

MODUL PRAKTIKUM
OBJECT ORIENTED PROGRAMMING
BAHASA PEMROGRAMAN JAVA

DOSEN PENGAMPU
NURUL KHAIRINA

UNIVERSITAS MEDAN AREA
MEDAN
2020

KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah SWT. Atas rahmat dan karunianya, Modul Praktikum *Object Oriented Programming* Bahasa Pemrograman Java ini dapat diselesaikan dengan baik. Modul ini merupakan pengembangan dari modul yang telah dibuat sebelumnya pada modul praktikum Bahasa pemrograman Java di salah satu PTS di Kota Medan. Modul ini terdiri dari 10 pertemuan yaitu : Memahami Dasar-Dasar Pemrograman Java, Memberikan Input dari Keyboard, Struktur Pengendali Proses, Java Array, Membuat Class Sendiri, Abstraksi (*Abstraction*) & Pembungkusan (*Encapsulation*), Pewarisan (*Inheritance*) & Polimorfisme (*Polymorphism*), *Exception Handling*, *Graphical User Interface* (GUI), dan Aplikasi Java CRUD dengan Database MySQL. Modul ini diharapkan dapat membantu mahasiswa dalam memahami praktikum *object oriented programming* (OOP) yang akan dibangun dengan bahasa pemrograman java dengan *software* Netbeans 8.2. Kritik dan saran yang membangun akan sangat diharapkan untuk perkembangan modul praktikum ini.

Medan, 6 April 2020

Nurul Khairina

MODUL PRAKTIKUM

Mata Kuliah Praktikum : **Praktikum Pemrograman Berorientasi Objek**
Kode Mata Kuliah Praktikum : **TIF16025**
SKS : **1 SKS**
Program Studi : **Teknik Informatika**
Semester : **IV**

DISETUJUI OLEH	DIPERIKSA OLEH	DIBUAT OLEH
Ka. Prodi	Ka. Lab	Dosen Pengampu
Rizki Muliono, S.Kom, M.Kom	Muhathir, S.Kom.,M.Kom	Nurul Khairina, S.Kom, M.Kom

DAFTAR ISI

Kata Pengantar	1
Daftar Isi	3
Modul 1 : Memahami Dasar-Dasar Pemrograman Java	4
Modul 2 : Memberikan Input dari Keyboard	11
Modul 3 : Struktur Pengendali Proses	15
Modul 4 : Java Array	22
Modul 5 : Membuat Class Sendiri	27
Modul 6 : Abstraksi (<i>Abstraction</i>) & Pembungkusan (<i>Encapsulation</i>)	35
Modul 7 : Pewarisan (<i>Inheritance</i>) & Polimorfisme (<i>Polymorphism</i>)	40
Modul 8 : <i>Exception Handling</i>	45
Modul 9 : <i>Graphical User Interface</i> (GUI)	52
Modul 10 : Aplikasi Java CRUD dengan Database MySQL	59
Daftar Pustaka	67
Lampiran	68

MODUL 1

MEMAHAMI DASAR - DASAR PEMROGRAMAN JAVA

A. TUJUAN

Setelah praktikum ini, praktikan diharapkan dapat mengenal Java melalui *Class*, *Object*, *Method*, *Constructor*.

B. PERALATAN DAN BAHAN

1. Personal Komputer
2. Perangkat Lunak Netbeans

C. TEORI

Pemrograman Berorientasi Objek (PBO) atau *Object Oriented Programming* (OOP) menggunakan Bahasa Pemrograman Java. Bahasa Pemrograman Java memiliki beberapa komponen, antara lain :

- *Class* : tempat untuk mendeklarasikan tipe data
- *Object* : entiti yang memiliki keadaan/ tingkah laku
- *Attribute* : elemen dari sebuah objek yang berisi informasi tentang objek
- *Method* : tingkah laku dari objek/ sub program
- *Construction* : method yang digunakan untuk membuat objek baru

Bahasa pemrograman java dikenal dengan *case sensitive*, artinya ada banyak aturan dalam Pemrograman Java yang harus benar-benar diperhatikan. Berikut ini beberapa aturan dalam pemberian nama *class* :

