

Rabu, 23 Maret 2022

TUGAS 2

Pemograman Berbasis Objek

Dosen Pengampu: Mardiyah Hasnawi, S.Kom., M.T.



Oleh:

Nama: Lina Soraya

NIM : 13020200255

Kelas : B3

Semester IV

Program Studi Teknik Informatika

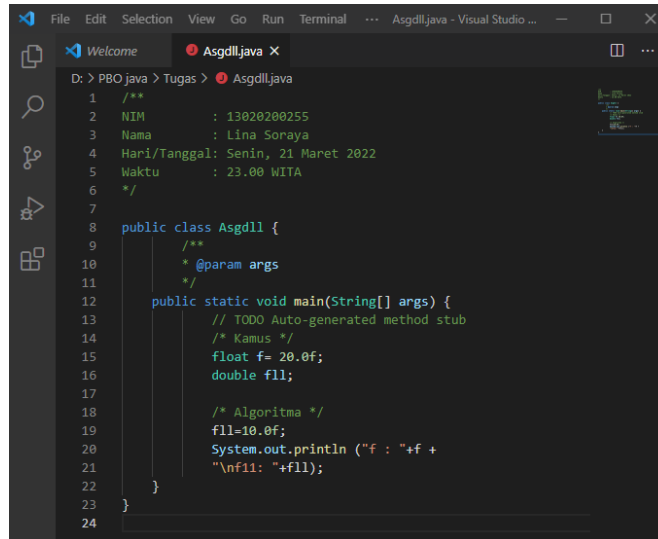
Fakultas Ilmu Komputer

UNIVERSITAS MUSLIM INDONESIA

2022

1. Kode program

- Listing program Asgdll



```
1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Senin, 21 Maret 2022
5   * Waktu    : 23.00 WITA
6   */
7
8  public class Asgdll {
9      /**
10     * @param args
11     */
12     public static void main(String[] args) {
13         // TODO Auto-generated method stub
14         /* Kamus */
15         float f= 20.0f;
16         double fl1;
17
18         /* Algoritma */
19         fl1=10.0f;
20         System.out.println ("f : "+f +
21                             "\nfl1: "+fl1);
22     }
23 }
24
```

- Tujuan program

Program tersebut adalah program yang menampilkan bilangan pecahan, yang memiliki nilai pecahan desimal. Tipe data yang digunakan adalah Float dan Double.

- Keyword dan penjelasannya

Public class Asgdll { adalah pendeklarasian kelas dengan nama kelas **Asgdll** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. **Public static void main(String[] args) {** adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

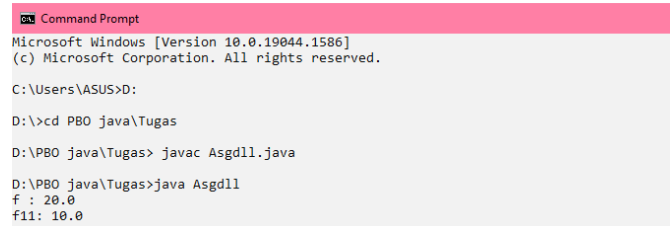
Float f= 20.0f; Float adalah tipe data yang dapat menampung bilangan pecahan. Float memiliki besar 4bytes dan 32bit dengan jangkauan $3.4028235E38 - 1.4E-45$. Penulisan bilangan float selalu diakhiri dengan simbol (f). tanpa adanya (f) di akhir bilangan maka bilangan akan dianggap sebagai double. Dengan nilai F yaitu 20.0f.

Double fl1; Double adalah tipe data yang dapat menampung bilangan pecahan. Double memiliki besar 8bytes dan 64bit dengan jangkauan $1.7076931348623157E308 - 4.9E-324$. Penulisan bilangan float selalu diakhiri dengan simbol (d).

Fll= 10.0f; merupakan variabel bertipe data double dengan nilai 10.0f.

`System.out.println("f:" + f + "\nfl;" + fl);` adalah sebuah pernyataan (statement) yang memanggil *method* `println` dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari `f` dan `fl`. `Println` berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator* (pemisah baris). Sedangkan simbol `\n` (slash n) pada *method* memiliki fungsi yang sama yaitu perintah untuk menampilkan baris baru.

