

LAPORAN AWAL LAPORAN PEMROGRAMAN I

LAPORAN KE-2



Disusun Oleh :

Nama :Febiyanto Rizki Qurbandi

NIM :231011450284

Kelas :04TPLP003

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan – Banten

Link GITHUB Repo:

<https://github.com/NeiaKI/Belajar-Coding/tree/main/Pemrograman-1%20Kuliah-Java/Pertemuan%2013>

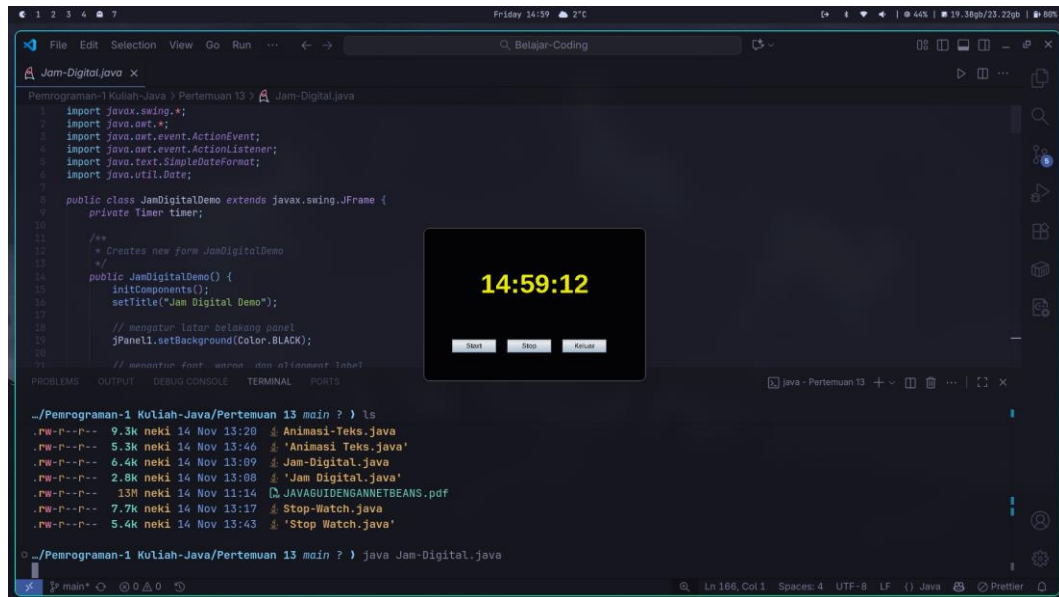
Code JAM DIGITAL

```

1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.ActionEvent;
4 import java.awt.event.ActionListener;
5 import java.text.SimpleDateFormat;
6 import java.util.Date;
7
8 public class JamDigitalDemo extends javax.swing.JFrame {
9     private Timer timer;
10
11     /**
12      * Creates new form JamDigitalDemo
13      */
14     public JamDigitalDemo() {
15         initComponents();
16         setTitle("Jam Digital Demo");
17
18         // mengatur latar belakang panel
19         JPanel1.setBackground(Color.BLACK);
20
21         // mengatur font dan alignment label
22         lblJamDigital.setFont(new Font("Arial", Font.BOLD, 48));
23         lblJamDigital.setForeground(Color.YELLOW);
24         lblJamDigital.setHorizontalAlignment(JLabel.CENTER);
25
26         // membuat timer dengan interval 1000ms (1 detik)
27         timer = new Timer(1000, new ActionListener() {
28             public void actionPerformed(ActionEvent e) {
29                 // mendapatkan waktu sistem sekarang
30                 Date now = new Date();
31
32                 // format waktu menggunakan SimpleDateFormat
33                 SimpleDateFormat sdf = new SimpleDateFormat("HH:mm:ss");
34
35                 // menampilkan waktu ke label
36                 lblJamDigital.setText(sdf.format(now));
37             }
38         });
39
40     }
41
42     /**
43      * This method is called from within the constructor to initialize the form.
44      */
45     private void initComponents() {
46         JPanel1 = new javax.swing.JPanel();
47         lblJamDigital = new javax.swing.JLabel();
48         btnStart = new javax.swing.JButton();
49         btnStop = new javax.swing.JButton();
50         btnKeluar = new javax.swing.JButton();
51
52         setDefaultCloseOperation(javax.swing.JFrame.EXIT_ON_CLOSE);
53         setResizable(false);
54
55         JPanel1.setBackground(new java.awt.Color(0, 0, 0));
56
57         lblJamDigital.setFont(new java.awt.Font("Arial", Font.BOLD, 48));
58         lblJamDigital.setForeground(new java.awt.Color(255, 255, 0));
59         lblJamDigital.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
60         lblJamDigital.setText("00:00:00");
61
62         btnStart.setText("Start");
63         btnStart.addActionListener(new java.awt.event.ActionListener() {
64             public void actionPerformed(java.awt.event.ActionEvent evt) {
65                 btnStartActionPerformed(evt);
66             }
67         });
68
69         btnStop.setText("Stop");
70         btnStop.addActionListener(new java.awt.event.ActionListener() {
71             public void actionPerformed(java.awt.event.ActionEvent evt) {
72                 btnStopActionPerformed(evt);
73             }
74         });
75
76         btnKeluar.setText("Keluar");
77         btnKeluar.addActionListener(new java.awt.event.ActionListener() {
78             public void actionPerformed(java.awt.event.ActionEvent evt) {
79                 btnKeluarActionPerformed(evt);
80             }
81         });
82
83         // layout menggunakan GroupLayout
84         javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(JPanel1);
