___Toc3748_811265792)

[​ Заключение 5](#__RefHeading___Toc3750_811265792)

[​ Библиографический список 5](#__RefHeading___Toc3752_811265792)

# Теория

Данная практическая работа посвящена закреплению практических навыков по созданию приложений на Java c использованием следующих элементов GUI:

* Текстовые поля и области ввода текста;
* Менеджеры компоновки компонентов;
* Слушатель мыши;
* Создание меню.

Text Fields - текстовое поле или поля для ввода текста (можно ввести только одну строку). Примерами текстовых полей являются поля для ввода логина и пароля, например, используемые, при входе в электронную почту.

Пример создания объекта класса JTextField:

JTextField jta = new JTextField (10);

В параметрах конструктора задано число 10, это количество символов, которые могут быть видны в текстовом поле. Текст веденный в поле JText может быть возвращен с помощью метода getText(). Также в поле можно записать новое значение с помощью метода setText(String s).

Как и у других компонентов, мы можем изменять цвет и шрифт текста в текстовом поле.

Ответственность за выполнение проверки на наличие ошибок в коде лежит полностью на программисте, например, чтобы проверить произойдет ли ошибка, когда в качестве входных данных в JTextField ожидается ввод числа. Компилятор не будет ловить такого рода ошибку, поэтому ее необходимо обрабатывать пользовательским кодом.

# Задание

Напишите интерактивную программу с использованием GUI имитирует таблицу результатов матчей между командами Милан и Мадрид. Создайте JFrame приложение у которого есть следующие компоненты GUI:

* одна кнопка JButton labeled “AC Milan”
* другая JButton подписана “Real Madrid”
* надпись JLabel содержит текст “Result: 0 X 0”
* надпись JLabel содержит текст “Last Scorer: N/A”
* надпись Label содержит текст “Winner: DRAW”;

Всякий раз, когда пользователь нажимает на кнопку AC Milan, результат будет увеличиваться для Милана, сначала 1 X 0, затем 2 X 0 и так далее. Last Scorer означает последнюю забившую команду. В этом случае: AC Milan. Если пользователь нажимает кнопку для команды Мадрид, то счет приписывается ей. Победителем становится команда, которая имеет больше кликов кнопку на соответствующую, чем другая.

# Код

### Counter.java

package dev.ky3he4ik.pr.pr4;  
  
import javax.swing.\*;  
import java.awt.\*;  
  
// 4.1  
public class Counter extends JFrame {  
 private final String MILAN\_STR = "AC Milan";  
 private final String MADRID\_STR = "Real Madrid";  
 private final String DRAW\_STR = "DRAW";  
 private final String RESULT\_STR\_BASE = "Result: ";  
 private final String LAST\_SCORER\_STR\_BASE = "Last Scorer: ";  
 private final String WINNER\_STR\_BASE = "Winner: ";  
  
 private JLabel resultLbl;  
 private JLabel lastScorerLbl;  
 private JLabel winnerLbl;  
 private int milanScore = 0;  
 private int madridScore = 0;  
  
 public Counter() {  
 setLayout(new BorderLayout());  
  
 JButton milanBtn = new JButton(MILAN\_STR);  
  
 milanBtn.addActionListener(actionEvent -> {  
 milanScore++;  
 replaceStrings(MILAN\_STR);  
 });  
 JButton madridBtn = new JButton(MADRID\_STR);  
 madridBtn.addActionListener(actionEvent -> {  
 madridScore++;  
 replaceStrings(MADRID\_STR);  
 });  
 resultLbl = new JLabel(RESULT\_STR\_BASE + milanScore + " X " + madridScore, SwingConstants.CENTER);  
 lastScorerLbl = new JLabel(LAST\_SCORER\_STR\_BASE + "N/A", SwingConstants.CENTER);  
 winnerLbl = new JLabel(WINNER\_STR\_BASE + DRAW\_STR, SwingConstants.CENTER);  
  
 add(milanBtn, BorderLayout.WEST);  
 add(madridBtn, BorderLayout.EAST);  
 add(resultLbl, BorderLayout.CENTER);  
 add(lastScorerLbl, BorderLayout.NORTH);  
 add(winnerLbl, BorderLayout.SOUTH);  
  
 setSize(640, 400);  
 setVisible(true);  
 }  
  
 private void replaceStrings(String lastScorerName) {  
 resultLbl.setText(RESULT\_STR\_BASE + milanScore + " X " + madridScore);  
 lastScorerLbl.setText(LAST\_SCORER\_STR\_BASE + lastScorerName);  
 if (madridScore == milanScore)  
 winnerLbl.setText(WINNER\_STR\_BASE + DRAW\_STR);  
 else  
 winnerLbl.setText(WINNER\_STR\_BASE + ((madridScore > milanScore) ? MADRID\_STR : MILAN\_STR));  
 }  
  
 public static void main(String[] args) {  
 new Counter();  
 }  
}

# Скриншот



# Заключение

В данной практической работе я освоил на практике событийное программирование на языке Java.

# Библиографический список

1. Зорина Н.В. Курс лекций по Объектно-ориентированному программированию на Java, МИРЭА, Москва, 2016
2. Программирование на языке Java: работа со строками и массивами. Методические указания. [Электронный ресурс] : Учебно-методические пособия — Электрон. дан. — СПб. : ПГУПС, 2015. — 24 с.
3. Кожомбердиева, Г.И. Программирование на языке Java: создание графического интерфейса пользователя: учеб. пособие. [Электронный ресурс] : Учебные пособия / Г.И. Кожомбердиева, М.И. Гарина. — Электрон. дан. — СПб.: ПГУПС, 2012. — 67 с.
4. Вишневская, Т.И. Технология программирования. Часть 1. [Электронный ресурс] / Т.И. Вишневская, Т.Н. Романова. — Электрон. дан. — М. : МГТУ им. Н.Э. Баумана, 2007. — 59 с.