- **Outputnya**



```
Microsoft Windows [Version 10.0.19044.1586]
(c) Microsoft Corporation. All rights reserved.

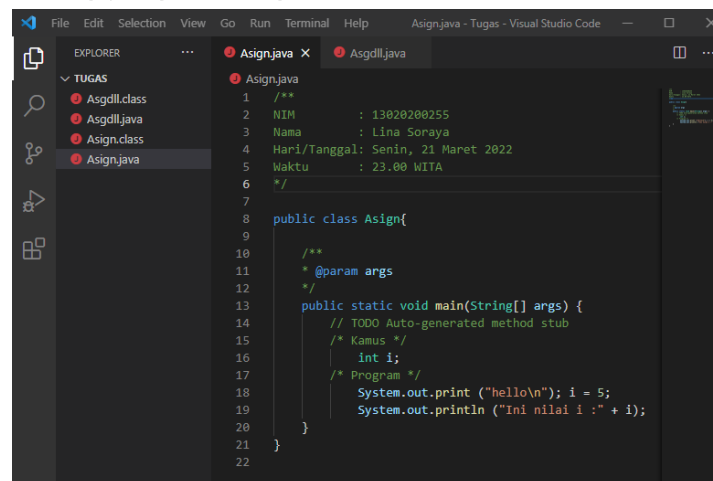
C:\Users\ASUS>D:

D:\>cd PBO java\Tugas

D:\PBO java\Tugas> javac Asgdll.java

D:\PBO java\Tugas> java Asgdll
f : 20.0
fl1: 10.0
```

- **Listing program Assign**



```
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 21 Maret 2022
5  Waktu    : 23.00 WITA
6  */
7
8  public class Assign{
9
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* Kamus */
16         int i;
17         /* Program */
18         System.out.print ("hello\n"); i = 5;
19         System.out.println ("Ini nilai i : " + i);
20     }
21 }
22
```

- **Tujuan program**

Sebuah program yang bertujuan untuk menampilkan sebuah kata “hello” dan menampilkan sebuah bilangan dengan nilai `i` dari tipe data integer.

- **Keyword dan penjelasannya**

`Public class Assign {` adalah pendeklarasian kelas dengan nama kelas **Assign** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu *method* utama yang akan dipanggil ketika program java dijalankan.

`Int i;` merupakan pendeklarasian variabel `i` pada tipe data integer tanpa masukkan nilai. Integer merupakan tipe data pada java untuk menampung bilangan bulat dengan jangkauan $32.767 - (-32.768)$.

`System.out.print ("hello\n");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan kata “hello”, diikuti dengan `\n` untuk perintah baris baru.

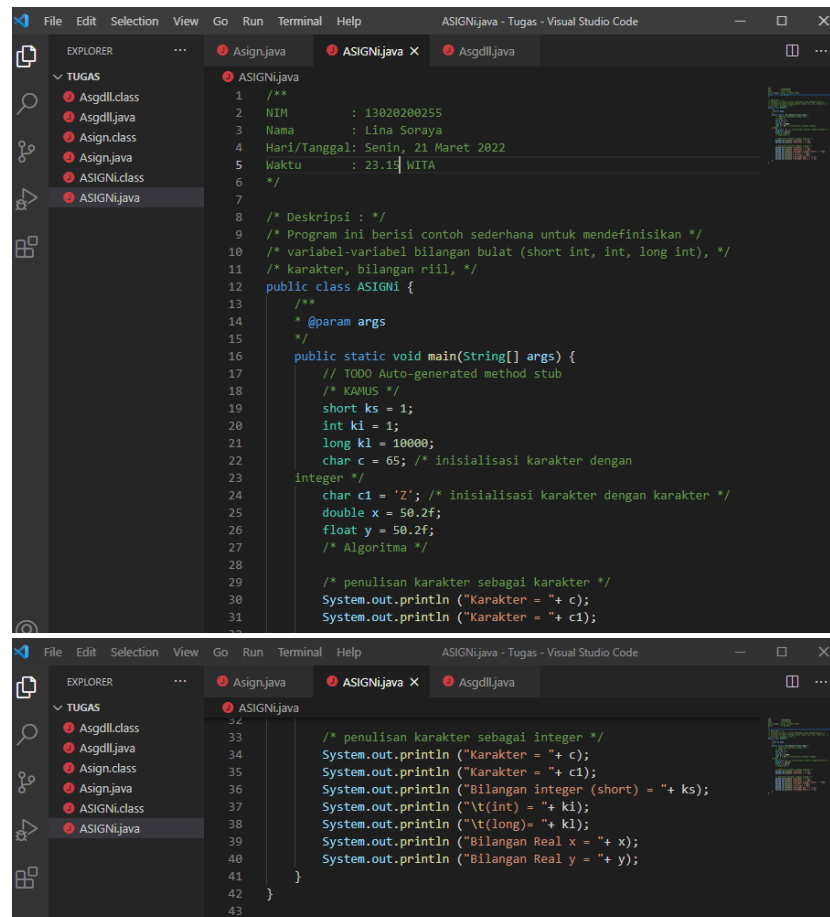
`i = 5;` variabel `i` memiliki nilai sama dengan 5.