85         JPanel1.setLayout(jPanel1Layout);
86
87         jPanel1Layout.setHorizontalGroup(
88             jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
89                 .addGroup(jPanel1Layout.createSequentialGroup()
90                     .addComponent(lblJamDigital, javax.swing.GroupLayout.DEFAULT_SIZE, 400, Short.MAX_VALUE)
91                     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
92                     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
93                         .addComponent(btnStart, javax.swing.GroupLayout.PREFERRED_SIZE, 80, javax.swing.GroupLayout.PREFERRED_SIZE)
94                         .addComponent(btnStop, javax.swing.GroupLayout.PREFERRED_SIZE, 80, javax.swing.GroupLayout.PREFERRED_SIZE)
95                         .addComponent(btnKeluar, javax.swing.GroupLayout.PREFERRED_SIZE, 80, javax.swing.GroupLayout.PREFERRED_SIZE)
96                     )
97                 )
98         );
99
100         jPanel1Layout.setVerticalGroup(
101             jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
102                 .addGroup(jPanel1Layout.createSequentialGroup()
103                     .addComponent(lblJamDigital, javax.swing.GroupLayout.PREFERRED_SIZE, 100, javax.swing.GroupLayout.PREFERRED_SIZE)
104                     .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
105                     .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
106                         .addComponent(btnStart)
107                         .addComponent(btnStop)
108                         .addComponent(btnKeluar)
109                     )
110                 )
111         );
112
113         javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
114         getContentPane().setLayout(layout);
115         layout.setHorizontalGroup(
116             layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
117                 .addComponent(JPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
118         );
119         layout.setVerticalGroup(
120             layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
121                 .addComponent(JPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
122         );
123
124         pack();
125         setLocationRelativeTo(null);
126     }
127
128     /**
129      * Menjalankan Timer untuk memulai tampilan jam digital
130      */
131     private void btnStartActionPerformed(java.awt.event.ActionEvent evt) {
132         timer.start();
133     }
134
135     /**
136      * Menghentikan Timer untuk berhenti tampilan jam digital
137      */
138     private void btnStopActionPerformed(java.awt.event.ActionEvent evt) {
139         timer.stop();
140     }
141
142     /**
143      * Menutup aplikasi
144      */
145     private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
146         System.exit(0);
147     }
148
149     /**
150      * Main method untuk menjalankan aplikasi
151      */
152     public static void main(String args[]) {
153         java.awt.EventQueue.invokeLater(new Runnable() {
154             public void run() {
155                 new JamDigitalDemo().setVisible(true);
156             }
157         });
158     }
159
160     // Deklarasi komponen
161     private javax.swing.JPanel jPanel1;
162     private javax.swing.JLabel lblJamDigital;
163     private javax.swing.JButton btnStart;
164     private javax.swing.JButton btnStop;
165     private javax.swing.JButton btnKeluar;
166 }

