

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* :

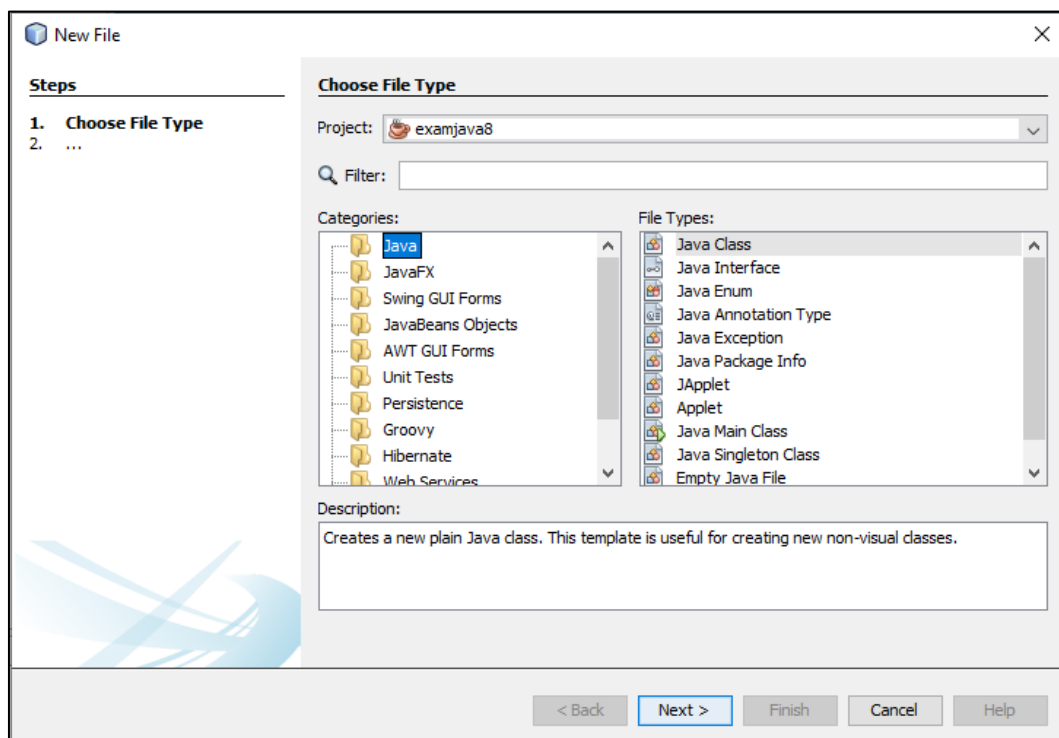
- a. Nama file **halaman kerja java** harus sama dengan **nama class**
- b. Nama class boleh terdiri dari huruf kecil dan huruf besar
- c. Nama class yang terdiri dari dua suku kata, dapat dituliskan dengan tanda underscore (_) sebagai pemisah. Contoh : Belajar_Java
- d. Nama class tidak boleh terdiri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

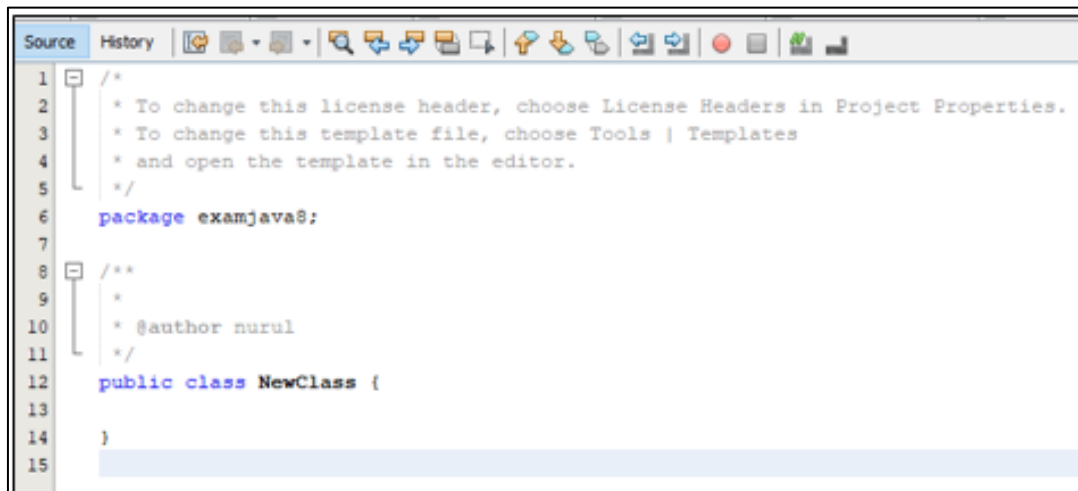
Location:

