laporan praktikum

Pemrogramam berorientasi objek

Modul : 2 Tanggal praktikum : 29-03-2023 *(Format: DD-MM-YYYY)*

Nama : Muhamad Fahmi

NIM : 20220810029

Kelas : TINFC-2022-02 *(contoh: TINFC-2022-01)*

# jUDUL/TEMA MODUL

## Tujuan Pembelajaran

|  |
| --- |
| Tuliskan tujuan pembelajaran/praktikum. Sesuaikan dengan tujuan pembelajaran pada modul praktikum. |

* 1. Memahami konsep OOP
  2. Memahami konsep Objek, Class, Atribut, Method, This, Access Modifier, Instance of Class.

## Dasar Teori

|  |
| --- |
| Pada bagian ini, silahkan kalian tulis/tambahkan materi pendukung untuk kegiatan praktikumnya. Tambahkan gambar / tabel/ flowchart apabila diperlukan. |

1. **Objek**

Kesatuan entitas (benda), baik baik yang berwujud nyata ataupun hanya suatu system/konsep yang memiliki sifat, karakteristik dan fungsi.

# Class

Class merupakan cetak biru (blue print) dari objek atau dengan kata lain sebuah Class menggambarkan ciri-ciri objek secara umum. Sebagai contoh Suzuki Smash, Yamaha VegaR, Honda SupraFit, dan Kawasaki KazeR merupakan objek dari Class sepeda motor. Suzuki Smash dan objek lainnya juga memiliki kesamaan atribut (merk, tipe, berat, kapasitas bensin, tipe mesin, warna, harga) dan method untuk mengakses data pada atributnya (misal fungsi untuk menginputkan data merk, tipe, berat, dsb serta fungsi untuk mencetak data merk, tipe, berat, dsb).

Contoh :

class SepedaMotor

{ private String merk, tipe; private int tangki; private long harga;

public void setMerk(String merk) { this.merk = merk;

} public String getMerk(){ return merk;

}

}

# Atribut

Attribut merupakan nilai data yang terdapat pada suatu object yang berasal dari class. Attributes merepresentasikan karakteristik dari suatu object.

# Method

Apa saja yang dapat dilakukan / dialami oleh suatu objek, jika bersifat static berarti dilakukan oleh semua objek dalam kelas tersebut.

Sintaks untuk membuat method :

Parameter å tipeData1 namaVar1

accessModifier tipeMethod namaMethod(………)

Contoh :

public long getHarga()

# This

This digunakan untuk membedakan variabel yang dideklarasikan pada parameter di dalam method dengan variabel yang dideklarasikan pada class.

Perhatikan keyword “***this***” di bawah ini(lihat pada class SepedaMotor) :

public void setMerk(String merk) { this.merk

= merk;

}

Untuk membedakan variable merk pada parameter dan variabel merk pada atribut dari class SepedaMotor, digunakanlah keyword “*this*”. Sehingga untuk menggunakan atribut merk pada class SepedaMotor, digunakan :

this.merk

# Access Modifier

Yang dimaksud dengan *access modifier* adalah pengaturan hak akses class maupun method. Ada 4 akses yang tersedia, yaitu default, public, protected, private Untuk lebih jelasnya, silahkan lihat kedua tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| No | Modifier | Pada class dan Interface | Pada Method & Variabel |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | *default* (tak ada modifier) | Dapat diakses oleh yang sepaket. | Diwarisi oleh subkelas dipaket yang sama, dapat diakses oleh method- method  yang sepaket. |
|  |  |  |  |
| 2 | Public | Dapat diakses dimanapun | Diwarisi oleh subkelasnya,  dapat diakses dimanapun. |
| 3 | Protected | Tidak bias diterpakan | Diwarisi oleh subkelasnya, dapat diakses oleh method-  method yang sepaket |
| 4 | Private | Tidak bisa diterapkan | Tidak dapat diakses dimanapun kecuali oleh method-method yang ada  dalam kelas itu sendiri. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aksesabilitas** | **public** | **private** | **protected** | **default** |
| Dari kelas yang sama | Ya | Ya | Ya | Ya |
| Dari sembarang kelas dalam paket yang sama | Ya | Tidak | Ya | Ya |
| Dari sembarang kelas di luar paket | Ya | Tidak | Tidak | Tidak |
| Dari subkelas dalam paket yang sama | Ya | Tidak | Ya | Ya |
| Dari subkelas di luar paket | Ya | Tidak | Ya | Tidak |