```

OUTPUT:



PENJELASAN:

- GUI Java Swing: Kode ini mendefinisikan sebuah kelas TampilanTimer yang merupakan turunan dari JFrame, digunakan untuk membuat jendela aplikasi berbasis grafis.
- Objek Timer: Di dalam konstruktor, sebuah objek Timer dibuat dan diatur dengan interval 1000 milidetik (1 detik).
- Logika Counter: Setiap kali timer berdetak (per detik), program akan:
 - Menambah variabel counter sebesar 1.
 - Memperbarui teks pada label (jLabel1) untuk menampilkan nilai counter terbaru ("Counter : [angka]").
- Tombol Mulai: Terdapat event handler jButton1ActionPerformed. Saat tombol diklik, perintah timer.start() dijalankan untuk memulai proses penghitungan waktu.
- Pengaturan Layout: Metode initComponents() (kode yang digenerate otomatis oleh IDE seperti NetBeans) bertugas mengatur posisi tombol, label, ukuran jendela, dan tata letak (layout) aplikasi.
- Metode Main: Titik awal program yang membuat dan menampilkan jendela aplikasi TampilanTimer agar terlihat di layar.
- Format Waktu: Waktu mentah diformat menjadi pola "HH:mm:ss" (Jam:Menit:Detik) menggunakan kelas SimpleDateFormat sebelum ditampilkan ke label.

- Fungsi Tombol:
 - Start: Memanggil timer.start() untuk mulai menjalankan jam.
 - Stop: Memanggil timer.stop() untuk membekukan waktu (jeda).
 - Keluar: Memanggil System.exit(0) untuk menutup program sepenuhnya.

Code STOP WATCH:

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.util.concurrent.TimeUnit;

public class StopwatchDemo extends javax.swing.JFrame {

    private static final String Start = "Start";

    private static final String Pause = "Pause";

    private static final String Reset = "Reset";

    private boolean isRunning;

    private Timer timer = new Timer(100, new WaktuListener());

    private long initTime = System.currentTimeMillis();

    private long startTime;

    private long pauseTime;

    /**
     * Creates new form StopwatchDemo
     */
    public StopwatchDemo() {

        initComponents();

        setTitle("Stop Watch Demo");

        jPanel1.setBackground(Color.BLACK);
```

```

        btnReset.setEnabled(false);

        setWaktu();
    }

    /**
     * Mengatur format waktu dan tampilan label
     */
    private void setWaktu() {
        lblWaktu.setFont(new Font("Arial", Font.BOLD, 48));
        lblWaktu.setForeground(Color.YELLOW);
        lblWaktu.setText("00:00:00.0"); // Mulai dari 00:00:00.0
        lblWaktu.setHorizontalAlignment(JLabel.CENTER);
    }

    /**
     * Memulai stop watch
     */
    public void start() {
        if (isRunning == false) {
            startTime = System.currentTimeMillis();
        } else {
            startTime = System.currentTimeMillis() - (pauseTime - startTime);
        }
        isRunning = true;
        timer.start();
    }

    /**
     * Menjeda stop watch
     */

```

```

public void pause() {

    pauseTime = System.currentTimeMillis();

    timer.stop();

