

1. Berdasarkan UI pada soal praktikum-7 nomor 3, 4 dan 5 buatlah flow aplikasi sebagai berikut:

- a. Source Code - Login Page

```

1 package com.ibik.pbo.praktikum;
2
3 import java.awt.EventQueue;
4
5 public class Login extends JFrame {
6
7     private JPanel contentPane;
8     private JTextField field_Email;
9     private JTextField field_Password;
10    private JButton button_Login;
11    private JButton btnRegister;
12
13    /**
14     * Launch the application.
15     */
16    public static void main(String[] args) {
17        EventQueue.invokeLater(new Runnable() {
18            public void run() {
19                try {
20                    Login frame = new Login();
21                    frame.setVisible(true);
22                } catch (Exception e) {
23                    e.printStackTrace();
24                }
25            }
26        });
27    }
28
29    /**
30     * Create the frame.
31     */
32    public Login() {
33        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
34        setBounds(100, 100, 250, 182);
35        setLocationRelativeTo(null);
36        contentPane = new JPanel();
37        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
38
39        setContentPane(contentPane);
40
41        contentPane.setLayout(null);
42
43        JLabel label_Email = new JLabel("Email");
44        label_Email.setBounds(10, 14, 46, 14);
45        contentPane.add(label_Email);
46
47        field_Email = new JTextField();
48        field_Email.setColumns(10);
49        field_Email.setBounds(99, 11, 133, 20);
50        contentPane.add(field_Email);
51
52        JLabel label_Password = new JLabel("Password");
53        label_Password.setBounds(10, 42, 79, 14);
54        contentPane.add(label_Password);
55
56        field_Password = new JTextField();
57        field_Password.setColumns(10);
58        field_Password.setBounds(99, 39, 133, 20);
59        contentPane.add(field_Password);
60
61        JCheckBox cb_RemPassword = new JCheckBox("Remember Password");
62        cb_RemPassword.setFont(new Font("Tahoma", Font.PLAIN, 11));
63        cb_RemPassword.setBounds(99, 66, 125, 23);
64        contentPane.add(cb_RemPassword);
65
66        button_Login = new JButton("LOGIN");
67        button_Login.addMouseListener(new MouseAdapter() {
68            @Override
69            public void mouseClicked(MouseEvent e) {
70                if (field_Email.getText().equals("") || field_Password.getText().equals("")) {
71                    JOptionPane.showMessageDialog(null, "Silakan mengisi data dengan benar", "Error", JOptionPane.ERROR_MESSAGE);
72                } else if (field_Email.getText().equals("test@gmail.com") && field_Password.getText().equals("123")) {
73                    JOptionPane.showMessageDialog(null, "Selamat Datang Ferdy");
74                    new MainMenu().setVisible(true);
75                } else {
76                    JOptionPane.showMessageDialog(null, "Data yang anda masukkan salah", "Error", JOptionPane.ERROR_MESSAGE);
77                }
78            }
79        });
80        button_Login.setBounds(125, 96, 107, 30);
81        contentPane.add(button_Login);
82    }
83
84 }
85
86
87
88
89
90
91
92
93
94
95
96

```

```

97
98     btnRegister = new JButton("REGISTER");
99     btnRegister.addMouseListener(new MouseAdapter() {
100         @Override
101         public void mouseClicked(MouseEvent e) {
102             new Register().setVisible(true);
103         }
104     });
105     btnRegister.setBounds(10, 96, 107, 30);
106     contentPane.add(btnRegister);
107 }
108 }

```

## b. Source Code - Main Menu

```

1 package com.ibik.pbo.praktikum;
2
3 import java.awt.EventQueue;
4
5
6 public class MainMenu extends JFrame {
7     private JTable table;
8     private JTable table_1;
9     private JTable table_2;
10    private JTextField textField;
11    private JTextField textField_1;
12
13    /**
14     * Launch the application.
15     */
16    public static void main(String[] args) {
17        EventQueue.invokeLater(new Runnable() {
18            public void run() {
19                try {
20                    MainMenu frame = new MainMenu();
21                    frame.setVisible(true);
22                } catch (Exception e) {
23                    e.printStackTrace();
24                }
25            }
26        });
27    }
28
29    /**
30     * Create the frame.
31     */
32    public MainMenu() {
33        setTitle("Latihan 04");
34        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
35        setBounds(100, 100, 899, 614);
36        setLocationRelativeTo(null);
37
38        JMenuBar menuBar = new JMenuBar();
39        setJMenuBar(menuBar);
40
41        JMenu menu_File = new JMenu("File");
42        menuBar.add(menu_File);

