



# TUGAS & EVALUASI

## Soal Tugas & Evaluasi

1. Buatlah ArrayList dari Class Transaksi dengan atribut tanggal(string) dan nominal(float) lalu Inputkan 5 Object ke dalam ArrayList tersebut!

## Jawaban

### Source Code

```
import java.util.ArrayList;

class Transaksi {
    private String tanggal;
    private float nominal;

    // Constructor
    public Transaksi(String tanggal, float nominal) {
        this.tanggal = tanggal;
        this.nominal = nominal;
    }

    // Getter untuk tanggal
    public String getTanggal() {
        return tanggal;
    }

    // Getter untuk nominal
    public float getNominal() {
        return nominal;
    }

    // Overriding toString() untuk mempermudah tampilan data
    @Override
    public String toString() {
        return "Tanggal: " + tanggal + ", Nominal: " + nominal;
    }
}
```



# TUGAS & EVALUASI

## Penjelasan

### Kelas Transaksi:

- Memiliki dua atribut: tanggal (tipe String) dan nominal (tipe float).
- Memiliki konstruktor untuk menginisialisasi atribut.
- Memiliki metode getter untuk masing-masing atribut.
- toString() di-override untuk memudahkan dalam menampilkan data.

## Source Code

```
public class main {  
    public static void main(String[] args) {  
        // Membuat ArrayList untuk menyimpan objek Transaksi  
        ArrayList<Transaksi> daftarTransaksi = new  
ArrayList<>();  
  
        // Menambahkan 5 objek Transaksi ke dalam ArrayList  
        daftarTransaksi.add(new Transaksi("2024-12-01", 50000));  
        daftarTransaksi.add(new Transaksi("2024-12-02",  
150000));  
        daftarTransaksi.add(new Transaksi("2024-12-03", 75000));  
        daftarTransaksi.add(new Transaksi("2024-12-04",  
200000));  
        daftarTransaksi.add(new Transaksi("2024-12-05",  
100000));  
  
        // Menampilkan daftar transaksi  
        System.out.println("Daftar Transaksi:");  
        for (Transaksi transaksi : daftarTransaksi) {  
            System.out.println(transaksi);  
        }  
    }  
}
```



# TUGAS & EVALUASI

## Penjelasan

### Kelas Main:

- Membuat ArrayList<Transaksi> untuk menyimpan objek Transaksi.
- Menambahkan 5 objek Transaksi ke dalam ArrayList.
- Menggunakan perulangan for-each untuk menampilkan semua transaksi.

## Output

Daftar Transaksi: Tanggal: 2024-12-01, Nominal: 50000.0 Tanggal: 2024-12-02, Nominal: 150000.0 Tanggal: 2024-12-03, Nominal: 75000.0 Tanggal: 2024-12-04, Nominal: 200000.0 Tanggal: 2024-12-05, Nominal: 100000.0	
---	--



# TUGAS & EVALUASI

## Soal Tugas & Evaluasi

2. Lalu buatlah method “exportToTxt” yang dimana method tersebut akan:

- Membuat file txt baru.
- Menuliskan semua value yang terdapat pada ArrayList ke dalam file baru yang telah dibuat.

## Jawaban

### Source Code

```
import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;

class Transaksi {
    private String tanggal;
    private float nominal;

    // Constructor
    public Transaksi(String tanggal, float nominal) {
        this.tanggal = tanggal;
        this.nominal = nominal;
    }

    // Overriding toString() untuk menampilkan data sebagai teks
    @Override
    public String toString() {
        return "Tanggal: " + tanggal + ", Nominal: " + nominal;
    }
}
```

## Penjelasan

### Kelas Transaksi:

- Memiliki dua atribut: tanggal (tipe String) dan nominal (tipe float).



# TUGAS & EVALUASI

- Memiliki konstruktor untuk menginisialisasi atribut.
- Memiliki metode getter untuk masing-masing atribut.
- toString() di-override untuk memudahkan dalam menampilkan data.