`System.out.println ("Ini nilai i : " + i);` akan menampilkan “ini nilai i” dari variabel `i` yaitu 5.

- **Outputnya**

```
D:\PBO java\Tugas>javac Asign.java
D:\PBO java\Tugas>java Asign
Error: Could not find or load main class Asign
Caused by: java.lang.ClassNotFoundException: Asign
D:\PBO java\Tugas>javac Asign.java
D:\PBO java\Tugas>java Asign
hello
Ini nilai i :5
```

- **Listing program ASIGNi**



```
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 21 Maret 2022
5  Waktu    : 23.15 WITA
6  */
7
8  /* Deskripsi : */
9  /* Program ini berisi contoh sederhana untuk mendefinisikan */
10 /* variabel-variabel bilangan bulat (short int, int, long int), */
11 /* karakter, bilangan riil, */
12 public class ASIGNi {
13     /**
14     * @param args
15     */
16     public static void main(String[] args) {
17         // TODO Auto-generated method stub
18         /* KAHUS */
19         short ks = 1;
20         int ki = 1;
21         long kl = 10000;
22         char c = 65; /* inisialisasi karakter dengan
23         integer */
24         char c1 = 'Z'; /* inisialisasi karakter dengan karakter */
25         double x = 50.2f;
26         float y = 50.2f;
27         /* Algoritma */
28
29         /* penulisan karakter sebagai karakter */
30         System.out.println ("Karakter = " + c);
31         System.out.println ("Karakter = " + c1);
32
33         /* penulisan karakter sebagai integer */
34         System.out.println ("Karakter = " + c);
35         System.out.println ("Karakter = " + c1);
36         System.out.println ("Bilangan integer (short) = " + ks);
37         System.out.println ("\t(int) = " + ki);
38         System.out.println ("\t(long)= " + kl);
39         System.out.println ("Bilangan Real x = " + x);
40         System.out.println ("Bilangan Real y = " + y);
41     }
42 }
43
```

- **Tujuan program**

Program tersebut bertujuan untuk menampilkan variabel-variabel bilangan bulat menggunakan berbagai macam tipe data yaitu short, int, double, long, dan float. Selain

itu program tersebut juga berguna untuk mempelajari bagaimana cara penulisan karakter.

- **Keyword dan penjelasan**

`Public class ASIGNi{` adalah pendeklarasian kelas dengan nama kelas **ASIGNi** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Public static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`Short ks=1;` Variable **ks** menyimpan nilai **1** dan bertipe data short, tipe data short adalah tipe data modifier untuk menyatakan bilangan dengan kapasitas penyimpanan sampai 16 bit biner atau 2 bytes dan dapat berisi nilai negatif.

`Int ki=1;` sebuah tipe data integer dengan variabel **ki** yang memiliki nilai 1.

`Char c= 65; char cl='z';` variabel **c** dengan nilai 65 dan variabel **ci** dengan nilai Z. **Char** digunakan untuk menyimpan satu digit karakter dan karakter tersebut harus ditulis diantara tanda kutip.

`Double x= 50.2f;` Double adalah tipe data yang dapat menampung bilangan pecahan. Double disini memiliki variabel **x** dengan nilai 50.2f.

`Float y= 50.2f;` tipe data float dengan variabel **y** dan memiliki nilai 50.2f.

`Sistem.out.println ("Karater=" +c);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabel **c** yaitu A. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator* (pemisah baris).

`Sistem.out.println ("Karater=" +cl);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabel **cl** yaitu Z. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator* (pemisah baris).

`Sistem.out.println ("Karater=" +c);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabel **c** yaitu A. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator* (pemisah baris).

`Sistem.out.println ("Karater=" +cl);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan

hasil dari Karakter pada variabel cl yaitu Z. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris).