```

```

63
64 JMenuItem new_MenuItem = new JMenuItem("New");
65 menu_File.add(new_MenuItem);
66
67 JMenuItem save_MenuItem = new JMenuItem("Save");
68 menu_File.add(save_MenuItem);
69
70 JMenuItem exit_MenuItem = new JMenuItem("Exit");
71 menu_File.add(exit_MenuItem);
72
73 JMenu menu_Edit = new JMenu("Edit");
74 menuBar.add(menu_Edit);
75
76 JMenu menu_Help = new JMenu("Help");
77 menuBar.add(menu_Help);
78 getContentPane().setLayout(null);
79
80 JScrollPane scrollPane = new JScrollPane();
81 scrollPane.setBounds(348, 88, 535, 454);
82 getContentPane().add(scrollPane);
83
84 table_2 = new JTable();
85 table_2.setShowVerticalLines(false);
86 table_2.setShowHorizontalLines(false);
87 table_2.setShowGrid(false);
88 table_2.setModel(new DefaultTableModel(
89     new Object[][] {
90         {"212310018", "FATHURAHMAN AL FARIDZI", "B"},
91         {"212310019", "MUHAMMAD RAFI ZUHAIR ARTA", "C"},
92         {"212310020", "MUHAMAD SUBHAN RIZKI AFIA", "A"},
93     },
94     new String[] {
95         "NPM", "Nama", "Nilai"
96     }
97 ));
98 table_2.getColumnModel().getColumn(0).setMaxWidth(150);
99 scrollPane.setViewportView(table_2);
100
101 JLabel lblNewLabel = new JLabel("FORM PENILAIAN MATA KULIAH PBO ");
102 lblNewLabel.setHorizontalAlignment(SwingConstants.CENTER);
103 lblNewLabel.setFont(new Font("Tahoma", Font.PLAIN, 30));
104 lblNewLabel.setBounds(164, 11, 553, 58);
105 getContentPane().add(lblNewLabel);
106
107 JLabel lblNewLabel_1_1 = new JLabel("NPM");
108 lblNewLabel_1_1.setFont(new Font("Tahoma", Font.PLAIN, 15));
109 lblNewLabel_1_1.setBounds(30, 140, 117, 14);
110 getContentPane().add(lblNewLabel_1_1);
111
112 JTextField textField = new JTextField();
113 textField.setBounds(40, 166, 265, 39);
114 getContentPane().add(textField);
115 textField.setColumns(10);
116
117 JTextField_1 = new JTextField();
118 textField_1.setColumns(10);
119 textField_1.setBounds(40, 267, 265, 39);
120 getContentPane().add(textField_1);
121
122 JLabel lblNewLabel_1_1_1 = new JLabel("Nama");
123 lblNewLabel_1_1_1.setFont(new Font("Tahoma", Font.PLAIN, 15));
124 lblNewLabel_1_1_1.setBounds(30, 241, 117, 14);
125 getContentPane().add(lblNewLabel_1_1_1);
126
127 JLabel lblNewLabel_1_1_1_1 = new JLabel("Nilai");
128 lblNewLabel_1_1_1_1.setFont(new Font("Tahoma", Font.PLAIN, 15));
129 lblNewLabel_1_1_1_1.setBounds(30, 343, 117, 14);
130 getContentPane().add(lblNewLabel_1_1_1_1);
131
132 JRadioButton rdbtnNewRadioButton = new JRadioButton("A");
133 rdbtnNewRadioButton.setBounds(38, 377, 33, 23);
134 getContentPane().add(rdbtnNewRadioButton);
135
136 JRadioButton rdbtnB = new JRadioButton("B");
137 rdbtnB.setBounds(84, 377, 33, 23);
138 getContentPane().add(rdbtnB);
139
140 JRadioButton rdbtnC = new JRadioButton("C");
141 rdbtnC.setBounds(130, 377, 33, 23);
142 getContentPane().add(rdbtnC);
143
144 JRadioButton rdbtnD = new JRadioButton("D");
145 rdbtnD.setBounds(175, 377, 33, 23);

