**LAPORAN PRAKTIKUM**

**MODUL I**

**Pengertian OOP**

****

Nama :

Zarif Afzal Ramadhan (21103097)

Dosen :

Nicolaus Euclides Wahyu Nugroho, S. Kom., M.Cs

**PROGRAM STUDI S1 SISTEM INFORMASI**

**FAKULTAS INFORMATIKA**

**INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

**2022**

# I. TUJUAN

1. Memahami lingkungan dasar NetBeans IDE serta cara meng-compile dan menjalankan program
2. Membandingkan pemrograman terstruktur dengan pemrograman berorientasi objek dengan membuat program dalam bahasa C++ dan Java.

**II. TOOL**

1. CodeBlocks / Borland C++ / Visual C++
2. NetBeans IDE 8.1
3. Java SE Development Kit 8

# III. DASAR TEORI

1. **Pengantar Pemrograman Berorientasi Objek**

**Apa itu OOP ?** Merupakan teknik membuat suatu program berdasarkan objek dan apa yang bisa dilakukan objek tersebut. Object-oriented program terdiri dari objek-objek yang berinteraksi satu sama lain untuk menyelesaikan sebuah tugas.

**Kenapa menggunakan OOP ?** Kode-kode di-breakdown agar lebih mudah di-manage. Breakdown berdasarkan objek-objek yang ada pada program tersebut. Dianjurkan diimplementasikan untuk program dengan berbagai ukuran karena lebih mudah untuk men-debug. Pemograman procedural mengatur program dalam barisan-barisan linier yang bekerja dari atas ke bawah. Kumpulan tahapan yang dijalankan setelah yang lain berjalan. Baik untuk program kecil yang berisi sedikit code. Tidak dianjurkan diimplementasikan pada program berukuran besar, karena susah untuk dimanage dan di-debug.

**OOP mempunyai karakteristik sebagai berikut :**

1. Abstraksi

Menemukan hal-hal yang penting pada suatu objek dan mengabaikan hal-hal yang sifatnya incidental.

1. Enkapsulasi

Pengkapsulan adalah proses pemaketan data objek bersama method-methodnya

1. Pewarisan (Inheritance)

Proses penciptaan kelas baru (subclass/kelas turunan) dengan mewarisi karakteristik dari kelas yang udah ada (superclass/kelas induk), ditambah karakteristik unik kelas baru itu.

1. Reuseability

Reuseability adalah kemampuan untuk menggunakan kembali kelas yang sudah ada.

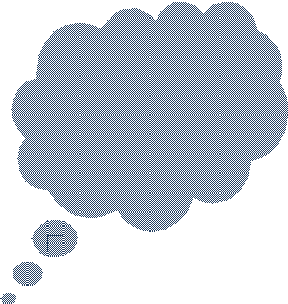
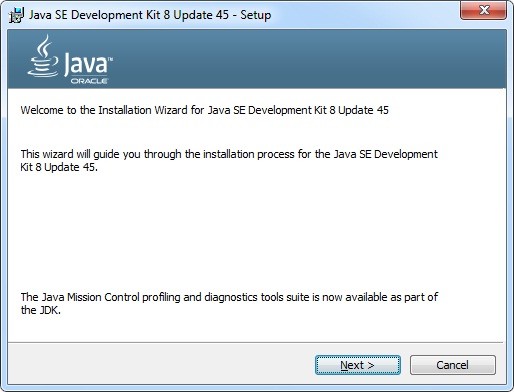
1. Polymorphism

Polymorphism berasal dari Bahasa Yunani yang berarti benyak bentuk.

1. **Installasi Java SE Development Kit 8**

Langkah-langkah Installasi Java SE Development Kit 8 :

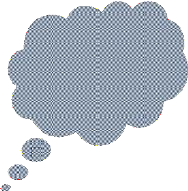
1. Pilih File Setup, kemudian akan muncul tampilan sebagai berikut :



Klik

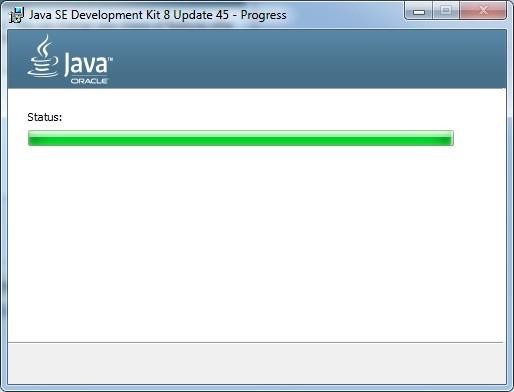
Next

**Gambar 1.** Java SE Development Kit 8 Setup



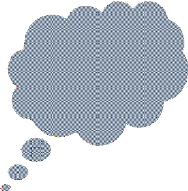
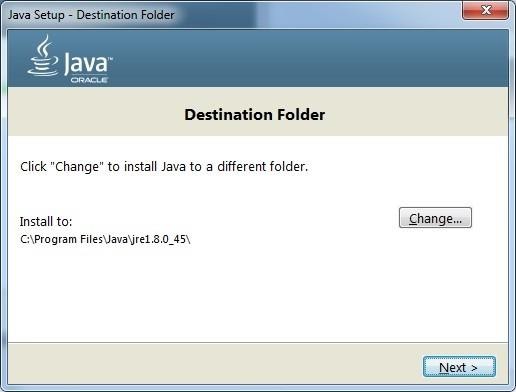
Klik

**Gambar 2.** Java SE Development Kit 8 Custom Setup



**Gambar 3.** Java SE Development Kit 8 Custom Setup

* + 1. Pilih destination Folder dimana JDK tersebut mau diinstallasi, seperti pada tampilan sebagai berikut :