Package:

Created File:

< Back Next > **Finish** Cancel Help

Tampilan awal halaman kerja java dengan NewClass :

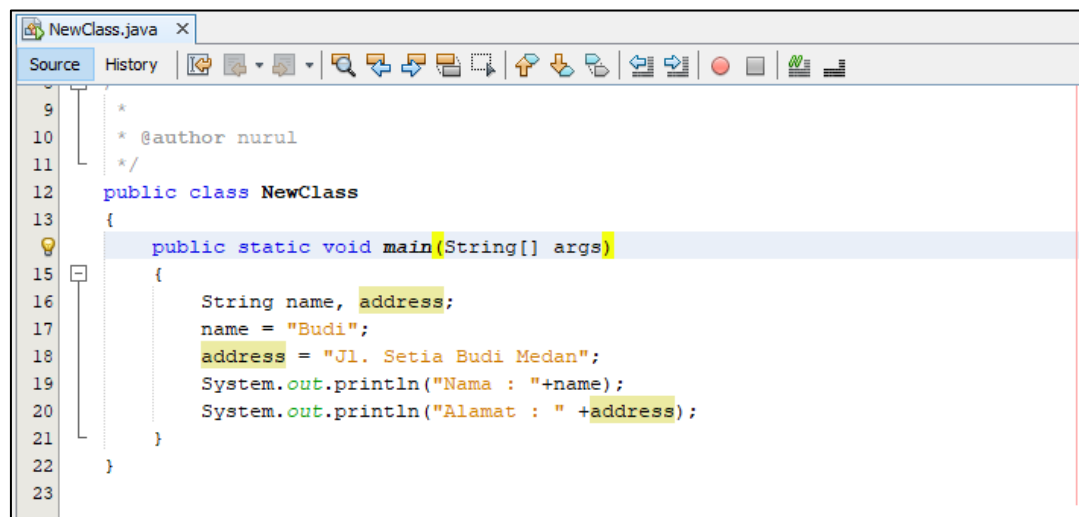


The screenshot shows the initial state of a Java IDE workspace. The 'Source' tab is active, displaying a new class template for 'NewClass'. The code includes a package declaration 'package examjava8;', a license header, an author tag '@author nurul', and the start of a public class 'NewClass'.

```
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

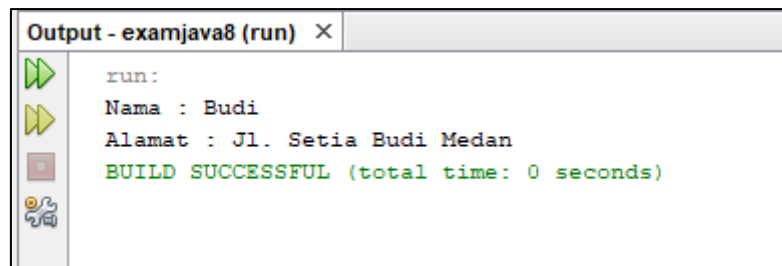
❖ Code Pemrograman:



The screenshot shows the completed code for 'NewClass.java'. The code defines a public class 'NewClass' with a 'main' method. Inside the 'main' method, variables 'name' and 'address' are declared and assigned values. The 'System.out.println' statements are used to print the values of 'name' and 'address'.