```

```

146         getContentPane().add(rdbtnD);
147
148         JRadioButton rdbtnE = new JRadioButton("E");
149         rdbtnE.setBounds(224, 377, 33, 23);
150         getContentPane().add(rdbtnE);
151
152         JRadioButton rdbtnF = new JRadioButton("F");
153         rdbtnF.setBounds(269, 377, 33, 23);
154         getContentPane().add(rdbtnF);
155
156         JButton btnNewButton = new JButton("Save");
157         btnNewButton.setBounds(34, 427, 89, 39);
158         getContentPane().add(btnNewButton);
159
160         JButton btnDelete = new JButton("Delete");
161         btnDelete.setBounds(133, 427, 89, 39);
162         getContentPane().add(btnDelete);
163
164         JButton btnClear = new JButton("Clear");
165         btnClear.setBounds(232, 427, 89, 39);
166         getContentPane().add(btnClear);
167
168         JPanel panel = new JPanel();
169         panel.setBorder(new TitledBorder(new EtchedBorder(EtchedBorder.LOWERED, new Color(255, 255, 255), new Color(160, 160, 160)
170         panel.setBounds(10, 81, 328, 461);
171         getContentPane().add(panel);
172
173         JMenu fileMenu = new JMenu("File");
174         JMenu editMenu = new JMenu("Edit");
175         JMenu helpMenu = new JMenu("Help");
176     }
177 }

```

### c. Source Code - Register

```

1 package com.ibik.pbo.praktikum;
2
3 import java.awt.EventQueue;
4
19 public class Register extends JFrame {
20
21     private JPanel contentPane;
22     private JTextField field_Email;
23     private JTextField field_Fullname;
24     private JTextField field_Phone;
25     private JLabel label_Gender;
26
27
28 /**
29  * Launch the application.
30  */
31 public static void main(String[] args) {
32     EventQueue.invokeLater(new Runnable() {
33         public void run() {
34             try {
35                 Register frame = new Register();
36                 frame.setVisible(true);
37             } catch (Exception e) {
38                 e.printStackTrace();
39             }
40         }
41     });
42 }
43
44 /**
45  * Create the frame.
46  */
47 public Register() {
48     setTitle("Register");
49     setResizable(false);
50     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
51     setBounds(100, 100, 256, 241);
52     setLocationRelativeTo(null);
53     contentPane = new JPanel();
54     contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
55
56     setContentPane(contentPane);

```

```

57     contentPane.setLayout(null);
58
59     final JLabel label_Fullname = new JLabel("Fullname");
60     label_Fullname.setBounds(26, 11, 46, 14);
61     contentPane.add(label_Fullname);
62
63     field_Email = new JTextField();
64     field_Email.setBounds(82, 39, 133, 20);
65     contentPane.add(field_Email);
66     field_Email.setColumns(10);
67
68     final JLabel label_Email = new JLabel("Email");
69     label_Email.setBounds(26, 42, 46, 14);
70     contentPane.add(label_Email);
71
72     field_Fullname = new JTextField();
73     field_Fullname.setColumns(10);
74     field_Fullname.setBounds(82, 8, 133, 20);
75     contentPane.add(field_Fullname);
76
77     final JLabel label_Phone = new JLabel("Phone");
78     label_Phone.setBounds(26, 73, 46, 14);
79     contentPane.add(label_Phone);
80
81     field_Phone = new JTextField();
82     field_Phone.setColumns(10);
83     field_Phone.setBounds(82, 70, 133, 20);
84     contentPane.add(field_Phone);
85
86     label_Gender = new JLabel("Gender");
87     label_Gender.setBounds(26, 101, 46, 14);
88     contentPane.add(label_Gender);
89
90     final JRadioButton rb_Female = new JRadioButton("Female");
91     rb_Female.setBounds(82, 97, 59, 23);
92     contentPane.add(rb_Female);
93
94     final JRadioButton rb_Male = new JRadioButton("Male");
95     rb_Male.setBounds(156, 97, 59, 23);
96     contentPane.add(rb_Male);
97
98     final JLabel label_Citizenship = new JLabel("Citizenship");
99     label_Citizenship.setBounds(21, 129, 51, 14);
100    contentPane.add(label_Citizenship);
101
102    final JComboBox cb_Citizenship = new JComboBox();
103    cb_Citizenship.setModel(new DefaultComboBoxModel(new String[] {"Indonesia", "Malaysia", "Singapore", "Laos", "Cambodia"}));
104    cb_Citizenship.setToolTipText("");
105    cb_Citizenship.setBounds(82, 127, 133, 22);
106    contentPane.add(cb_Citizenship);
107
108    JButton button_Submit = new JButton("SUBMIT");
109    button_Submit.addMouseListener(new MouseAdapter() {
110        @Override
111        public void mouseClicked(MouseEvent e) {
112            if (field_Fullname.getText().equals("") || field_Email.getText().equals("") || field_Phone.getText().equals("") ||
113                JOptionPane.showMessageDialog(null, "Data tidak boleh kosong. Silakan periksa kembali\r\n"
114                    + "isian anda", "Error", JOptionPane.ERROR_MESSAGE);
115            }else {
116                String gender;
117                if (rb_Female.isSelected() == true) {
118                    gender = "Female";
119                }else {
120                    gender = "Male";
121                }
122                JOptionPane.showMessageDialog(null, label_Fullname.getText() + " : " + field_Fullname.getText() + "\n" + label
123                    new Login().setVisible(true);
124            }
125        }
126    });
127    button_Submit.addActionListener(new ActionListener() {
128        public void actionPerformed(ActionEvent e) {
129        }
130    });
131    button_Submit.setBounds(79, 160, 136, 30);
132    contentPane.add(button_Submit);
133 }
134 }
135