# Instance of Class

Objek yang di instan atau dibuat dari class.

# Input Data

Dalam pemrograman java, untuk menginputkan data dari keyboard ada 2 cara, yaitu :

* + 1. Input dari mode console, yaitu dengan memanfaatkan class

# BufferedReader dan InputStreamReader.

Untuk bisa mengakses class BufferedReader maka perlu mengimpor dari package **java.io.\*** dan menambahkan statemen **throws IOException** pada header method main.

* + 1. Inputan dengan memanfaatkan class **JOptionPane**.

Untuk bisa menggunakan class JoptionPane, maka perlu mengimpor dari package **javax.swing.\*** dan gunakan method **showInputDialog()** yang terdapat pada class JOptionPane.

# Java Documentation

Merupakan dokumentasi dari kelas-kelas yang sudah disediakan oleh java yang berupa API (*Application Programming Interface*), dan siap untuk digunakan.

# CLASS KENDARAAN



**Atribut :**

Merk Warna

**Method :**

Maju Mundur

*OBJEK MOBIL*

## Hasil dan Pembahasan

|  |
| --- |
| * Bagian ini berisi hasil praktikum yang telah anda kerjakan yang meliputi *pretest*, tahapan praktikum, tugas, dan *posttest*. * Lampirkan hasil pekerjaan berupa tangkapan layar *(schreenshoot)* kegiatan praktikum sesuai dengan langkah kerja yang ada. |