}

/**
 * Mereset stop watch
 */

public void reset() {

    startTime = 0;

    isRunning = false;

    timer.stop();

    lblWaktu.setText("00:00:00.0"); // Reset ke 00:00:00.0

    btnStart.setText(Start);

}

/**
 * Memformat waktu menjadi format HH:MM:SS.ms
 */

private String formatWaktu(final long time) {

    final long hr = TimeUnit.MILLISECONDS.toHours(time);

    final long min = TimeUnit.MILLISECONDS.toMinutes(time -
        TimeUnit.HOURS.toMillis(hr));

    final long sec = TimeUnit.MILLISECONDS.toSeconds(time -
        TimeUnit.HOURS.toMillis(hr) -
        TimeUnit.MINUTES.toMillis(min));

    final long ms = TimeUnit.MILLISECONDS.toMillis(time -
        TimeUnit.HOURS.toMillis(hr) -
        TimeUnit.MINUTES.toMillis(min) -
        TimeUnit.SECONDS.toMillis(sec));

```

```

        return String.format("%02d:%02d:%02d.%01d", hr, min, sec, ms/100);
    }

    /**
     * Mendapatkan waktu saat ini
     */
    private String getCurrentTime(long time) {
        return formatWaktu(time);
    }

    /**
     * Inner class untuk mendengarkan event Timer
     */
    class WaktuListener implements ActionListener {
        public void actionPerformed(ActionEvent e) {
            lblWaktu.setText(getCurrentTime(System.currentTimeMillis() - startTime));
        }
    }

    /**
     * This method is called from within the constructor to initialize the form.
     */
    private void initComponents() {
        jPanel1 = new javax.swing.JPanel();
        lblWaktu = new javax.swing.JLabel();
        btnReset = new javax.swing.JButton();
        btnStart = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.JFrame.EXIT_ON_CLOSE);

```

```

setResizable(false);

jPanel1.setBackground(new java.awt.Color(0, 0, 0));

lblWaktu.setFont(new java.awt.Font("Arial", Font.BOLD, 48));
lblWaktu.setForeground(new java.awt.Color(255, 255, 0));
lblWaktu.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
lblWaktu.setText("00:00:00.0");

btnStart.setText("Start");
btnStart.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnStartActionPerformed(evt);
    }
});

btnReset.setText("Reset");
btnReset.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnResetActionPerformed(evt);
    }
});

// layout menggunakan GroupLayout
javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(lblWaktu, javax.swing.GroupLayout.DEFAULT_SIZE, 400, Short.MAX_VALUE)

```



```

        .addGroup(jPanel1Layout.createSequentialGroup())

        .addGap(100, 100, 100)

        .addComponent(btnStart, javax.swing.GroupLayout.PREFERRED_SIZE, 80,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(50, 50, 50)

        .addComponent(btnReset, javax.swing.GroupLayout.PREFERRED_SIZE, 80,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap(90, Short.MAX_VALUE))

    );

    jPanel1Layout.setVerticalGroup(

        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel1Layout.createSequentialGroup())

        .addGap(50, 50, 50)

        .addComponent(lblWaktu, javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(50, 50, 50)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

            .addComponent(btnStart)

            .addComponent(btnReset))

        .addContainerGap(50, Short.MAX_VALUE))

    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);

    layout.setHorizontalGroup(

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

    );

    layout.setVerticalGroup(

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        );

        pack();

        setLocationRelativeTo(null);
    }

    /**
     * Event handler untuk tombol Start/Pause
     */
    private void btnStartActionPerformed(java.awt.event.ActionEvent evt) {

        if (btnStart.getText().equals(Start)) {

            btnStart.setText(Pause);

            btnReset.setEnabled(false);

            start();

        } else {

            btnStart.setText(Start);

            btnReset.setEnabled(true);

            pause();

        }

    }

    /**
     * Event handler untuk tombol Reset
     */
    private void btnResetActionPerformed(java.awt.event.ActionEvent evt) {

        reset();