`System.out.println("Bilangan integer (short)=" +ks);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabe ks yaitu 1. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris).

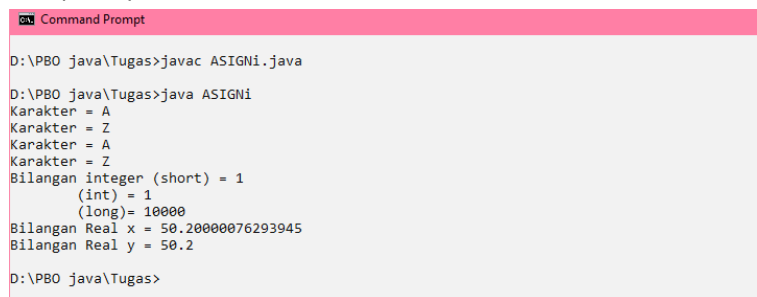
`System.out.println("\t(int)=" +ki);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabe ki yaitu 1. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris). `\t` digunakan untuk memberi beberapa spasi atau tab.

`System.out.println("\t(int)=" +kl);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabe kl yaitu 100000. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris). `\t` digunakan untuk memberi beberapa spasi atau tab.

`System.out.println("Bilangan Real x=" +x);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabe x yaitu 50.2. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris).

`System.out.println("Bilangan Real y=" +y);` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan hasil dari Karakter pada variabe y yaitu 50.2

- **Outputnya**



```
Command Prompt
D:\PBO java\Tugas>javac ASIGNi.java
D:\PBO java\Tugas>java ASIGNi
Karakter = A
Karakter = Z
Karakter = A
Karakter = Z
Bilangan integer (short) = 1
          (int) = 1
          (long)= 10000
Bilangan Real x = 50.20000076293945
Bilangan Real y = 50.2
D:\PBO java\Tugas>
```

- Listing program

```

1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Senin, 21 Maret 2022
5   * Waktu    : 23.42 WITA
6   */
7
8  import java.util.Scanner;
9  /* contoh membaca integer menggunakan Class Scanner*/
10 public class BacaData {
11     /**
12      * @param args
13      */
14     public static void main(String[] args) {
15         // TODO Auto-generated method stub
16         /* Kamus */
17         int a;
18         Scanner masukan;
19         /* Program */
20         System.out.print ("Contoh membaca dan menulis, ketik nilai integer: \n");
21         masukan = new Scanner(System.in);
22         a = masukan.nextInt(); /* coba ketik : masukan.nextInt(); ;
23
24         Apa akibatnya ?*/
25         System.out.print ("Nilai yang dibaca : "+ a);
26     }
27 }

```

- Tujuan program

Tujuan program adalah untuk menampilkan nilai integer dengan menggunakan class scanner, dimana variabel a akan menampilkan masukan dari variabel scanner, tetapi akan diubah ke tipe data integer dengan kunci nextInt().

- Keyword dan penjelasan

Import java.util.Scanner; { Untuk memasukkan paket scanner agar mempersingkat dan mempermudah pengetikan. Scanner merupakan fungsi untuk menginput data setelah program di jalankan.

Public class BacaData { sebuah kelas public yang dapat diakses oleh semua kelas dengan nama kelas BacaData.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Public static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

Int a; yaitu tipe data integer dengan variabel a. ini merupakan sebuah pendeklarasian tanpa masukan nilai.

Scanner masukan== new Scanner(System.in); Merupakan proses instansiasi scanner class ke dalam variable masukan. Hasil dari variable input akan berisi object dari class scanner.

System.out.print("Contoh membaca dan menulis, ketik nilai integer: \n"); System.out.println merupakan perintah mencetak kondisi yang berada dalam ("Contoh membaca dan Menulis, ketik nilai integer "). \n digunakan untuk new line.