**PreTest**

Class barang

Method

Menulis

Membaca

menggambar

atribut

merek

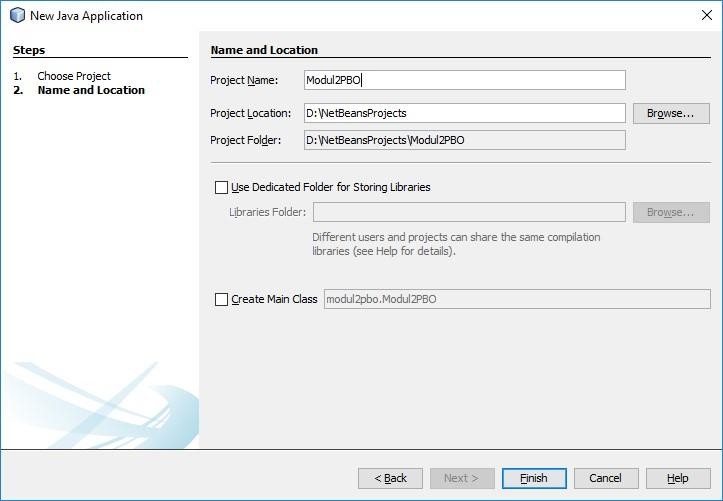
warna

ukuran

Objek buku

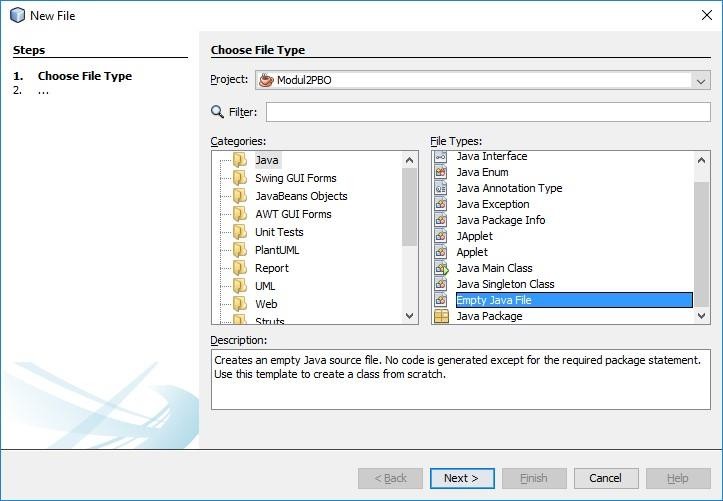
**PRAKTIKUM PRAKTIKUM1**

* 1. Buatlah Project Baru dengan Netbeans
  2. File  New Project Choose Project (Java Aplication)
  3. Named and Location (Tentukan nama file project dan lokasi penyimpanan yang akan dibuat)

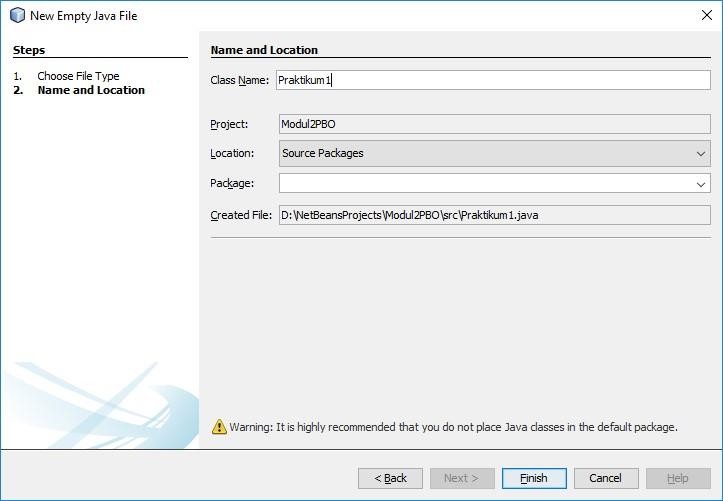


**Membuat File Java pada Project (Empty Java File)**

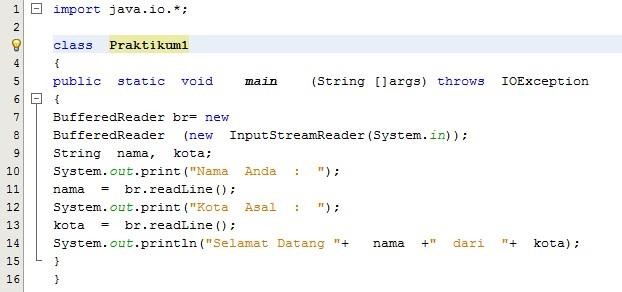
* 1. Buat File Empty Java File Baru (Klik kanan Source NewOther
  2. Pilih Java  Empty Java File



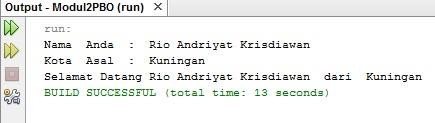
* 1. Berinama File Class Java yang akan dibuat serta tentukan lokasi penyimpanannya.



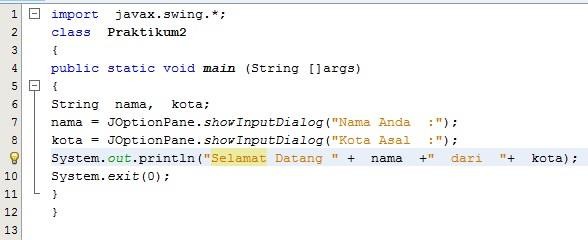
**Menuliskan Kode Program Java Java Input Data**

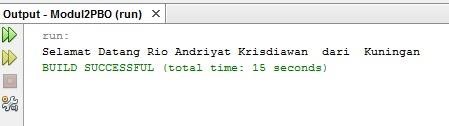
* 1. Ketikan Source Code berikut ini pada file Praktikum1.java
  2. Simpan File tersebut.