    }

    /**

```

```
* Main method untuk menjalankan aplikasi
*/

public static void main(String args[]) {

    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {

            new StopwatchDemo().setVisible(true);

        }

    });

}

// Deklarasi komponen

private javax.swing.JPanel jPanel1;

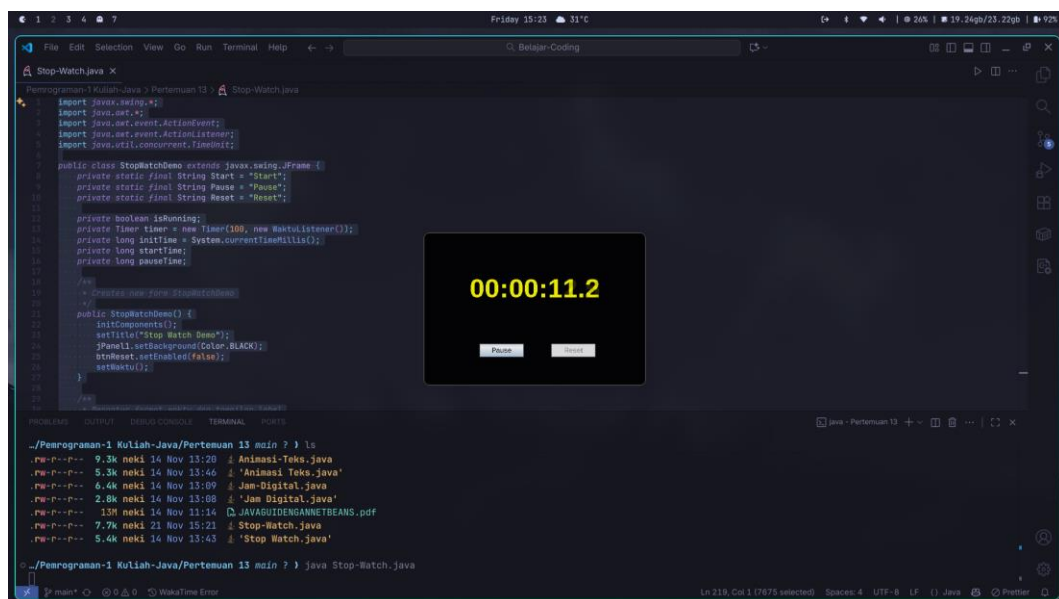
private javax.swing.JLabel lblWaktu;

private javax.swing.JButton btnReset;

private javax.swing.JButton btnStart;