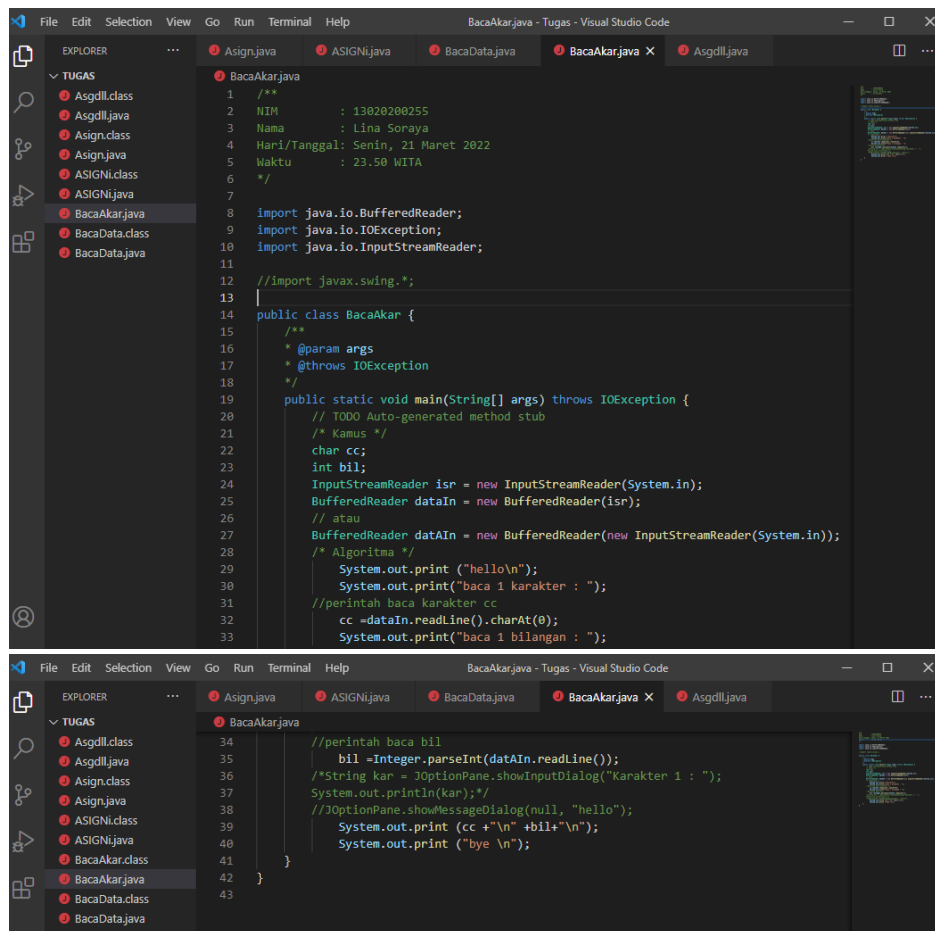
`a= masukan.nextInt();` Variable **a** akan melakukan perintah untuk memasukkan nilai `nextInt ()` untuk nilai yang bertipe integer.

`System.out.print("Nilai yang dibaca:" +a);` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan "Nilai yang dibaca" dari variabel **a**.

- **Outputnya**

```
Command Prompt
D:\PBO java\Tugas>java BacaData
Contoh membaca dan menulis, ketik nilai integer:
19012002
Nilai yang dibaca : 19012002
D:\PBO java\Tugas>javac BacaData.java
```

- **Listing program BacaAkar**



```
File Edit Selection View Go Run Terminal Help
BacaAkar.java - Tugas - Visual Studio Code

EXPLORER
TUGAS
  Asgdl.class
  Asgdl.java
  Asgn.class
  Asgn.java
  ASIGNi.class
  ASIGNi.java
  BacaAkar.java
  BacaData.class
  BacaData.java

BacaAkar.java
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 21 Maret 2022
5  Waktu    : 23.50 WITA
6  */
7
8  import java.io.BufferedReader;
9  import java.io.IOException;
10 import java.io.InputStreamReader;
11
12 //import javax.swing.*;
13
14 public class BacaAkar {
15     /**
16      * @param args
17      * @throws IOException
18      */
19     public static void main(String[] args) throws IOException {
20         // TODO Auto-generated method stub
21         /* Kamus */
22         char cc;
23         int bil;
24         InputStreamReader isr = new InputStreamReader(System.in);
25         BufferedReader dataIn = new BufferedReader(isr);
26         // atau
27         BufferedReader dataIn = new BufferedReader(new InputStreamReader(System.in));
28         /* Algoritma */
29         System.out.print ("hello\n");
30         System.out.print("baca 1 karakter : ");
31         //perintah baca karakter cc
32         cc =dataIn.readLine().charAt(0);
33         System.out.print("baca 1 bilangan : ");
34
35         //perintah baca bil
36         bil =Integer.parseInt(dataIn.readLine());
37         /*String kar = JOptionPane.showInputDialog("Karakter 1 : ");
38         System.out.println(kar);*/
39         //JOptionPane.showMessageDialog(null, "hello");
40         System.out.print (cc +"\\n" +bil+"\\n");
41         System.out.print ("bye \\n");
42     }
43 }
```

- **Keyword dan penjelasan**

`import java.io.BufferedReader;`

`import java.io.IOException;`

`import java.io.InputStreamReader;`

Class `BufferedReader`, `IOException`, `InputStreamReader` berada dalam `java.io` package yang classnya mempunyai fungsi

saling berhubungan seperti memungkinkan program untuk melakukan input dan output.