Klik

**Gambar 4.** Java SE Development Kit 8 Custom Setup

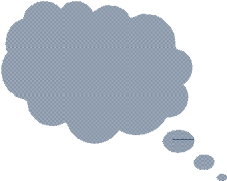
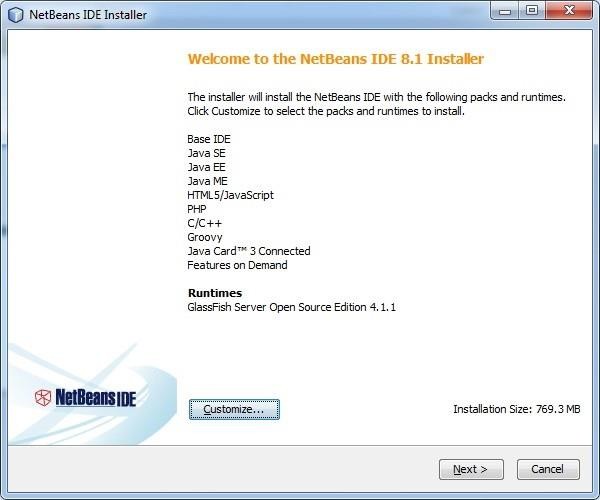


**Gambar 5.** Java SE Development Kit 8 Complete

* 1. **Installasi NetBeans IDE 8.1**

Langkah-langkah Installasi NetBeans IDE 8.1 :

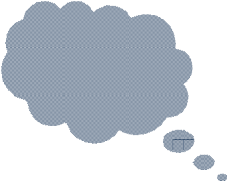
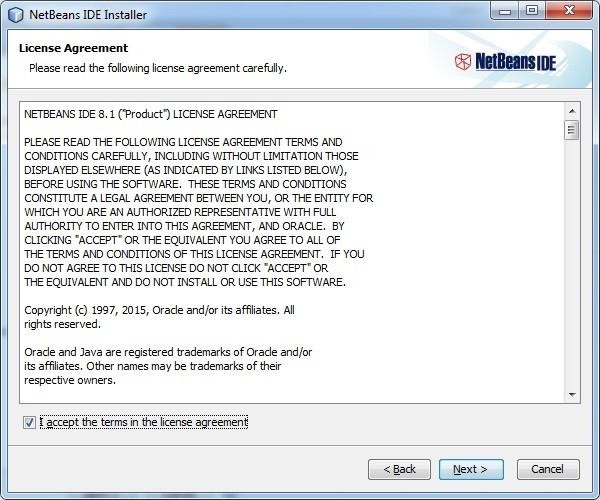
* + 1. Pilih File Setup, kemudian akan muncul tampilan sebagai berikut :



Klik

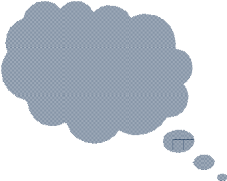
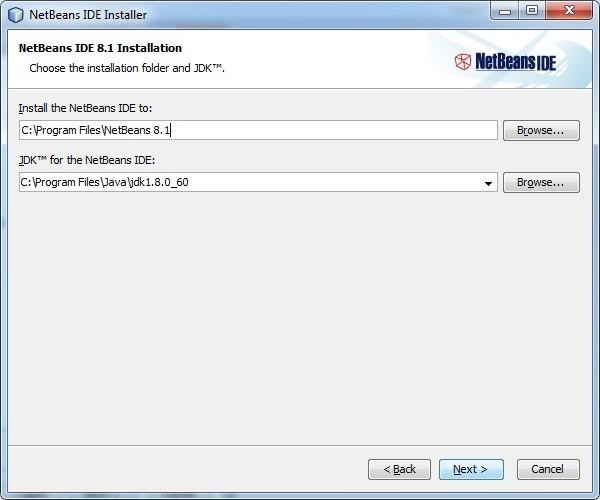
**Gambar 6.** Wellcome NetBeans IDE Installer

* + 1. Maka akan muncul tampilan License Agreement berikut ini



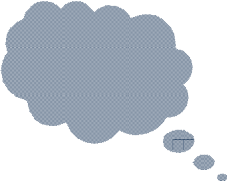
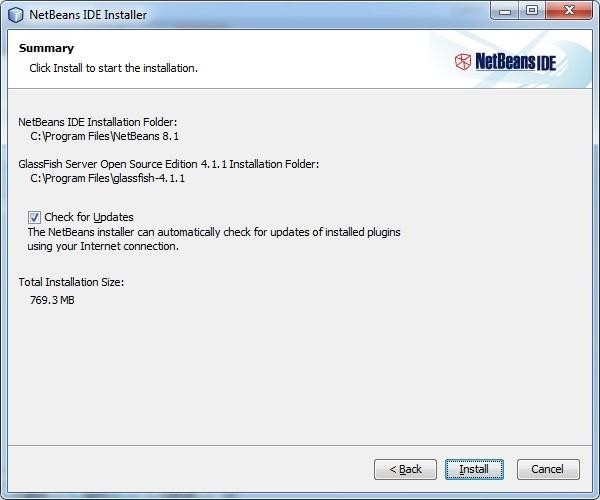
Klik