```

#### d. Output Flow

- Masuk menu register

The image shows two windows side-by-side. The 'Login Page' window on the left has fields for 'Email' and 'Password', a 'Remember Password' checkbox, and 'REGISTER' and 'LOGIN' buttons. The 'Register' window on the right has fields for 'Fullna...', 'Email', 'Phone', 'Gender' (with radio buttons for 'Fem...' and 'Male'), and a 'Citizen...' dropdown menu set to 'Indonesia'. It has a 'SUBMIT' button.

- Register data tidak lengkap

The 'Register' window shows 'Fullna...' as 'Ferdy Apr', 'Email' as 'test@gmail.com', and 'Phone' as empty. 'Gender' is set to 'Male' and 'Citizen...' is 'Indonesia'. The 'SUBMIT' button is visible. To the right, an 'Error' dialog box with a red 'X' icon displays the message: 'Data tidak boleh kosong. Silakan periksa kembali isian anda' (Data cannot be empty. Please check your input again) with an 'OK' button.

- Register data lengkap

The 'Register' window now has 'Phone' filled with '123'. All other fields remain the same. To the right, a 'Message' dialog box with an information icon displays the registered details: 'Fullname : Ferdy Apr', 'Email : test@gmail.com', 'Phone : 123', 'Gender : Male', and 'Citizenship : Indonesia'. It has an 'OK' button.

The screenshot shows a window titled "Login Page" with a standard Windows-style title bar (minimize, maximize, close buttons). Inside the window, there are two text input fields: "Email" and "Password". Below the "Password" field is a checkbox labeled "Remember Password". At the bottom of the window, there are two buttons: "REGISTER" and "LOGIN".

- Login data salah atau tidak lengkap

This screenshot shows the "Login Page" window with the "Email" field filled with "test@gmail.com" and the "Password" field empty. To the right of the login page, an "Error" dialog box is displayed. The dialog box has a red octagonal icon with a white "X" and the text "Silakan mengisi data dengan benar" (Please fill in the data correctly). There is an "OK" button at the bottom of the dialog box.

This screenshot shows the "Login Page" window with the "Email" field filled with "test@gmail.com" and the "Password" field filled with "122". To the right, an "Error" dialog box is displayed. The dialog box has a red octagonal icon with a white "X" and the text "Data yang anda masukkan salah" (The data you entered is incorrect). There is an "OK" button at the bottom of the dialog box.

- Login data sesuai

This screenshot shows the "Login Page" window with the "Email" field filled with "test@gmail.com" and the "Password" field filled with "123". To the right, a "Message" dialog box is displayed. The dialog box has a blue circular icon with a white "i" and the text "Selamat Datang Ferdy" (Welcome Ferdy). There is an "OK" button at the bottom of the dialog box.