`import javax.swing.*;` Artinya berisi kelas-kelas dan menampilkan atau menerima pesan yang berfungsi untuk mendeklarasikan sebuah komponen berupa grafis dengan perintah yang dilakukan dalam pemuatan program.

`public class BacaAkar {` kode publik yang artinya dapat diakses oleh semua kelas. dengan nama kelas yaitu BacaAkar.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

Cahar cc; Variable **c** dan ertipe data **char** yang digunakan untuk menyimpan satu digit karakter dan karakter itu harus ditulis diantara tanda kutip.

Int bil; tipe data integer dengan variabel bil tanpamasukkan nilai.

`InputStreamReader isr = new InputStreamReader(System.in);` Setiap variable yang menggunakan InputStreamReader pada akhiran sebuah akhiran data yang ingin diinput. Bisa berupa karakter. Untuk akhiran system.in adalah memasukkan data dalam system.

`BufferedReader dataIn = new BufferedReader(isr);`

`// atau`

`BufferedReader dataIn = new BufferedReader(new`

`InputStreamReader(System.in));` Objek sudah siap digunakan untuk pemrosesan input, dengan melakukan pemanggilan terhadap method `read ()` maupun `readline`. `BufferedReader dataIn` dan `BufferedReader dataAIn` membaca karakter yang dapat berisi sebuah bilangan, sehingga dapat dibaca dan di input dalam system.

`System.out.print("hello\n");` `System.out.print("baca 1 bilangan: ");` adalah sebuah pernyataan (statement) yang memanggil *method print* dimana fungsi dari *method* ini adalah untuk menampilkan "hello" dan "baca 1 bilangan".

karakter `cc`

`cc =dataIn.readLine().charAt(0);`

`System.out.print("baca 1 bilangan : ");` Mendeklarasikan variable `cc` untuk mendapatkan input, dengan menggunakan fungsi `readLine()` untuk mendapatkan input dari keyboard. Lalu akan mencetak statement "Baca 1 Bilangan : ".

`//perintah baca bil`

`bil =Integer.parseInt(datAIn.readLine());` mengubah tipe data string menjadi integer.

```
String kar = JOptionPane.showInputDialog("Karakter 1 : ");
```

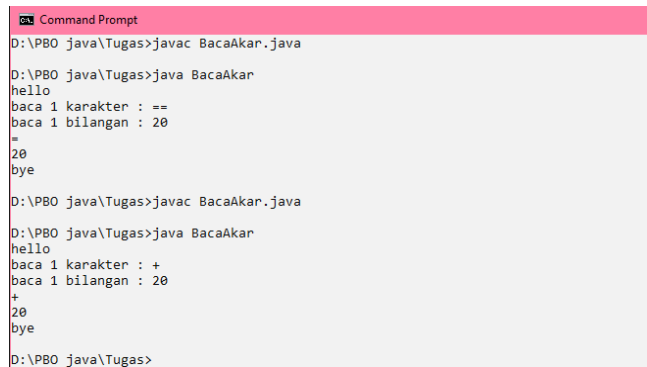
`System.out.println(kar);` Mendeklarasikan variable string kar menggunakan JOptionPane input dialog dimana user dapat menginputkan data atau nilai menggunakan JOptionPane input dialog dan akan mencetak statement “Karakter 1 : “, lalu akan mencetak nilai atau data yang telah diinput.

```
JOptionPane.showMessageDialog(null, "hello");
```

```
System.out.print (cc +"\n" +bil+"\n");
```