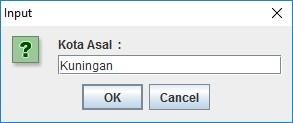
**Menjalankan Program Java**

1. Klik Kanan Pada file Praktikum1.java  pilih Run File
2. Masukan Nama, Kota asal, lalu Amati Hasilnya

**PRAKTIKUM2**

1. Buatlah file Empty Java File java pada project tersebut dengan nama Praktikum2.java
2. Ketikan kode program berikut :
3. Jalankan program tersebut, Amati Hasilnya!

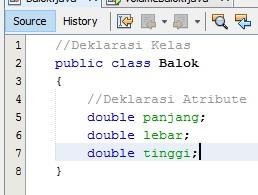




**PRAKTIKUM3**

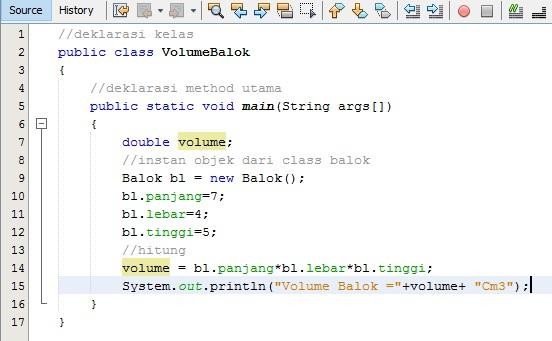
**Membuat objek pada sebuah class**

1. Buatlah file Empty Java File java pada project tersebut dengan nama **Balok.java**
2. Ketikan kode program berikut :



1. Buatlah file Empty Java File java pada project tersebut dengan nama

**VolumeBalok.java**

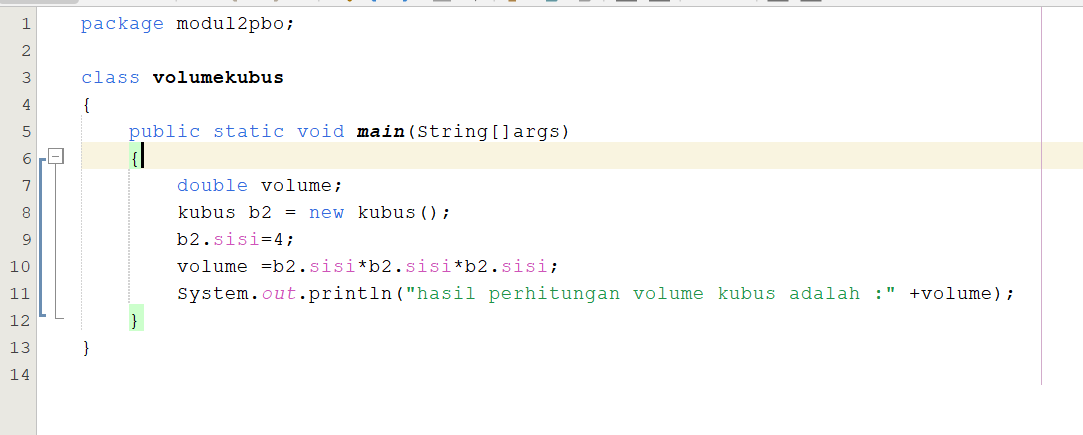
1. Ketikan kode program berikut :
2. Jalankan File **VolumeBalok.java** tersebut, Amati Hasilnya!

**LATIHAN / POST TEST**

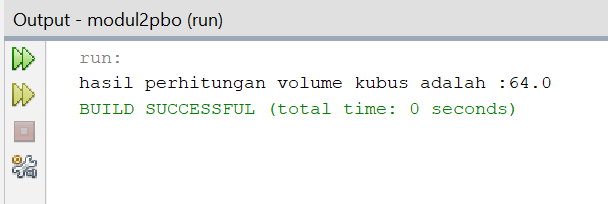
1. Buatlah Program penerapan class, object, atribut, method seperti praktikum 3 dengan class/objek yang lain.
2. Modifikasi Praktikum 3, sehingga niali balok untuk panjang, lebar dan tinggi di inputkan seperti Praktikum1.