Latihan 04

File Edit Help

## FORM PENILAIAN MATA KULIAH PBO

**Form Mahasiswa**

NPM

Nama

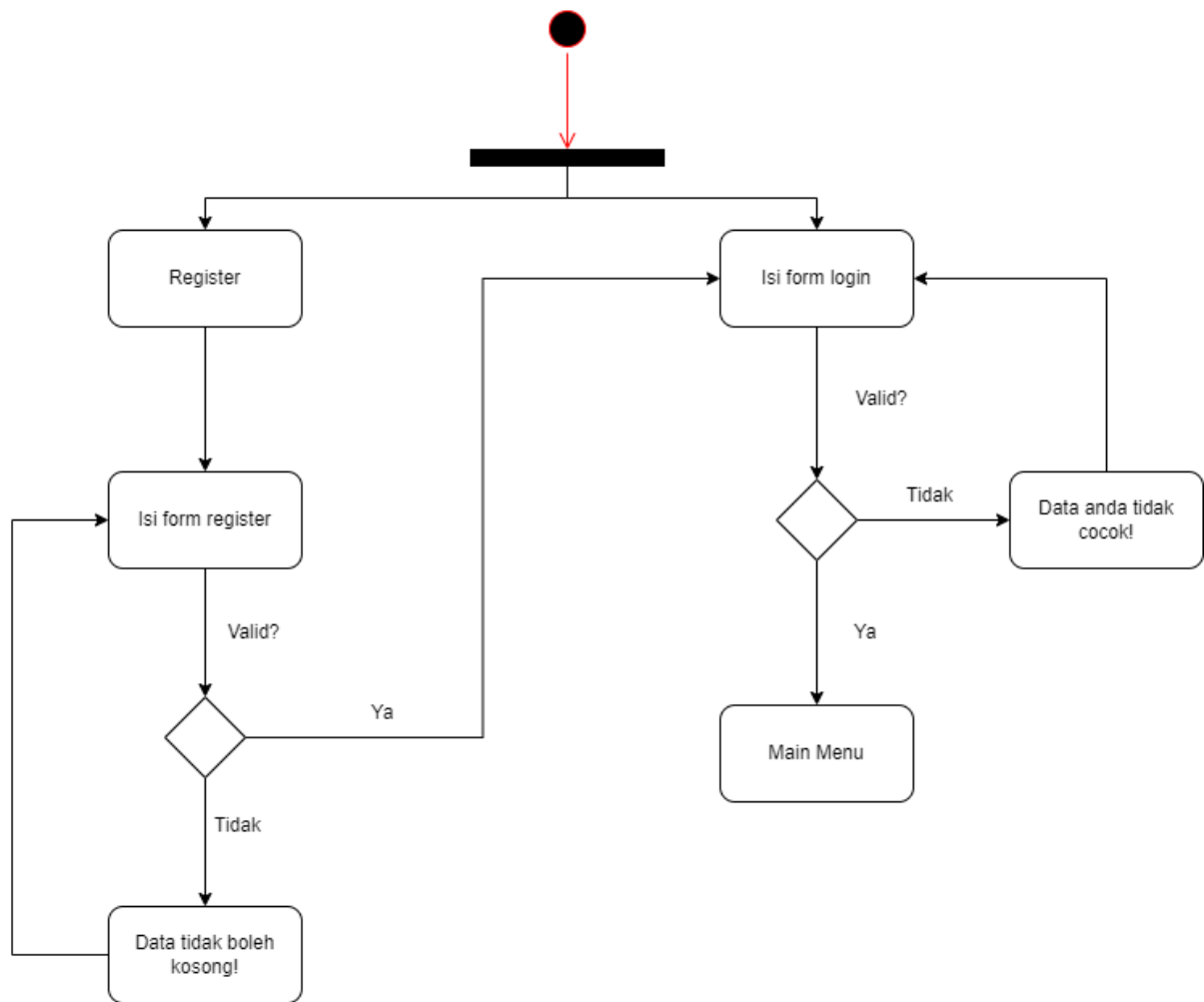
Nilai

☐ A ☐ B ☐ C ☐ D ☐ E ☐ F

NPM	Nama	Nilai
212310018	FATHURAHMAN AL FARIDZI	B
212310019	MUHAMMAD RAFI ZUHAIR ARTA	C
212310020	MUHAMAD SUBHAN RIZKI AFIA	A

