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

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**Дисциплина «Программирование на языке Джава»**

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №4**

Выполнил студент группы ИНБО-02-20 Резаев А.А

Принял Степанов П.В.

Практические работы выполнены «\_\_\_»\_\_\_\_\_\_\_2021г.

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_»\_\_\_\_\_\_\_2021г.

Отметка о выполнении

**Москва – 2021 г.**

## **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.

## **Ход Работы**

В ходе выполнения этой практической работы было получен исходный код.

Листинг программы.

|  |
| --- |
| package PR\_3;  import java.awt.\*;  import java.awt.event.\*;  import javax.swing.\*;  public class Match extends JFrame{  public int x;  public int y;  JButton button = new JButton("AC Milan");  JLabel schet = new JLabel("Result: 0 X 0");  JButton button1 = new JButton("Real Madrid");  JLabel last = new JLabel("Last Score: N/A");  Label win = new Label("Winner: DRAW");  public void PechatPobeda(){  if (x>y){  win.setText("Winner: AC Milan");}  else if (y>x) {  win.setText("Winner: Real Madrid");  }  else {  win.setText("Winner: DRAW");  }  }  public Match(){  super("Football");  setLayout(new FlowLayout());  setSize(350,200);  add(button);  add(button1);  add(win);  add(schet);  add(last);  button.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  ++x;  schet.setText("Result: " + x + " X " + y);  PechatPobeda();  last.setText("Last score: AC Milan");  }  });  button1.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  ++y;  schet.setText("Result: " + x + " X " + y);  PechatPobeda();  last.setText("Last score: Real Madrid");  }  });  }  public static void main(String[]args)  {  new Match().setVisible(true);  }  } |

## **Вывод**

В ходе выполнения этой практической работы я изучил основы событийного программирования и создал мини-программу.