## Source Code

```
public class main {
    public static void main(String[] args) {
        // Membuat ArrayList berisi objek Transaksi
        ArrayList<Transaksi> daftarTransaksi = new
ArrayList<>();
        daftarTransaksi.add(new Transaksi("2024-12-01",
50000.0f));
        daftarTransaksi.add(new Transaksi("2024-12-02",
150000.0f));
        daftarTransaksi.add(new Transaksi("2024-12-03",
75000.0f));
        daftarTransaksi.add(new Transaksi("2024-12-04",
200000.0f));
        daftarTransaksi.add(new Transaksi("2024-12-05",
100000.0f));

        // Memanggil method exportToTxt
        exportToTxt("transaksi.txt", daftarTransaksi);
    }

    // Method exportToTxt
    public static void exportToTxt(String fileName,
ArrayList<Transaksi> daftarTransaksi) {
        try (FileWriter writer = new FileWriter(fileName)) {
            // Menulis setiap elemen dari ArrayList ke dalam
file
            for (Transaksi transaksi : daftarTransaksi) {
                writer.write(transaksi.toString() + "\n");
            }
            System.out.println("Data berhasil diekspor ke file:
" + fileName);
        } catch (IOException e) {
```



# TUGAS & EVALUASI

```
        System.out.println("Terjadi kesalahan saat menulis  
ke file: " + e.getMessage());  
    }  
}  
}
```

## Penjelasan

### Method `exportToTxt`:

- Parameter `fileName` menentukan nama file yang akan dibuat.
- Parameter `daftarTransaksi` adalah `ArrayList` yang berisi objek-objek `Transaksi`.
- Menggunakan **`FileWriter`** untuk menulis ke file.

### Blok `try-with-resources`:

- Membuka `FileWriter` dalam blok `try-with-resources` untuk memastikan file tertutup otomatis setelah penulisan selesai.

### Iterasi dan Penulisan:

- Iterasi setiap elemen di dalam `daftarTransaksi` menggunakan perulangan `for-each`.
- Setiap elemen dikonversi ke string menggunakan metode `toString()` (yang sudah di-override pada kelas `Transaksi`) dan ditulis ke file.

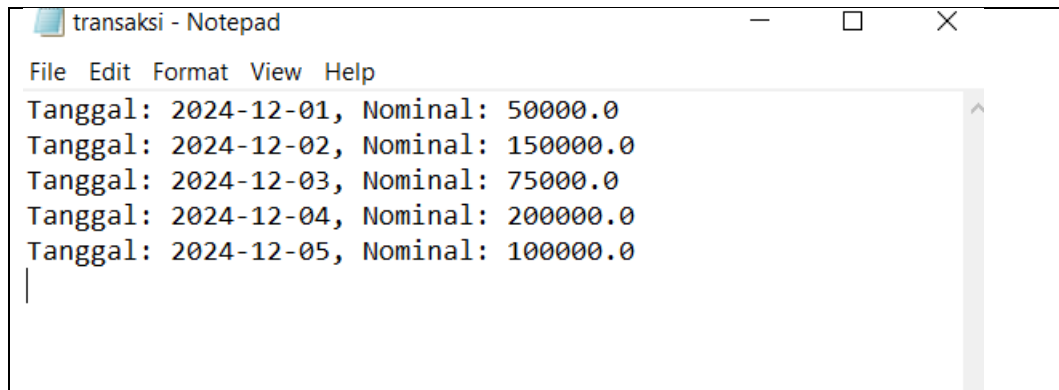
### Error Handling:

- Jika terjadi kesalahan (misalnya, file tidak bisa dibuat), pesan kesalahan ditampilkan dengan detail menggunakan `e.getMessage()`.