**Gambar 7.** License Agreement



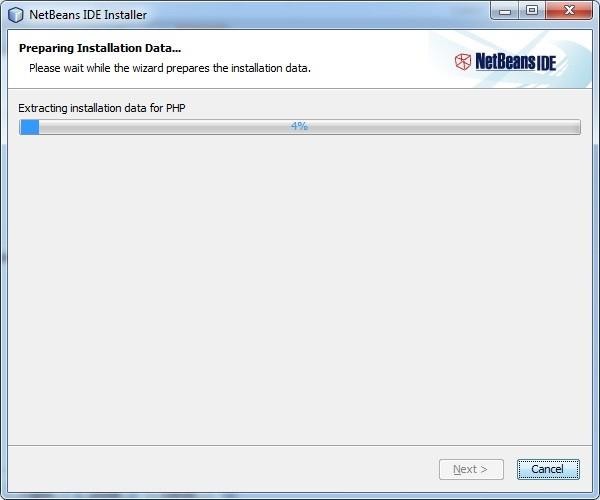
Klik

**Gambar 8.** Installasi Folder

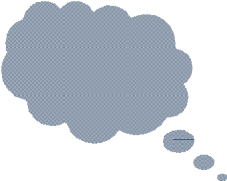
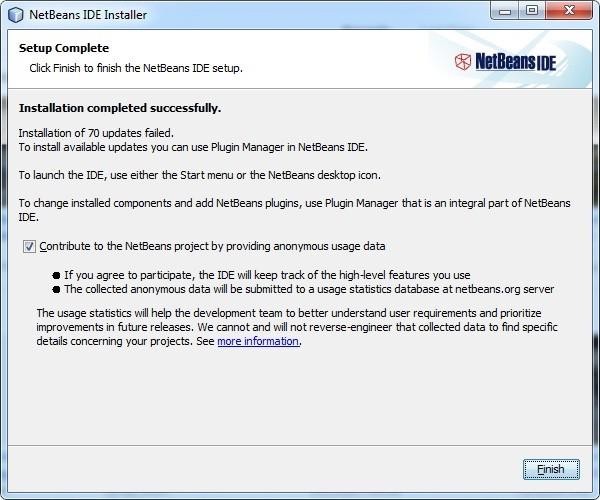


Klik

**Gambar 9.** Start the Installation

* + 1. Proses installasi berjalan seperti gambar dibawah ini.

**Gambar 10.** Proses Installasi

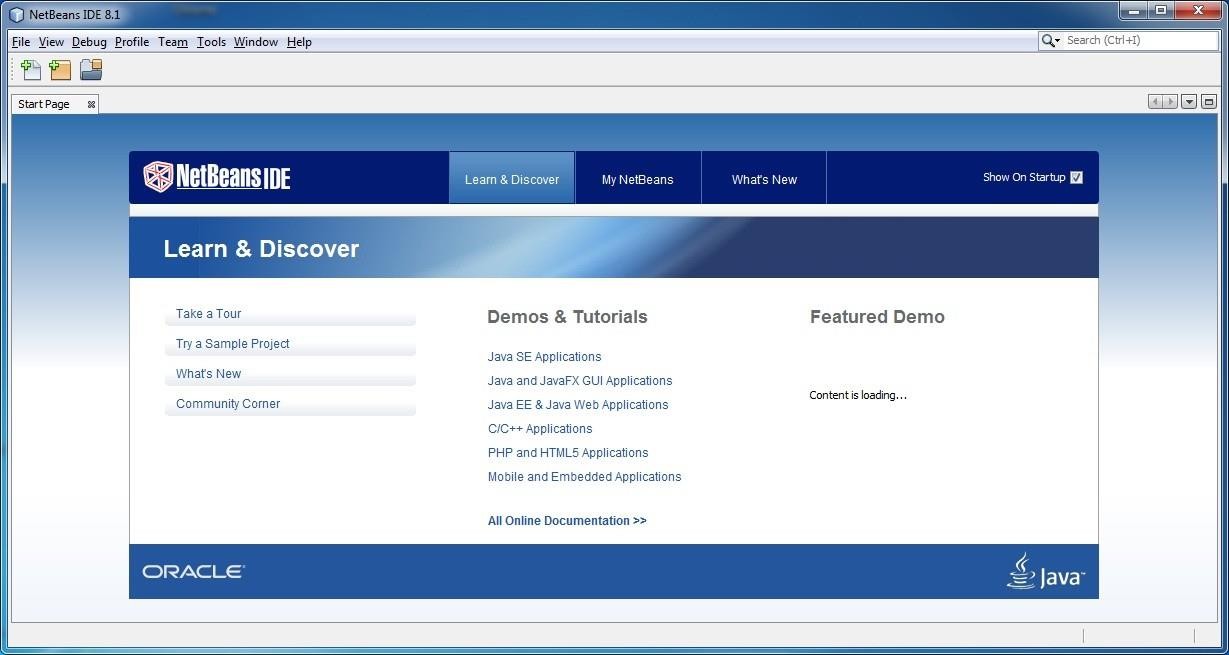


Klik

**Gambar 10.** Proses Installasi Selesai

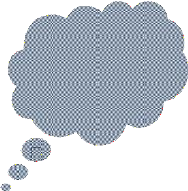
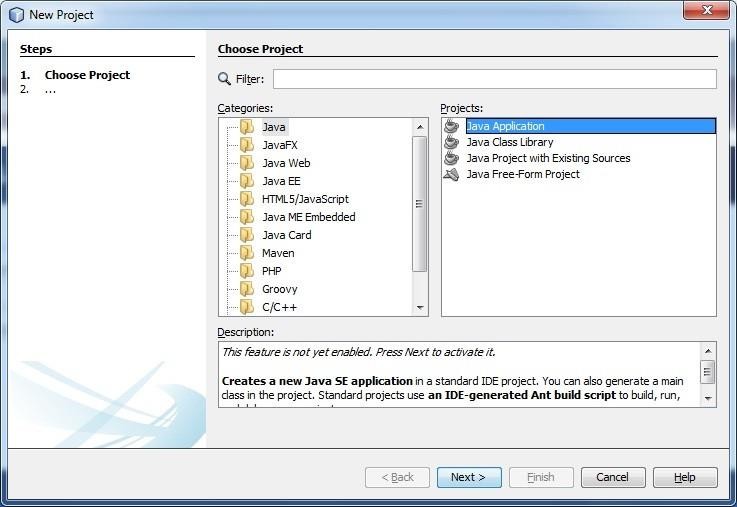
* 1. **Membuat Project Baru**