```
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 :

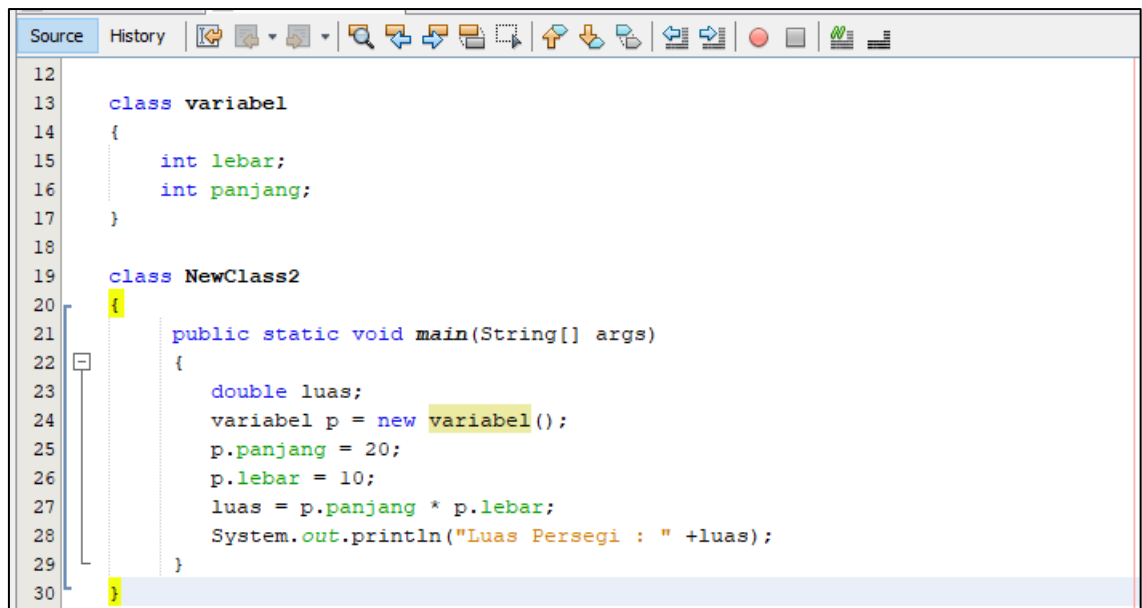


The screenshot shows the 'Output - examjava8 (run)' window. It displays the output of the program, which is 'Nama : Budi' and 'Alamat : Jl. Setia Budi Medan'. The output also indicates that the build was successful and took 0 seconds.

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

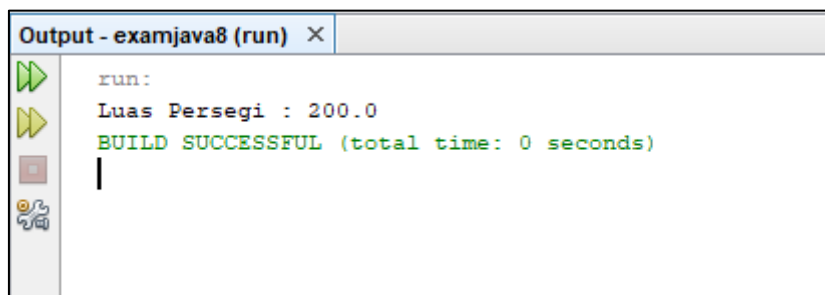
3. Membuat *Object*

❖ Code Pemrograman :



```
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     }
```