2. Berdasarkan soal nomor 2 buatlah Activity Diagram.





### 3.. Buatlah program sederhana dengan menggunakan MouseListener sebagai berikut (TicTacToe):

#### a. Source Code

```
1 package com.ibik.pbo.praktikum;
2
3 import java.awt.EventQueue;
4
5 public class TicTacToe extends JFrame {
6
7     private JPanel contentPane;
8     final JButton button_1 = new JButton("");
9     final JButton button_2 = new JButton("");
10    final JButton button_3 = new JButton("");
11    final JButton button_4 = new JButton("");
12    final JButton button_5 = new JButton("");
13    final JButton button_6 = new JButton("");
14    final JButton button_7 = new JButton("");
15    final JButton button_8 = new JButton("");
16    final JButton button_9 = new JButton("");
17    String value = "";
18
19    /**
20     * Launch the application.
21     */
22    public static void main(String[] args) {
23        EventQueue.invokeLater(new Runnable() {
24            public void run() {
25                try {
26                    TicTacToe frame = new TicTacToe();
27                    frame.setVisible(true);
28                } catch (Exception e) {
29                    e.printStackTrace();
30                }
31            }
32        });
33    }
34
35    /**
36     * Create the frame.
37     */
38    public TicTacToe() {
39        setTitle("TicTacToe");
40        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
41        setBounds(100, 100, 318, 311);
42
43        contentPane = new JPanel();
44        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
45
46        setContentPane(contentPane);
47        contentPane.setLayout(new GridLayout(0, 3, 5, 0));
48
49        button_1.addMouseListener(new MouseAdapter() {
50            @Override
51            public void mouseClicked(MouseEvent e) {
52                if (value == "X") {
53                    button_1.setText("O");
54                    value = "O";
55                } else {
56                    button_1.setText("X");
57                    value = "X";
58                }
59                if (button_1.getText() == button_2.getText() && button_2.getText() == button_3.getText()) {
60                    JOptionPane.showMessageDialog(null, "Win");
61                }
62            }
63        });
64        contentPane.add(button_1);
65
66        button_2.addMouseListener(new MouseAdapter() {
67            @Override
68            public void mouseClicked(MouseEvent e) {
69                if (value == "X") {
70                    button_2.setText("O");
71                    value = "O";
72                } else {
73                    button_2.setText("X");
74                    value = "X";
75                }
76                if (button_1.getText() == button_2.getText() && button_2.getText() == button_3.getText()) {
77                    JOptionPane.showMessageDialog(null, "Win");
78                }
79            }
80        });
81        contentPane.add(button_2);
82
83        button_3.addMouseListener(new MouseAdapter() {
```

```

93  @Override
94  public void mouseClicked(MouseEvent e) {
95      if(value == "X") {
96          button_3.setText("O");
97          value = "O";
98      }else {
99          button_3.setText("X");
100         value = "X";
101     }
102     if (button_1.getText() == button_2.getText() && button_2.getText() == button_3.getText()) {
103         JOptionPane.showMessageDialog(null, "Win");
104     }
105 }
106 });
107 contentPane.add(button_3);
108
109 button_4.addMouseListener(new MouseAdapter() {
110     @Override
111     public void mouseClicked(MouseEvent e) {
112         if(value == "X") {
113             button_4.setText("O");
114             value = "O";
115         }else {
116             button_4.setText("X");
117             value = "X";
118         }
119     }
120 });
121 contentPane.add(button_4);
122
123 button_5.addMouseListener(new MouseAdapter() {
124     @Override
125     public void mouseClicked(MouseEvent e) {
126         if(value == "X") {
127             button_5.setText("O");
128             value = "O";
129         }else {
130             button_5.setText("X");
131             value = "X";
132         }
133     }
134 });
135 contentPane.add(button_5);
136
137 button_6.addMouseListener(new MouseAdapter() {
138     @Override
139     public void mouseClicked(MouseEvent e) {
140         if(value == "X") {
141             button_6.setText("O");
142             value = "O";
143         }else {
144             button_6.setText("X");
145             value = "X";
146         }
147     }
148 });
149 contentPane.add(button_6);
150
151 button_7.addMouseListener(new MouseAdapter() {
152     @Override
153     public void mouseClicked(MouseEvent e) {
154         if(value == "X") {
155             button_7.setText("O");
156             value = "O";
157         }else {
158             button_7.setText("X");
159             value = "X";
160         }
161     }
162 });
163 contentPane.add(button_7);
164
165 button_8.addMouseListener(new MouseAdapter() {
166     @Override
167     public void mouseClicked(MouseEvent e) {
168         if(value == "X") {
169             button_8.setText("O");
170             value = "O";
171         }else {
172             button_8.setText("X");
173             value = "X";
174         }
175     }
176 });

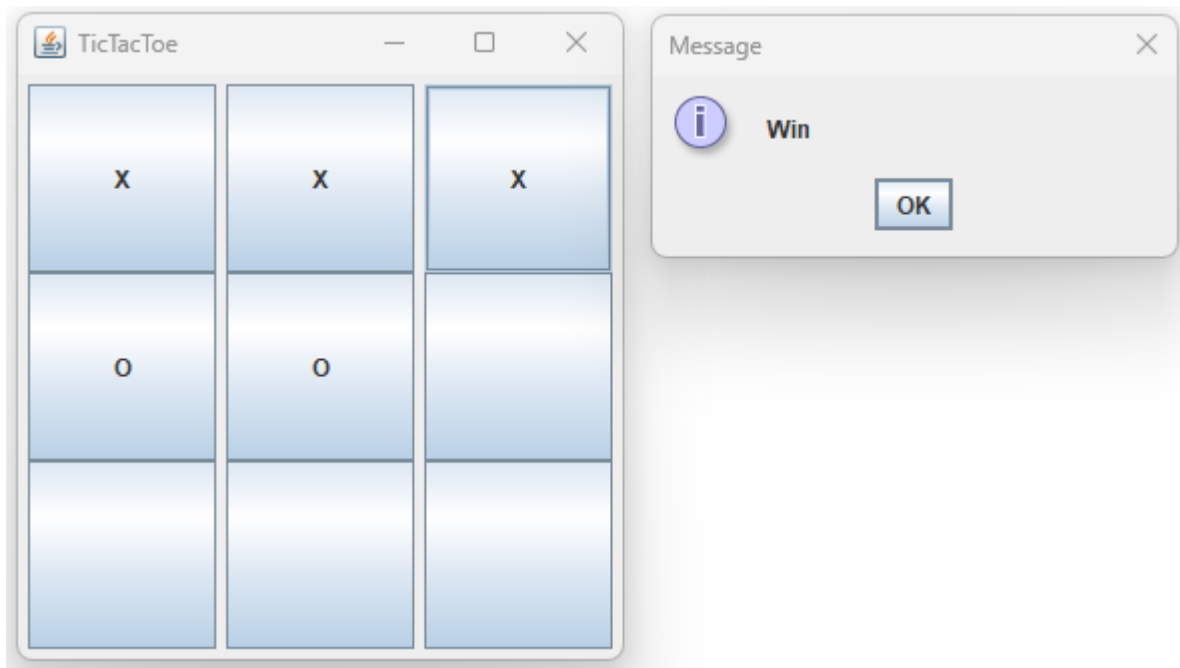
```

```

176         });
177         contentPane.add(button_8);
178
179         button_9.addMouseListener(new MouseAdapter() {
180             @Override
181             public void mouseClicked(MouseEvent e) {
182                 if(value == "X") {
183                     button_9.setText("O");
184                     value = "O";
185                 }else {
186                     button_9.setText("X");
187                     value = "X";
188                 }
189             }
190         });
191         contentPane.add(button_9);
192     }
193 }
194