**Langkah-langkah untuk membuat Project Baru :**

* + 1. Dari Start Program pilih Netbeans IDE 8.1

**Gambar 11.** Tampilan Awal NetBeans IDE

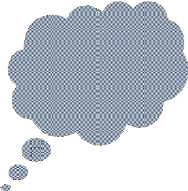
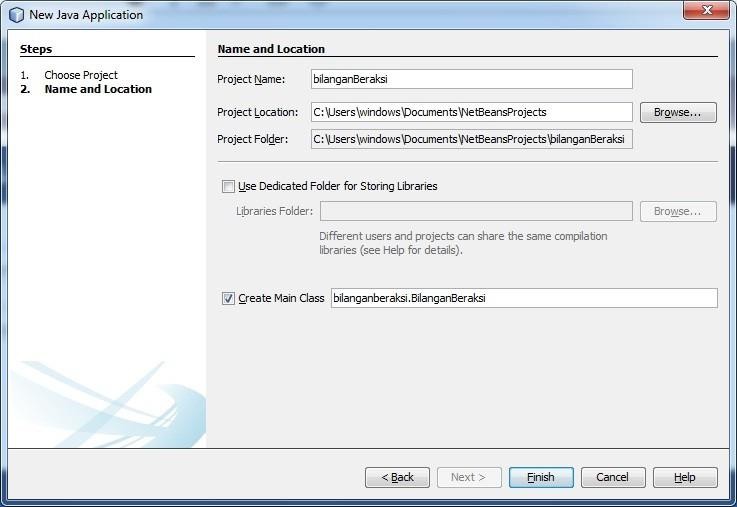
* + 1. Pilih File  New Project Java Java Application



Klik

**Gambar 12.** New Project

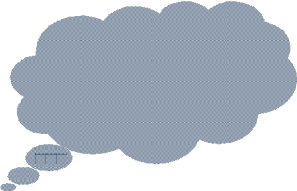
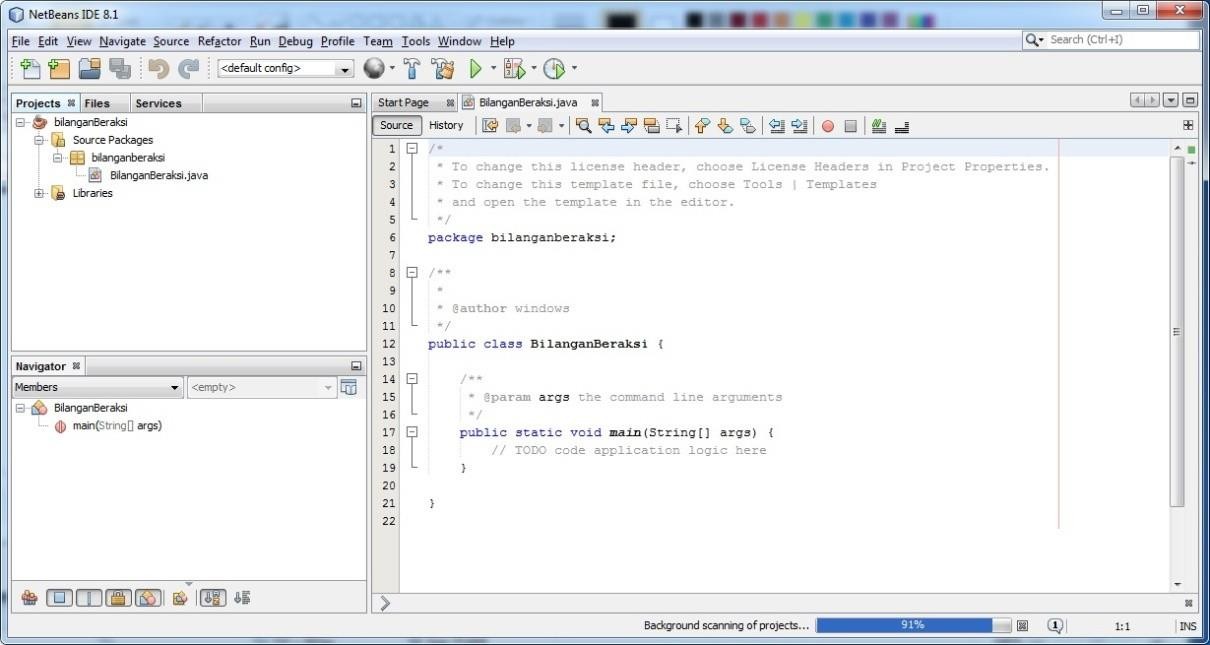
* + 1. Maka akan muncul Form dibawah ini untuk penamaan project dan lokasi penyimpanannya