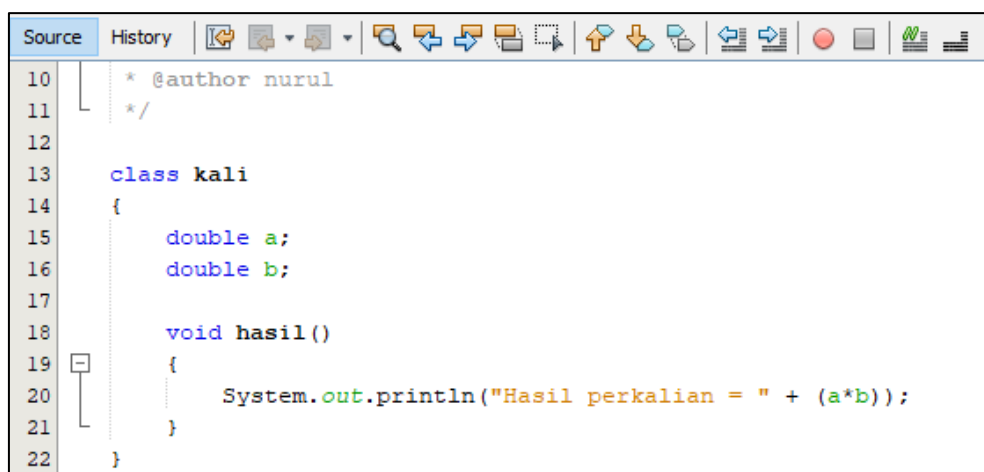
❖ Hasil Pemrograman :



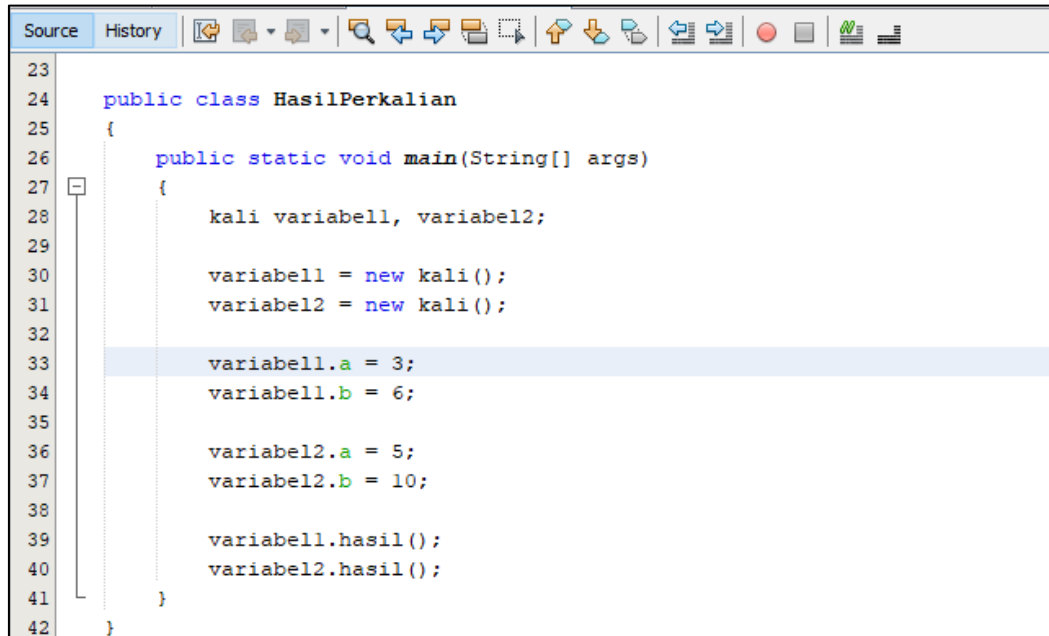
```
Output - examjava8 (run) X
run:
Luas Persegi : 200.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

4. Membuat *Method*

❖ Code :

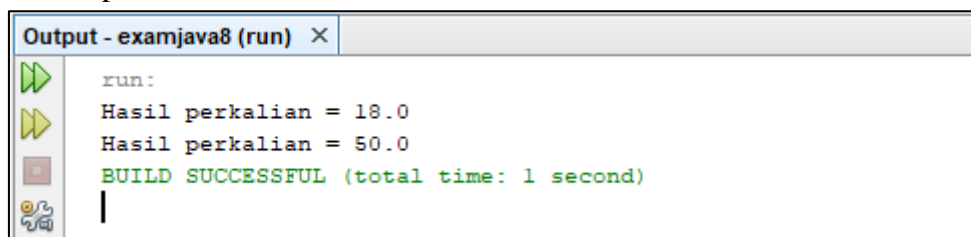