Jawab :

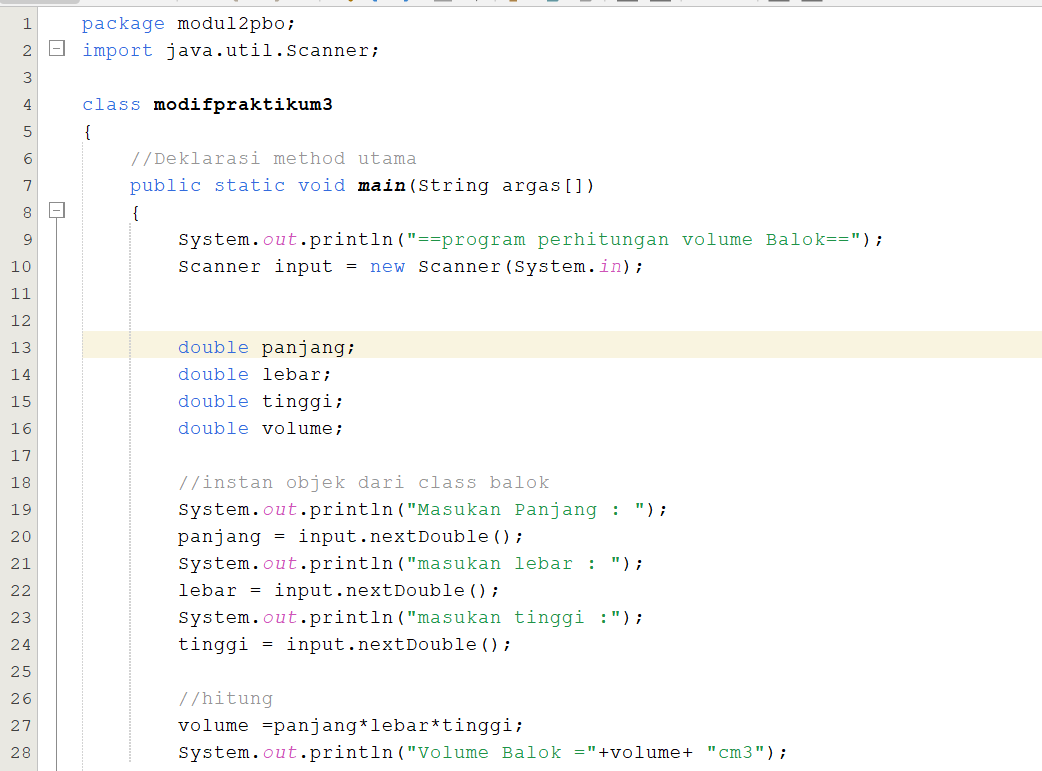
1. Source Code

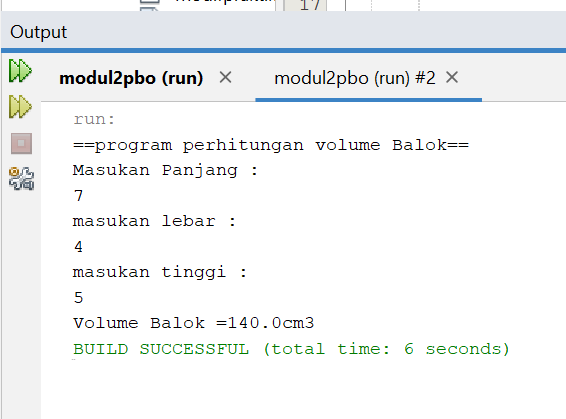


Hasil Output



1. Source Code

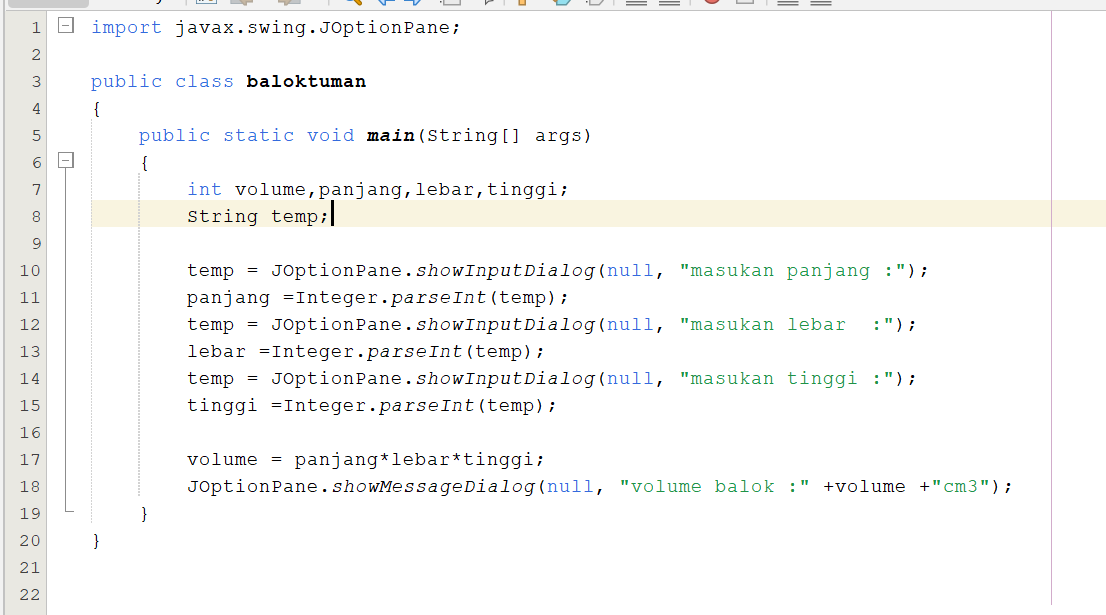


 Hasil Output

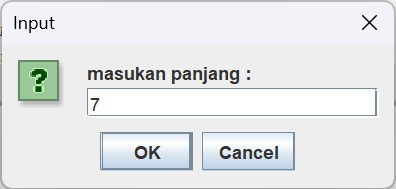
**TUGAS INDIVIDU UNTUK MINGGU DEPAN**

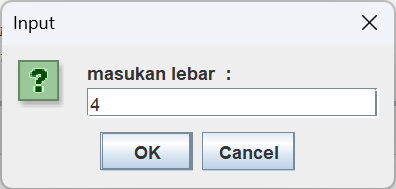
1. Buatlah Program Praktikum 3 dengan inputan data menggunakan JOptionPane.

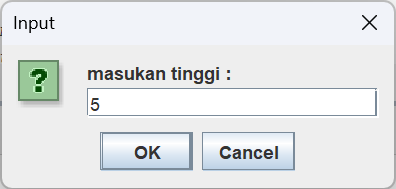
Source Code

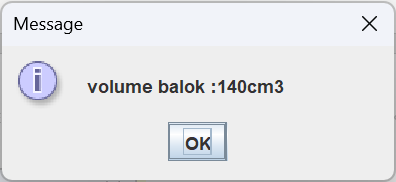


Hasil Output









## Kesimpulan

|  |
| --- |
| Buatlah kesimpulan dari pembahasan praktikum yang telah Anda kerjakan. |

## Pada modul 2 kali ini kita mempelajari object dan class yang di sertai oleh atribut dan method

mempelajari berbagai macam jenis inputan mulai dari bufferedreader , JOptionPane

## Referensi

|  |
| --- |
| Pada bagian ini cantumkan daftar pustaka/referensi yang ada gunakan dalam menyusun laporan praktikum ini. |

## Modul pembelajaran

Gambar di kutif : https://id.pinterest.com/