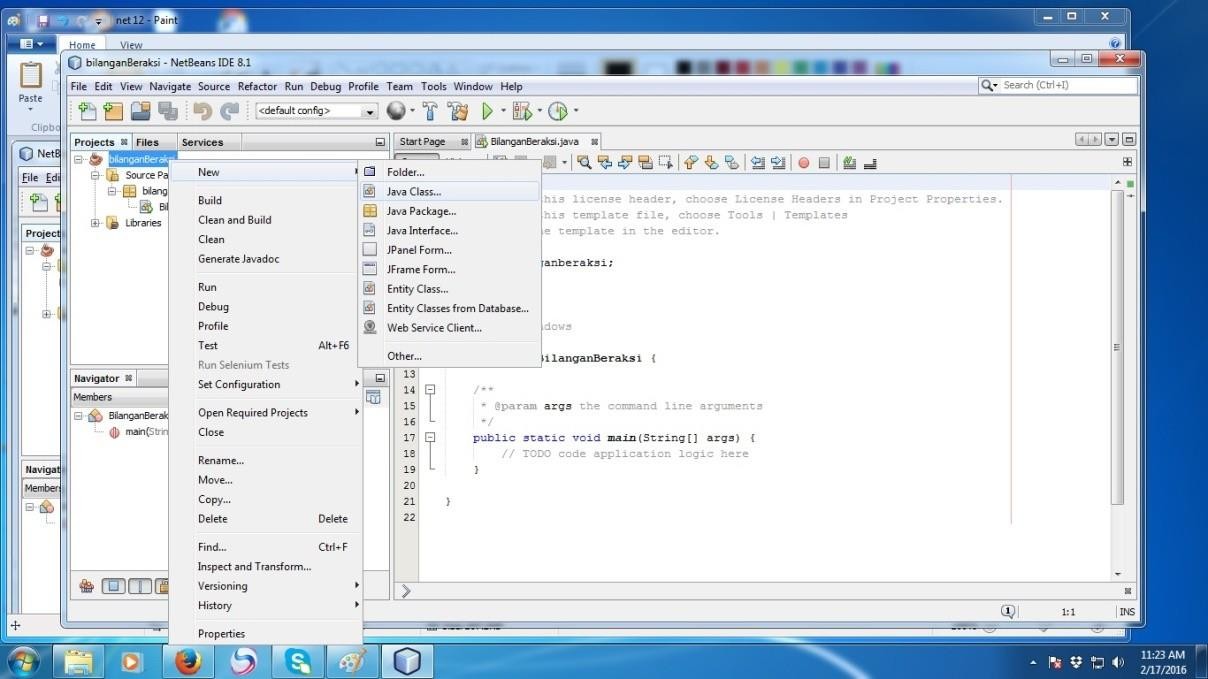
Klik

**Gambar 13.** Name and location project

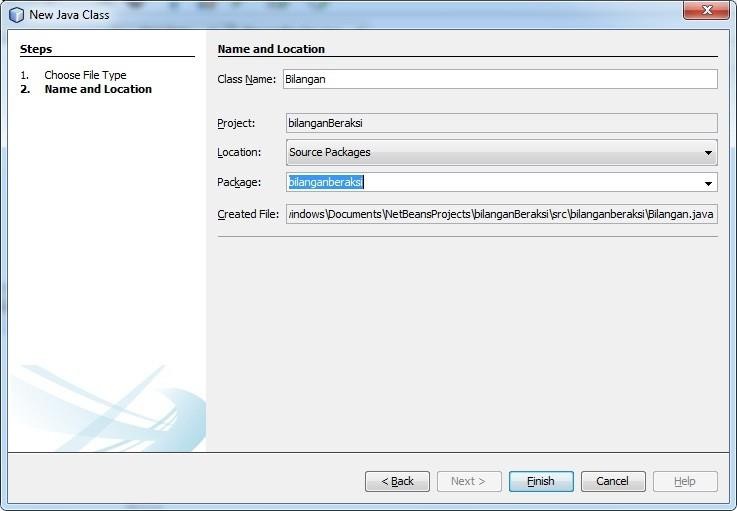
* + 1. Lembar kerja NetBeans untuk menuliskan Source Code



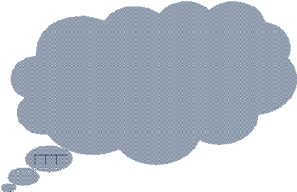
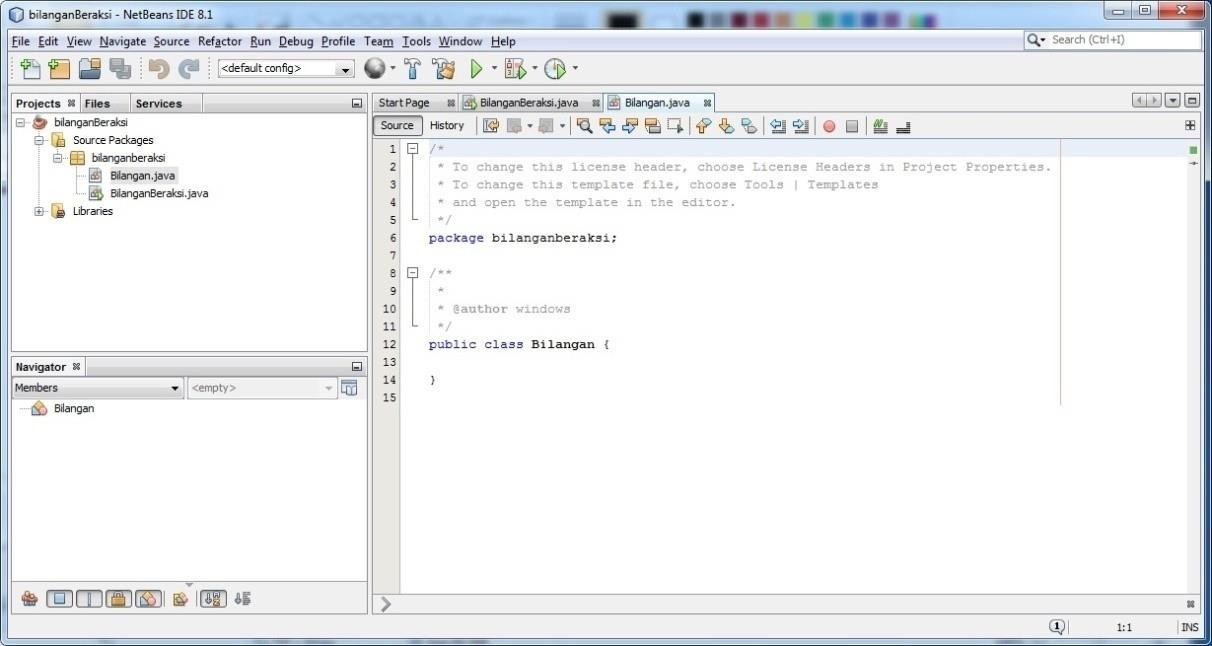
**Gambar 14.** Lembar kerja NetBeans

* + 1. Menambahkan kelas pada sebuah project dengan Langkah-langkah sebagai berikut.