```
10     * @author nurul
11     */
12
13     class kali
14     {
15         double a;
16         double b;
17
18         void hasil()
19         {
20             System.out.println("Hasil perkalian = " + (a*b));
21         }
22     }
```



```
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 }
```

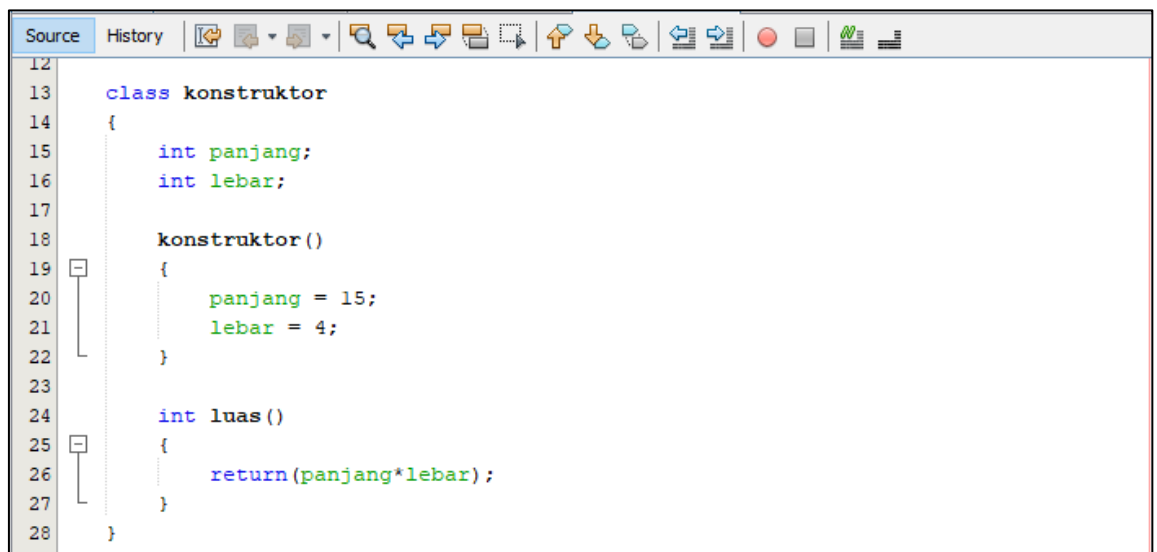
❖ Output :



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

5. Membuat *Constructor*

❖ Code :



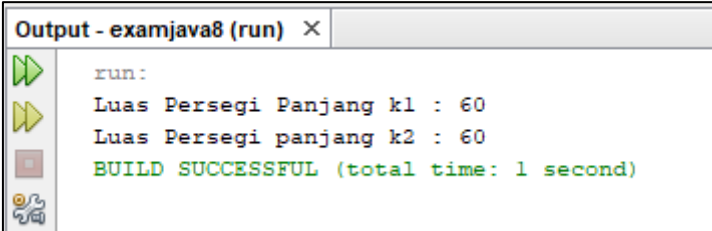
```
12
13 class konstruktor
14 {
15     int panjang;
16     int lebar;
17
18     konstruktor()
19     {
20         panjang = 15;
21         lebar = 4;
22     }
23
24     int luas()
25     {
26         return(panjang*lebar);
27     }
28 }
```

```

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 :



The screenshot shows the 'Output - examjava8 (run)' window. It displays the following text:

```

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.