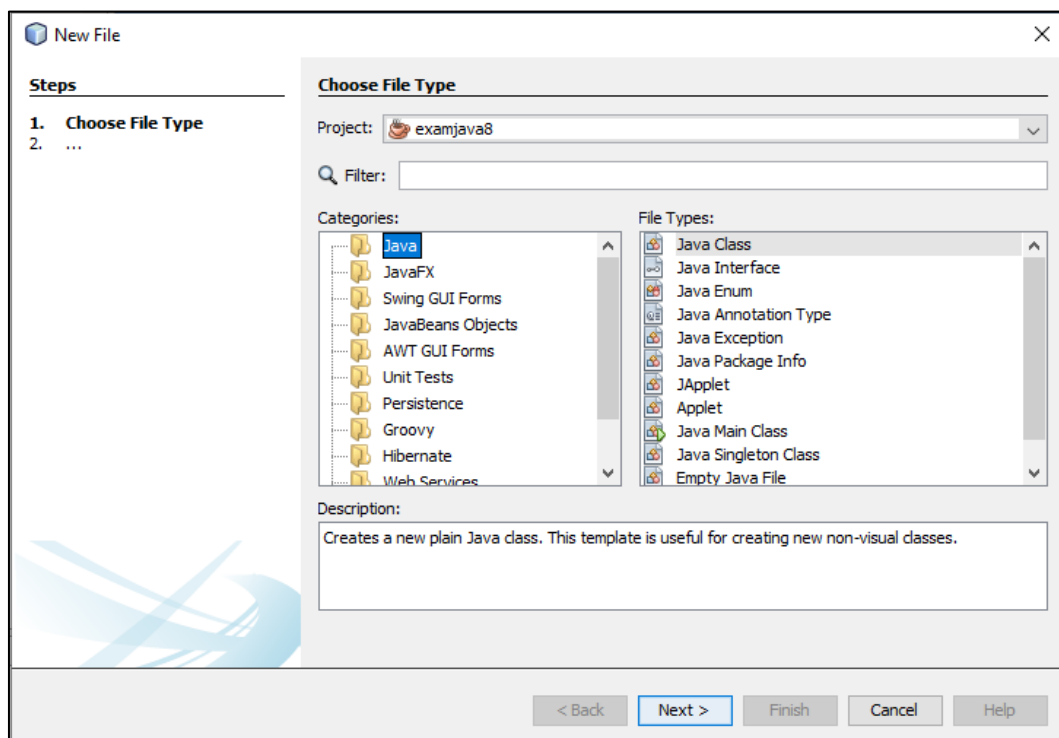
- a. Nama file **halaman kerja java** harus sama dengan **nama class**
- b. Nama class boleh terdiri dari huruf kecil dan huruf besar
- c. Nama class yang terdiri dari dua suku kata, dapat dituliskan dengan tanda underscore (_) sebagai pemisah. Contoh : Belajar_Java
- d. Nama class tidak boleh terdiri dari angka
- e. Nama class boleh ditulis dengan huruf abjad dan diikuti dengan angka.
Contoh : Belajar_Java1

- f. Nama class tidak boleh menggunakan keyword (*public*, *class*, *private*, dan sebagainya)

D. PRAKTIKUM

1. Memulai Aplikasi Netbeans

Praktikum Pemrograman Berorientasi Objek (PBO) ini akan menggunakan software Java NetBeans. Halaman baru pada Java NetBeans dapat dimulai dari menu **New – New File – Java Class**. Kemudian berilah nama project, nama class, dan nama package. Nama package dapat diisi sesuai dengan nama project.



New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name:

Project:

Location:

Package:

Created File:

 The file NewClass.java already exists.

< Back Next > Finish Cancel Help

New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name:

Project:

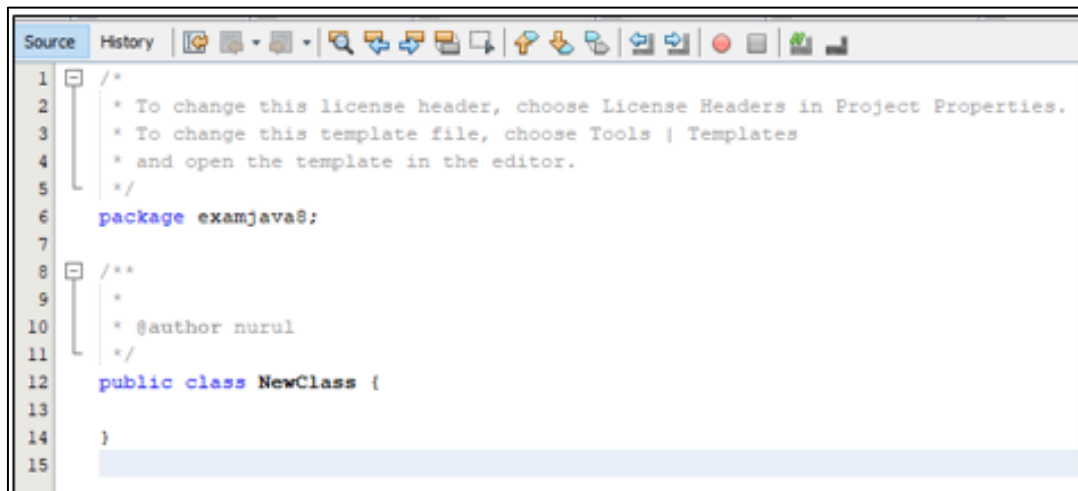
Location:

Package:

Created File:

< Back Next > **Finish** Cancel Help

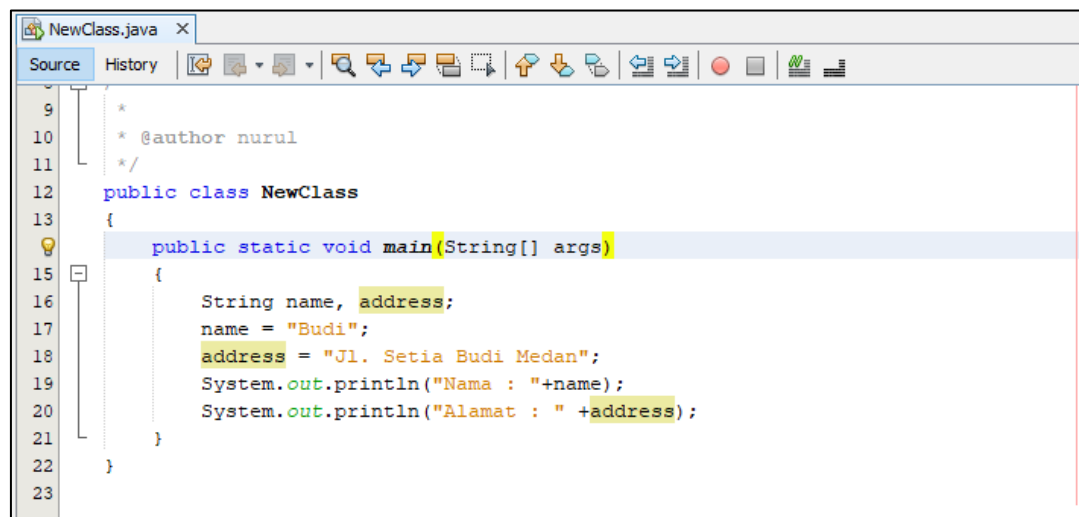
Tampilan awal halaman kerja java dengan NewClass :



```
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6   package examjava8;
7
8   /**
9    *
10   * @author nurul
11   */
12   public class NewClass {
13
14   }
15
```

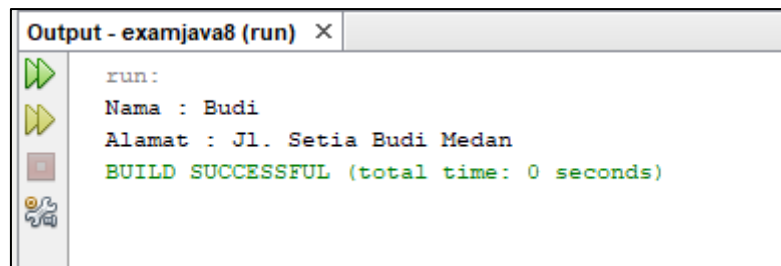
2. Membuat Class

❖ Code Pemrograman:



```
NewClass.java X
Source History
9   *
10  * @author nurul
11  */
12  public class NewClass
13  {
14      public static void main(String[] args)
15      {
16          String name, address;
17          name = "Budi";
18          address = "Jl. Setia Budi Medan";
19          System.out.println("Nama : "+name);
20          System.out.println("Alamat : " +address);
21      }
22  }
23
```