**Gambar 15.** Membuat class



**Gambar 16.** Class name



**Gambar 16.** Lembar kerja class

Sumber referensi :

- Modul 1

**Ⅳ. GUIDED**

Membandingkan Program Terstruktur dan Program Berorientasi Objek

**c. Pemograman Berorientasi Objek (Bahasa Java)**

1. Buat Project dengan nama DemoApp
2. Buat Class dalam Poject tersebut dengan nama DemoObjectOriented
3. Dalam file DemoObjectOriented. Java tuliskan code berikut ini

import java.io.\*;

class DemoObjectOriented

{

public void helloWorld(int jmlh\_looping)

{

System.out.print ("\n-Display Hello World-\n");for(int i=0;i< jmlh\_looping;i++)

{

System.out.print (+(i+1)+". HelloWorld.

.................\n");

}

}

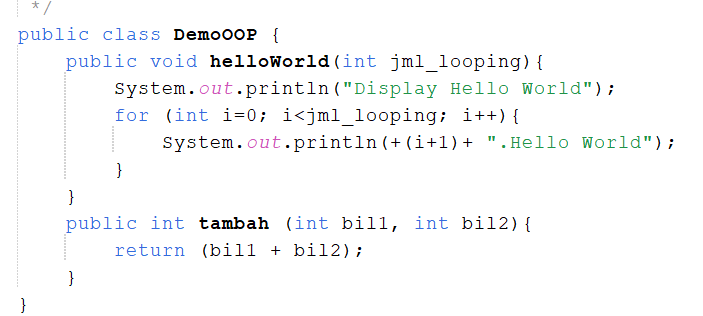
public int tambah(int bil1, int bil2)

{

return(bil1+bil2);

}

}

Program :

Penjelasan : diatas adalah perintah untuk menampilkan output hello word

4. Dalam file class Main.java tuliskan code berikut ini

public static void main(String args[])

{

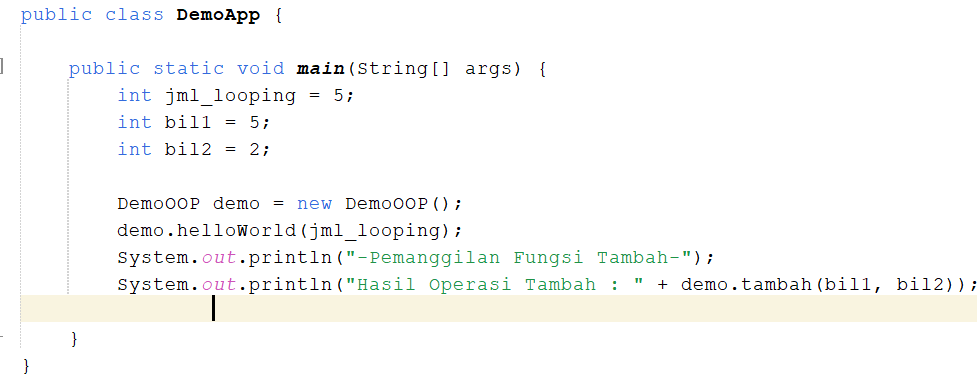
int jumlah\_looping=5; int bil1=5;

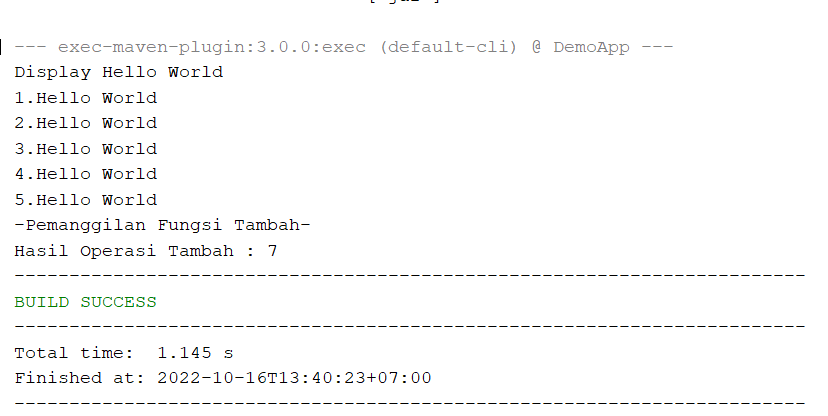
int bil2=2;

DemoObjectOriented Demo = new DemoObjectOriented (); Demo.helloWorld(jumlah\_looping);

System.out.print("\n-Pemanggilan Fungsi Tambah-\n"); System.out.print(" Hasil Operasi Tambah : "+Demo.tambah(bil1,bil2)+"\n");

}

Program :

hasil :

Penjelasan : diatas merupakan program perulangan Hello World menggunakan perintah for

# GUIDED 2

1. Membuat kelas untuk menampilkan kalimat “ Selamat Datang di Praktikum PBO “ di dalam java

Welcome.java

public class Welcome

{

public void display()

{

System.out.println("Selamat Datang di Praktikum PBO");

}

}

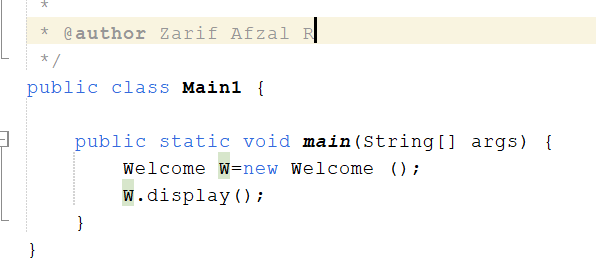
Main.java

public static void main(String []args)