```

#### b. Output



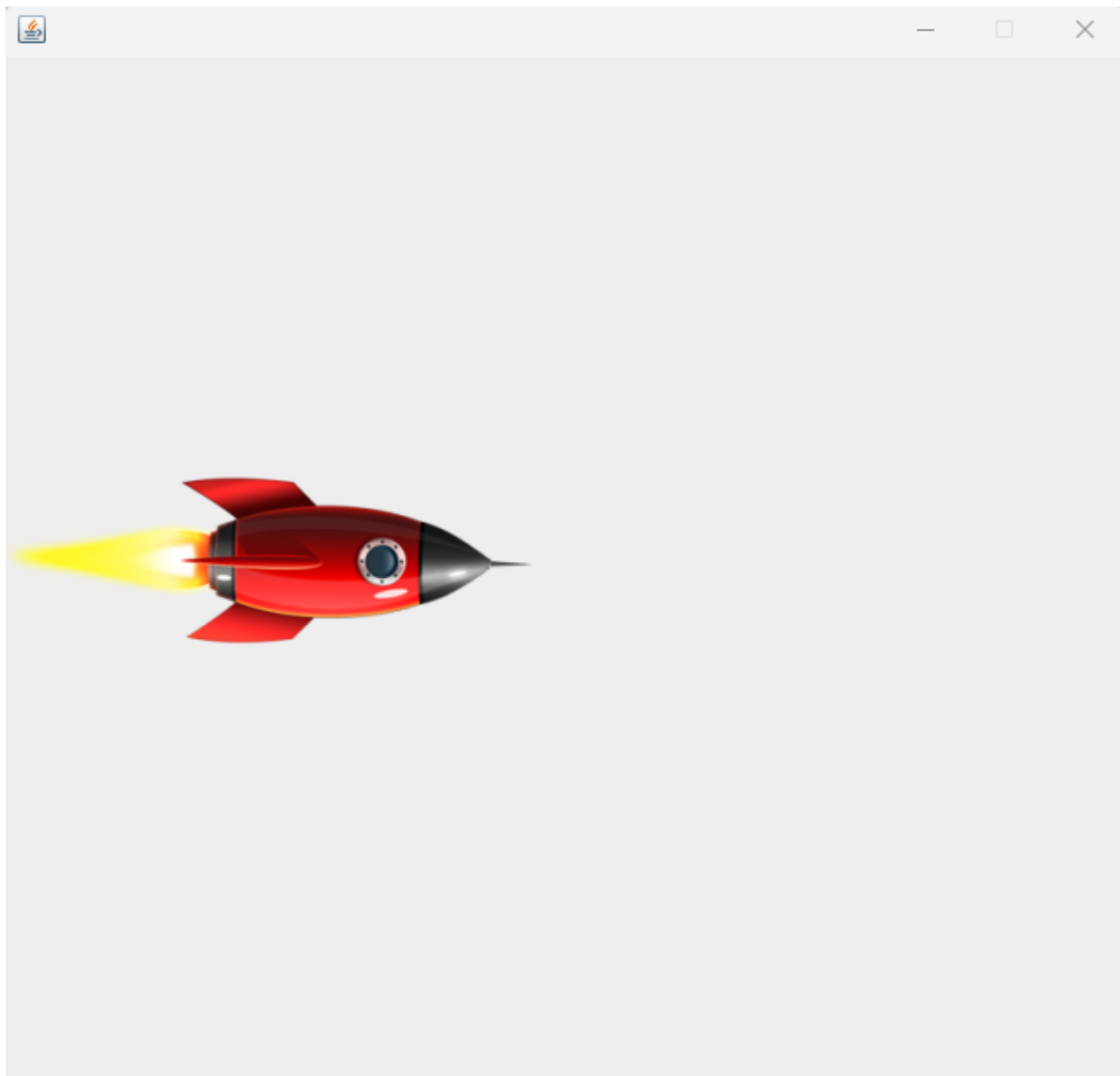
4. Buatlah program sederhana seperti dibawah ini dengan menggunakan KeyListener (Rocket)