}
```

Output :

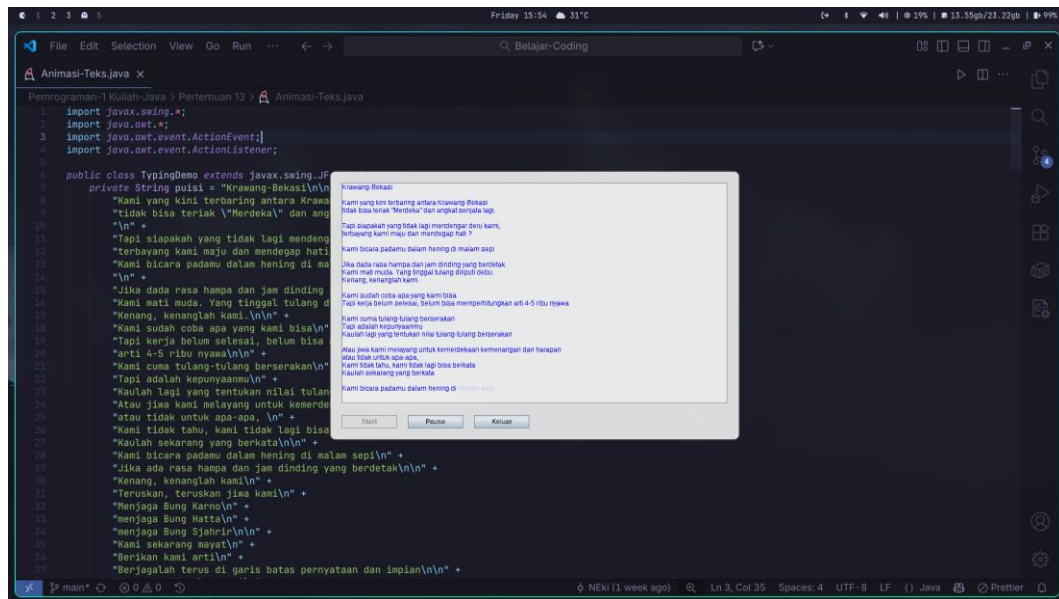


Penjelasan:

- Tujuan Program: Membuat aplikasi Stopwatch yang memiliki fungsi Start (Mulai), Pause (Jeda), dan Reset.
- Logika Perhitungan Waktu (Time Delta): Berbeda dengan program counter sebelumnya yang hanya menambah angka (+1), kode ini menghitung selisih waktu nyata.
 - Rumusnya: Waktu Sekarang - Waktu Mulai.
 - Ini membuat perhitungan waktu tetap akurat sesuai jam sistem komputer.
- Logika Tombol Toggle (Start/Pause): Tombol btnStart memiliki fungsi ganda:
 - Jika tulisan "Start": Timer berjalan, tombol berubah jadi "Pause", tombol Reset dimatikan (disable).
 - Jika tulisan "Pause": Timer berhenti, tombol berubah jadi "Start", tombol Reset diaktifkan (enable).
- Mekanisme Pause yang Cerdas:
 - Saat tombol "Start" ditekan kembali setelah Pause, program tidak memulai dari nol.
 - Program menggunakan rumus $startTime = System.currentTimeMillis() - (pauseTime - startTime)$ untuk memastikan waktu berlanjut mulus dari angka terakhir, seolah-olah waktu "berhenti" saat dipause.
- Format Tampilan (TimeUnit):
 - Menggunakan TimeUnit untuk mengonversi ribuan milidetik menjadi format Jam, Menit, Detik, dan Milidetik.
 - Hasilnya diformat string menjadi 00:00:00.0.
- Timer UI: Objek Timer diset dengan interval 100ms (0.1 detik) hanya untuk memperbarui tampilan label agar terlihat berjalan mulus, bukan untuk menghitung logika waktunya.

Code:

Output:



Penjelasan:

- **Tujuan Program:** Membuat efek "mesin ketik" (typing animation) di mana teks puisi muncul karakter demi karakter secara otomatis.
- **Data Teks:** Sebuah variabel String puisi menampung seluruh teks puisi "Karawang-Bekasi" karya Chairil Anwar yang akan ditampilkan.
- **Variabel Index:** int charIndex digunakan sebagai penanda posisi karakter yang sedang ditampilkan saat ini (dimulai dari 0).
- **Mekanisme Timer:**
 - Objek Timer diset dengan interval **100 ms**.
 - Setiap detak timer, program mengambil satu karakter dari string puisi berdasarkan charIndex.
 - Karakter tersebut ditambahkan ke JTextArea (txaPuisi.setText(...)).
 - charIndex ditambah 1 (charIndex++) untuk persiapan karakter selanjutnya.
- **Kondisi Berhenti:** Jika charIndex sudah mencapai panjang total string puisi (puisi.length()), timer dimatikan (timer.stop()) dan tombol Start diaktifkan kembali.
- **Logika Tombol:**
 - **Start:** Memulai timer. Jika puisi sudah selesai, teks direset kosong dan indeks dikembalikan ke 0 untuk mulai ulang.
 - **Pause/Continue:** Tombol ini berfungsi sebagai *toggle*.
 - Jika diklik saat berjalan ("Pause"), timer dihentikan (stop).
 - Jika diklik saat berhenti ("Continue"), timer dilanjutkan (start).