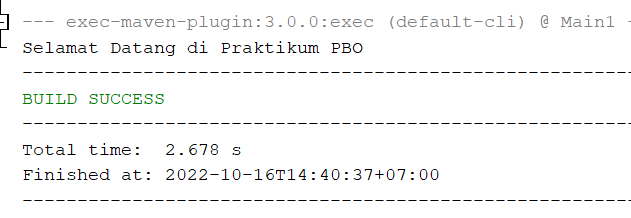
{

Welcome W=new Welcome (); W.display();

}

Program :

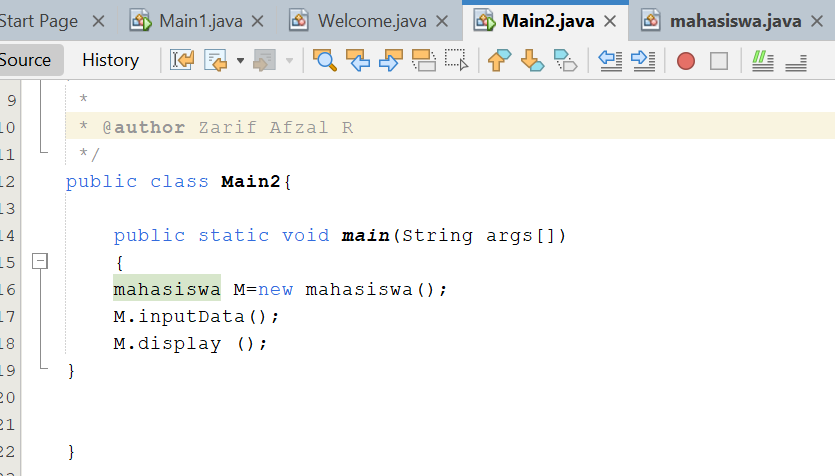
Penjelasan : saat kita nge-run Main1, maka yang tampil adalah perintah di Welcome. Karena perintah di welcome sudah dipanggil

Hasil :

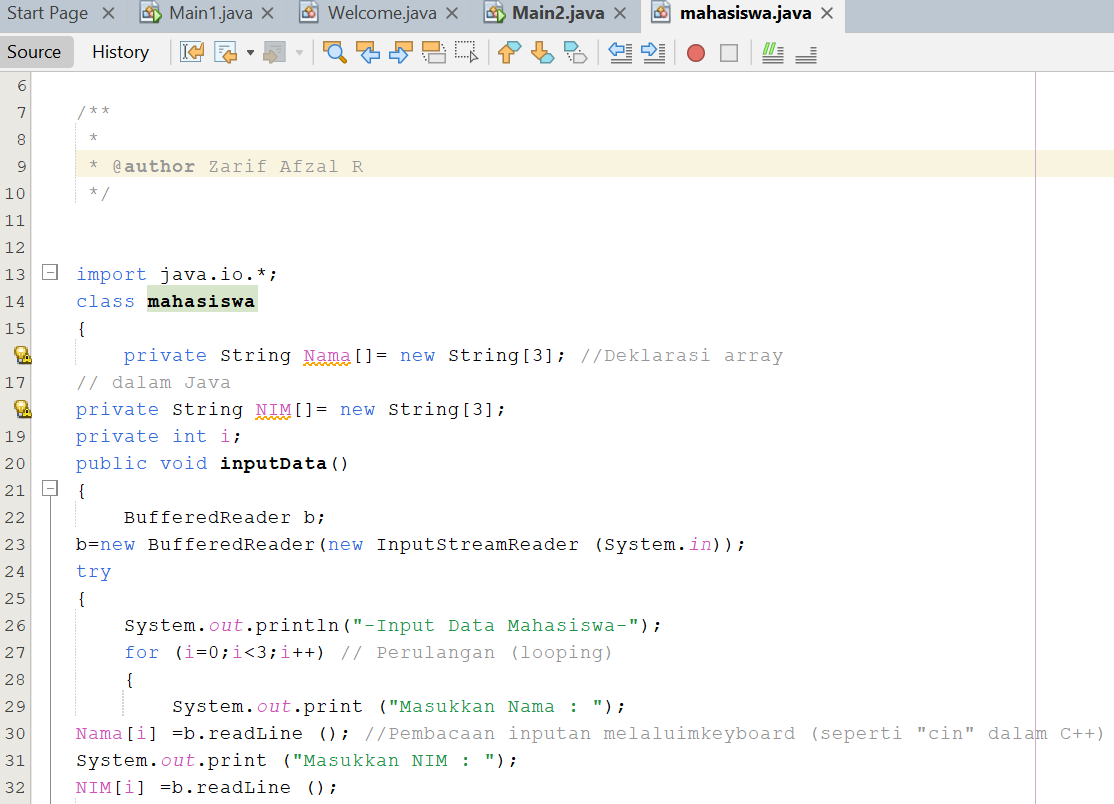
Penjelasan : Tulisan yang tampil diatas adalah tulisan yang ditulis pada file welcome