## a. Source Code

```
1 package com.ibik.pbo.praktikum;
2
3 import java.awt.EventQueue;
4
5 public class RocketListener extends JFrame implements KeyListener {
6
7     private JPanel contentPane;
8     JLabel rocket_img = new JLabel();
9
10    /**
11     * Launch the application.
12     */
13    public static void main(String[] args) {
14        EventQueue.invokeLater(new Runnable() {
15            public void run() {
16                try {
17                    RocketListener frame = new RocketListener();
18                    frame.setVisible(true);
19                } catch (Exception e) {
20                    e.printStackTrace();
21                }
22            }
23        });
24    }
25
26    /**
27     * Create the frame.
28     */
29    public RocketListener() {
30        setResizable(false);
31
32        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
33        setBounds(100, 100, 656, 630);
34        contentPane = new JPanel();
35        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
36
37        setContentPane(contentPane);
38        setLocationRelativeTo(null);
39        contentPane.setLayout(null);
40        setVisible(true);
41
42        rocket_img.setBounds(0, 0, 548, 591);
43        String packageName = "com/ibik/pbo/praktikum/images/";
44
45        URL rocket_path = RocketListener.class.getClassLoader().getResource(packageName+"rocket.png");
46        final ImageIcon rocket = new ImageIcon(rocket_path);
47
48        URL rocketleft_path = RocketListener.class.getClassLoader().getResource(packageName+"rocket_left.png");
49        final ImageIcon rocketleft = new ImageIcon(rocketleft_path);
50
51        rocket_img.setIcon(rocket);
52        contentPane.add(rocket_img);
53
54        addKeyListener(new KeyAdapter() {
55            @Override
56            public void keyTyped(KeyEvent e) {
57                switch (e.getKeyChar()) {
58                    case 'a':
59                        rocket_img.setLocation(rocket_img.getX()-10, rocket_img.getY());
60                        break;
61                    case 'w':
62                        rocket_img.setLocation(rocket_img.getX(), rocket_img.getY()-10);
63                        rocket_img.setX();
64                        break;
65                    case 's':
66                        rocket_img.setLocation(rocket_img.getX(), rocket_img.getY()+10);
67                        break;
68                    case 'd':
69                        rocket_img.setLocation(rocket_img.getX()+10, rocket_img.getY());
70                        break;
71                    default:
72                        break;
73                }
74
75                if (rocket_img.getX() >= 340) {
76                    rocket_img.setIcon(rocketleft);
77                    rocket_img.setBounds(340, 0, 548, 591);
78                } else if (rocket_img.getX() <= 0) {
79                    rocket_img.setIcon(rocket);
80                    rocket_img.setBounds(0, 0, 548, 591);
81                }
82            }
83        });
84    }
85}
```

```
92     }
93
94     @Override
95     public void keyTyped(KeyEvent e) {
96         // TODO Auto-generated method stub
97     }
98
99
100    @Override
101    public void keyPressed(KeyEvent e) {
102        // TODO Auto-generated method stub
103    }
104
105
106    @Override
107    public void keyReleased(KeyEvent e) {
108        // TODO Auto-generated method stub
109    }
110
111 }
112
```

## b. Output



- Ketika mencapai ujung layer frame maka gambar akan berganti