❖ Hasil Pemrograman :



```
Output - examjava8 (run) X
run:
Nama : Budi
Alamat : Jl. Setia Budi Medan
BUILD SUCCESSFUL (total time: 0 seconds)
```

3. Membuat *Object*

❖ Code Pemrograman :

```
Source History
12
13 class variabel
14 {
15     int lebar;
16     int panjang;
17 }
18
19 class NewClass2
20 {
21     public static void main(String[] args)
22     {
23         double luas;
24         variabel p = new variabel();
25         p.panjang = 20;
26         p.lebar = 10;
27         luas = p.panjang * p.lebar;
28         System.out.println("Luas Persegi : " +luas);
29     }
30 }
```

❖ Hasil Pemrograman :

```
Output - examjava8 (run) X
run:
Luas Persegi : 200.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

4. Membuat *Method*

❖ Code :

```
Source History
10 * @author nurul
11 */
12
13 class kali
14 {
15     double a;
16     double b;
17
18     void hasil()
19     {
20         System.out.println("Hasil perkalian = " + (a*b));
21     }
22 }
```

```
23
24 public class HasilPerkalian
25 {
26     public static void main(String[] args)
27     {
28         kali variabel1, variabel2;
29
30         variabel1 = new kali();
31         variabel2 = new kali();
32
33         variabel1.a = 3;
34         variabel1.b = 6;
35
36         variabel2.a = 5;
37         variabel2.b = 10;
38
39         variabel1.hasil();
40         variabel2.hasil();
41     }
42 }
```

❖ Output :

```
run:
Hasil perkalian = 18.0
Hasil perkalian = 50.0
BUILD SUCCESSFUL (total time: 1 second)
```

5. Membuat Constructor

❖ Code :

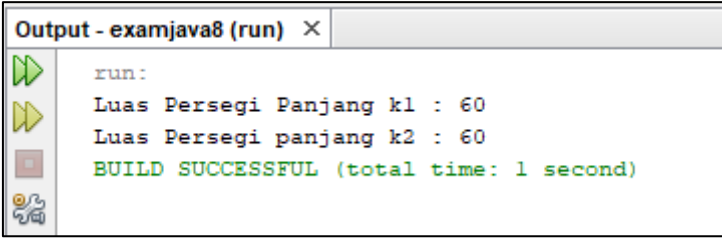
```
12
13 class konstruktor
14 {
15     int panjang;
16     int lebar;
17
18     konstruktor()
19     {
20         panjang = 15;
21         lebar = 4;
22     }
23
24     int luas()
25     {
26         return(panjang*lebar);
27     }
28 }
```

```

30 public class hitung
31 {
32     public static void main(String[] args)
33     {
34         konstruktor k1, k2;
35         k1 = new konstruktor();
36         k2 = new konstruktor();
37
38         System.out.println("Luas Persegi Panjang k1 : " + k1.luas());
39         System.out.println("Luas Persegi panjang k2 : " + k2.luas());
40     }
41 }

```

❖ Output :



```

Output - examjava8 (run) X
run:
Luas Persegi Panjang k1 : 60
Luas Persegi panjang k2 : 60
BUILD SUCCESSFUL (total time: 1 second)

```

E. TUGAS

1. Dengan menerapkan object dan method, buatlah program java menghitung :
 - a. Luas isi tabung
 - b. Luas persegi panjang
 - c. Luas trapesium
2. Buatlah program prosedural dan Program Berorientasi Objek (OOP) yang dapat menghitung volume Tabung. Tunjukkan dimana letak perbedaan pemrograman prosedural dan OOP.

LAMPIRAN

Format Pengumpulan Tugas :

Nama :

NPM :

Coding :

Screenshot Running Program :

Tugas setiap bab pertemuan diupload dalam versi PDF (Max 2 MB) melalui Elearning UMA dengan format nama file :

Nama_NPM_Tugas[Bab xx]

Contoh nama file Tugas Bab 1:

Budi_1812158909_Tugas[1]