## Output



# TUGAS & EVALUASI



## Soal Tugas & Evaluasi

3. Buatlah sebuah program untuk menampilkan sebuah gambar yang ada pada directory anda menggunakan Image Class!

## Jawaban

### Source Code

```
import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.SwingUtilities;

public class image {
    public static void main(String[] args) {
        SwingUtilities.invokeLater(()->{
            JFrame frame = new JFrame("Gambar");
            frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE)
;
        });
    }
}
```



# TUGAS & EVALUASI

```
frame.setSize(800,600);

        ImageIcon imageIcon = new ImageIcon("E:/Kuliah/Tugas
Kuliah praktikum dan lain lain/Praktikum PBO/pertemuan 5/Program
tugas eva/tugas eva bab 10/no 3/ghost.jpg");

        JLabel label = new JLabel(imageIcon);

        frame.getContentPane().add(label);
        frame.setVisible(true);
    });
}
```

## Penjelasan

### Jendela Utama:

- Membuat jendela (JFrame) dengan judul "Gambar" dan ukuran 800x600 piksel.
- Menutup jendela akan mengakhiri program.

### Memuat Gambar:

- Menggunakan ImageIcon untuk memuat gambar dari path spesifik (ghost.jpg).

### Menambahkan Gambar:

- Gambar dimasukkan ke dalam JLabel, kemudian ditambahkan ke jendela.

### Menampilkan Jendela:

- Jendela dibuat terlihat dengan setVisible(true).





# TUGAS & EVALUASI

## Output





# TUGAS & EVALUASI

## Soal Tugas & Evaluasi

4. Buatlah program untuk menampilkan sebuah persegi Panjang menggunakan BufferedImage Class!

## Jawaban

### Source Code

```
import java.awt.*;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;
import javax.imageio.ImageIO;
import javax.swing.*;

public class PersegiPanjangApp {
    public static void main(String[] args) {
        // Membuat jendela utama
        JFrame frame = new JFrame("Gambar Persegi Panjang");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setSize(400, 300);

        // Menambahkan panel untuk menggambar
        PersegiPanjangPanel panel = new PersegiPanjangPanel();
        frame.add(panel);

        // Menampilkan jendela
        frame.setVisible(true);

        // Menyimpan gambar ke file
        try {
            panel.simpanGambar("persegi_panjang.png");
            System.out.println("Gambar berhasil disimpan sebagai 'persegi_panjang.png'");
        } catch (IOException e) {
            System.out.println("Gagal menyimpan gambar: " + e.getMessage());
        }
    }
}
```



# TUGAS & EVALUASI

```
}  
}  
}
```

## Penjelasan

### Kelas PersegiPanjangApp

Kelas utama yang menjalankan program dan bertanggung jawab untuk:

- Membuat jendela GUI (JFrame).
- Menambahkan panel khusus (PersegiPanjangPanel) ke dalam jendela.
- Menyimpan gambar persegi panjang ke file menggunakan metode di panel.

## Source Code

```
class PersegiPanjangPanel extends JPanel {  
    private BufferedImage image;  
  
    public PersegiPanjangPanel() {  
        // Membuat BufferedImage dengan ukuran 400x300  
        image = new BufferedImage(400, 300,  
BufferedImage.TYPE_INT_ARGB);  
        Graphics2D g2d = image.createGraphics();  
  
        // Mengisi background dengan warna putih  
        g2d.setColor(Color.WHITE);  
        g2d.fillRect(0, 0, image.getWidth(), image.getHeight());  
  
        // Menggambar persegi panjang  
        g2d.setColor(Color.CYAN);  
        g2d.fillRect(100, 50, 200, 100); // (x, y, width,  
height)  
  
        // Membersihkan resource grafik  
        g2d.dispose();  
    }  
}
```



# TUGAS & EVALUASI

```
@Override
protected void paintComponent(Graphics g) {
    super.paintComponent(g);

    // Menampilkan BufferedImage di panel
    g.drawImage(image, 0, 0, null);
}

// Metode untuk menyimpan gambar ke file
public void simpanGambar(String fileName) throws IOException
{
    File outputfile = new File(fileName);
    ImageIO.write(image, "png", outputfile);
}
}
```

## Penjelasan

### Kelas PersegiPanjangPanel

Kelas ini memperluas **JPanel** dan bertanggung jawab untuk:

- Membuat gambar (BufferedImage) dengan persegi panjang.
- Menggambar gambar ke panel saat GUI ditampilkan.
- Menyediakan fungsi untuk menyimpan gambar ke file.

## Output



# TUGAS & EVALUASI