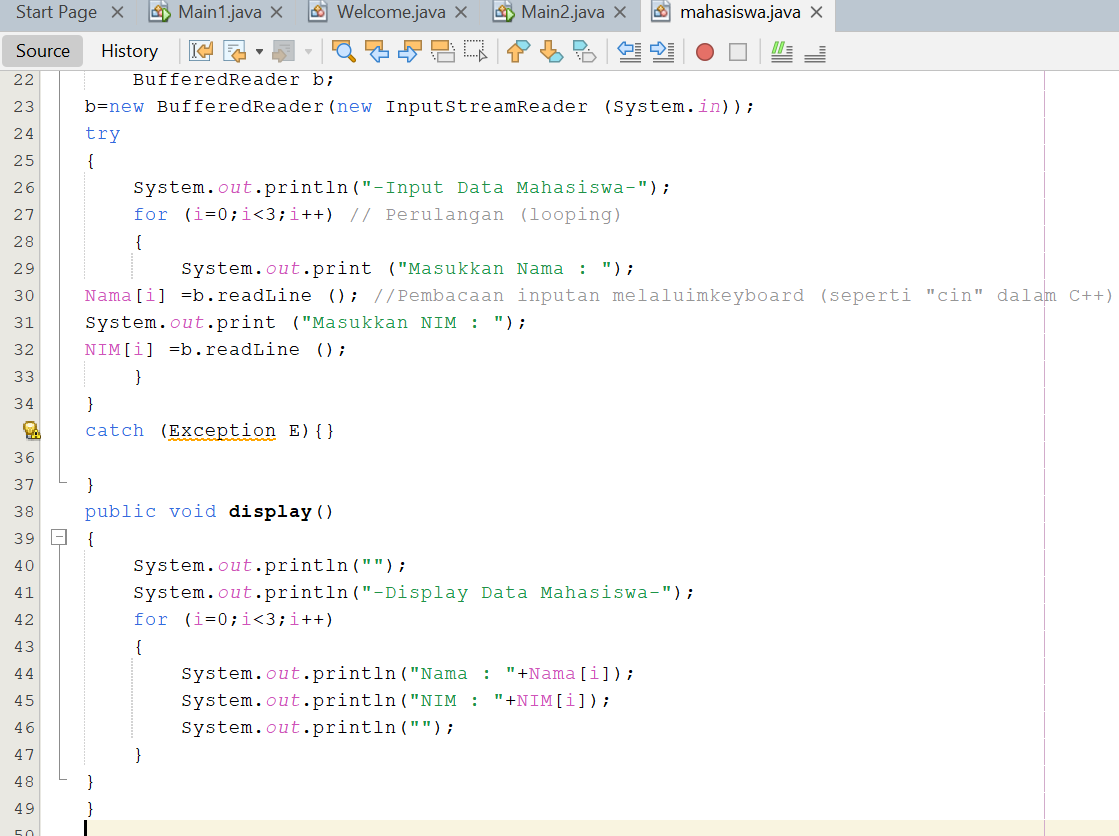
1. Membuat fungsi Input dan Output beserta Looping (perulangan) dalam java

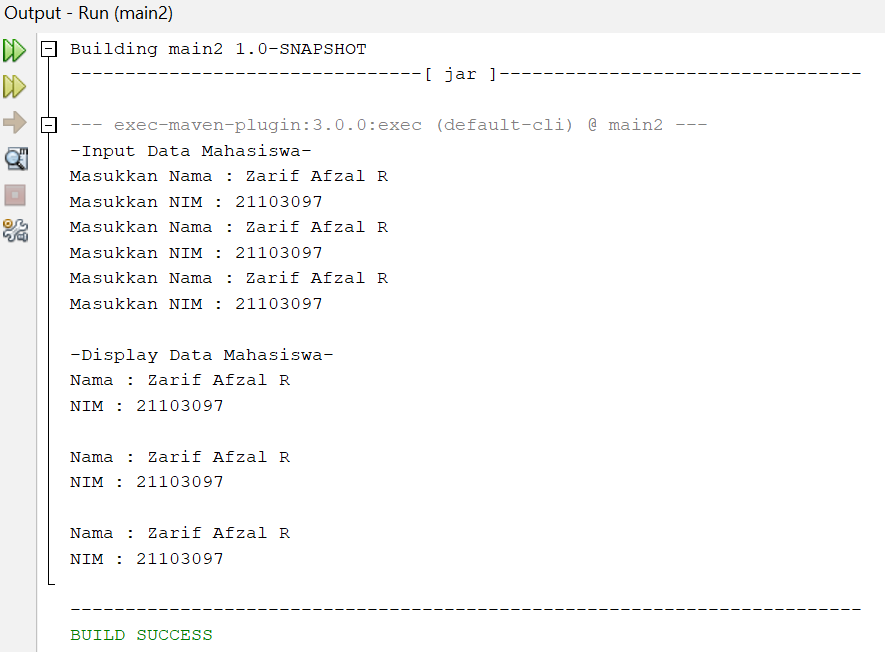
|  |
| --- |
| Mahasiswa. Java  import java.io.\*; //Pemanggilan Library dalam Java class Mahasiswa  {  private String Nama[]= new String[3]; //Deklarasi array  // dalam Java  private String NIM[]= new String[3]; private int i;  public void inputData()  {  BufferedReader b;  b=new BufferedReader(new InputStreamReader (System.in));  try //Penjelasan Exception lebih lanjut di berikutnya  {  System.out.println("-Input Data Mahasiswa-"); for (i=0;i<3;i++) // Perulangan (looping)  {  System.out.print ("Masukkan Nama : ");  Nama[i] =b.readLine (); //Pembacaan inputan melaluimkeyboard (seperti "cin" dalam C++)  System.out.print ("Masukkan NIM : "); NIM[i] =b.readLine ();  }  }  catch (Exception E){}  }  public void display()  {  System.out.println("");  System.out.println("-Display Data Mahasiswa-"); for (i=0;i<3;i++)  {  System.out.println("Nama : "+Nama[i]); System.out.println("NIM : "+NIM[i]); System.out.println("");  }  }  }  Main.java  public static void main(String args[])  {  Mahasiswa M=new Mahasiswa(); //Pembentukan Objek M.inputData(); // Pemanggilan method  M.display ();  } |

Program

Penjelasan : program yang ditulis di file mahasiswa akan dipanggil di main, dan nanti ketika program akan di run, maka yang akan di run adalah file main

Program :

penjelasan : gambar diatas adalah gambar dari file mahasiswa, yang berisi banyak perintah untuk menghasilkan output yang berupa perulangan.

Hasil :

### V. KESIMPULAN

Dari modul 1, kita dapat mempelajari penginstallan java, memahami karakteristik dari penulisan java, dan melakukan perulangan pada output.