**TUGAS PERTEMUAN KE - 2**

**(PRAKTIKUM OOP 2022-2023)**

|  |  |
| --- | --- |
| **NAMA** | Mizaell Phinehas A. M. |
| **NIM** | 2218005 |
| **KELAS** | A |
| **PEMBERI TUGAS** | MOHAMMAD HARIFIN(2118131) |

## Tugas Rumah 2:

Judul : Sistem Informasi Perpustakaan

Tema : Sistem Informasi

Class diagram (Class Perpustakaan.java):

**Konstraktor**

**Method**

**Attribute**

**Class**

|  |
| --- |
| Perpustakaan |
| * daftarBuku: ArrayList<String> |
| * Perpustakaan() * tambahBuku(judul: String): void * getDaftarBuku(): ArrayList<String> |

Class diagram (Class GUIperpustakaan.java):

**Konstraktor**

**Method**

**Attribute**

**Class**

|  |
| --- |
| GUIperpustakaan |
| * perp: Perpustakaan * jLabel1: JLabel * jLabel2: JLabel * jLabel3: JLabel * jLabel4: JLabel * jTextField1: JTextField * jTextField2: JTextField * jTextField3: JTextField * jButton1: JButton * jButton2: JButton * jScrollPane1: JScrollPane * jTextArea1: JTextArea * jScrollPane2: JScrollPane * jTextArea2: JTextArea |
| * Perpustakaan() * GUIperpustakaan() * initComponents() * jButton1ActionPerformed(evt: ActionEvent): void * jTextField3ActionPerformed(evt: ActionEvent): void * jTextField2ActionPerformed(evt: ActionEvent): void * jButton2ActionPerformed(evt: ActionEvent): void * refreshTextArea(): void * main(args: String[]): void |

*Source code Object Class*(Perpustakaan.java):

**Method**

|  |
| --- |
| public class Perpustakaan {    // DaftarBuku adalah ArrayList yang digunakan untuk menyimpan judul buku-buku dalam perpustakaan  private ArrayList<String> daftarBuku;  // Konstruktor Perpustakaan. Ini akan dijalankan ketika objek Perpustakaan dibuat.  public Perpustakaan() {  // Inisialisasi daftarBuku sebagai ArrayList kosong saat objek dibuat.  daftarBuku = new ArrayList<>();  }  // Metode untuk menambahkan buku baru ke daftar buku dalam perpustakaan.  public void tambahBuku(String judul) {  // Menambahkan judul buku baru ke daftarBuku.  daftarBuku.add(judul);  }  // Metode untuk mengambil daftar buku dalam perpustakaan.  public ArrayList<String> getDaftarBuku() {  // Mengembalikan daftarBuku sebagai ArrayList dari judul buku-buku dalam perpustakaan.  return daftarBuku;  }  } |

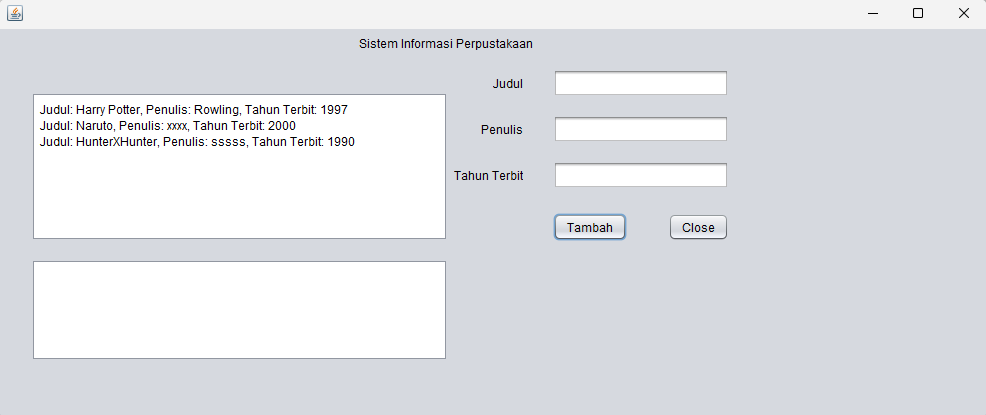
*Source code Driver Class*(Main.java):

|  |
| --- |
| public class Main {  public static void main(String[] args) {  Perpustakaan perpustakaan = new Perpustakaan();  // Inisialisasi Perpustakaan  // Tambahkan beberapa buku ke perpustakaan  perpustakaan.tambahBuku("Java Programming");  perpustakaan.tambahBuku("Data Structures");  perpustakaan.tambahBuku("Algorithms");  // Tampilkan daftar buku dari perpustakaan  System.out.println("Daftar Buku di Perpustakaan:");  for (String buku : perpustakaan.getDaftarBuku()) {  System.out.println(buku);  }  }  } |

*Source code Button*:

|  |
| --- |
| private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  // Mengambil data dari input pengguna  String judul = jTextField1.getText();  String penulis = jTextField2.getText();  String tahunTerbit = jTextField3.getText();  // Memeriksa apakah ada data yang kosong  if (judul.isEmpty() || penulis.isEmpty() || tahunTerbit.isEmpty()) {  // Menampilkan pesan kesalahan jika ada data yang kosong  jTextArea2.setText("Data tidak lengkap. Mohon isi semua field.");  } else {  // Jika semua data diisi, buat informasi buku dan tambahkan ke daftar buku  String informasiBuku = "Judul: " + judul + ", Penulis: " + penulis + ", Tahun Terbit: " + tahunTerbit;  perp.tambahBuku(informasiBuku);  // Membersihkan input dan textarea  refreshTextArea();  jTextField1.setText("");  jTextField2.setText("");  jTextField3.setText("");  jTextArea2.setText("");  }  }  private void refreshTextArea() {  // Membersihkan jTextArea1 dan menampilkan daftar buku yang telah dimasukkan  // Membersihkan jTextArea1  jTextArea1.setText("");  // Mengambil daftar buku dari objek 'perp'  ArrayList<String> daftarBuku = perp.getDaftarBuku();  // Menampilkan daftar buku dalam jTextArea1  for (String buku : daftarBuku) {  jTextArea1.append(buku + "\n");  }  } |

Hasil Tampilan:



Gambar 2.2 Tampilan Hasil GUIperpustakaan.java

Analisa:

konstruktor adalah metode khusus yang digunakan untuk menginisialisasi objek kelas. Dalam Perpustakaan.java, konstruktor kelas Perpustakaan digunakan untuk menginisialisasi objek yang akan digunakan untuk menyimpan daftar buku. Dalam GUIperpustakaan.java, konstruktor kelas GUIperpustakaan digunakan untuk menginisialisasi GUI dan objek yang akan digunakan dalam aplikasi manajemen perpustakaan.

## Kesimpulan

* + - 1. Constructor adalah method khusus yang otomatis dieksekusi pada saat menginstansiasi object dari class tertentu.
      2. Constructor digunakan khusus untuk membuat dan menginisialisasi objek baru, sehingga tujuan utamanya untuk memberikan atau mendefinisikan nilai awal pada sebuah attribute di dalam class.
      3. Default constructor adalah constructor tanpa parameter yang dibuat secara otomatis, jika tidak ada constructor lain yang didefinisikan dalam sebuah kelas.