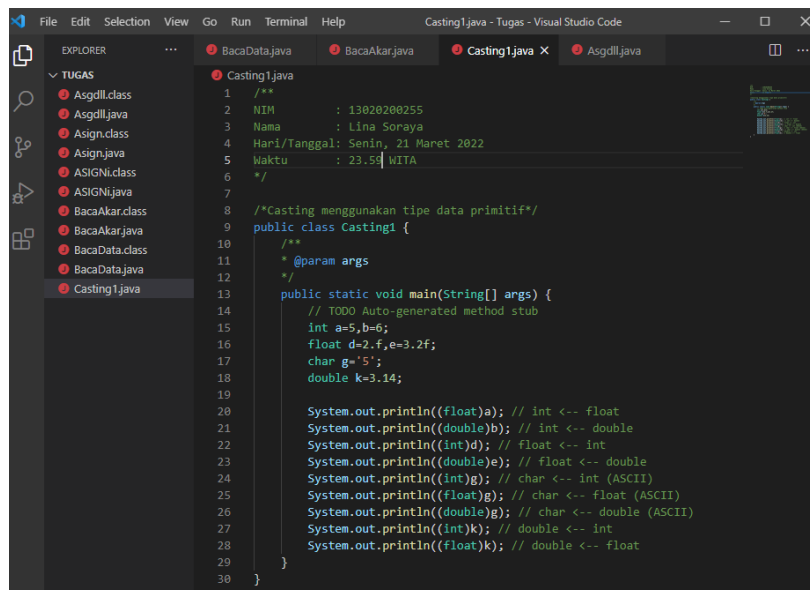
`System.out.print ("bye \n");` Menggunakan JOptionPane Message untuk menampilkan informasi seperti menampilkan pesan error, pesan validasi, dan pesan nilai yang diinputkan. Null menjadi parameter untuk komponen induk. Pada JOptionPane Message akan menampilkan statemnt “Hello”. Lalu akan mencetak dan menampilkan nilai atau data dari variable cc dan variable bil, \n digunakana untuk new line. Lalu mencetak atau menampilkan statement “bye”.

- **Outputnya**



```
Command Prompt
D:\P80 java\Tugas>javac BacaAkar.java
D:\P80 java\Tugas>java BacaAkar
hello
Baca 1 karakter : ==
Baca 1 bilangan : 20
=
20
bye
D:\P80 java\Tugas>javac BacaAkar.java
D:\P80 java\Tugas>java BacaAkar
hello
Baca 1 karakter : +
Baca 1 bilangan : 20
+
20
bye
D:\P80 java\Tugas>
```

- **Listing program Casting1**



```
File Edit Selection View Go Run Terminal Help Casting1.java - Tugas - Visual Studio Code
EXPLORER TUGAS
  Asgdl.class
  Asgdl.java
  Asgn.class
  Asgn.java
  ASIGNi.class
  ASIGNi.java
  BacaAkar.class
  BacaAkar.java
  BacaData.class
  BacaData.java
  Casting1.java
  Casting1.java

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 21 Maret 2022
5  Waktu    : 23.54 WITA
6  */
7
8  /*Casting menggunakan tipe data primitif*/
9  public class Casting1 {
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         int a=5,b=6;
16         float d=2.f,e=3.2f;
17         char g='5';
18         double k=3.14;
19
20         System.out.println((float)a); // int <-- float
21         System.out.println((double)b); // int <-- double
22         System.out.println((int)d); // float <-- int
23         System.out.println((double)e); // float <-- double
24         System.out.println((int)g); // char <-- int (ASCII)
25         System.out.println((float)g); // char <-- float (ASCII)
26         System.out.println((double)g); // char <-- double (ASCII)
27         System.out.println((int)k); // double <-- int
28         System.out.println((float)k); // double <-- float
29     }
30 }
```

- **Keyword**

`Public class Casting1` { sebuah kode publik yang dapat diakses oleh semua kelas dengan nama kelas Casting1.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Public static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int a= 5, b=6;` tipe data integer dengan variabel a dan masukkan nilainya 5, dan variabel b dengan masukkan nilai 6.

`char g='5';` tipe data char dengan variabel g dan masukkan nilai 5.

`double k= 3.14;` tipe data double dengan variabel k yang memiliki nilai 3.14. double merupakan tipe data yang menampung bilangan pecahan.

`float d = 2.f, e = 3.2f;` Variable **d** dan **e** menyimpan nilai **2.f** dan **3.2f** bertipe data **float** yang merupakan tipe data untuk menyatakan pecahan, tipe data float mempunyai 4 bytes dan 32 bit de keakuratan ng 6-7 digit.

`System.out.println((float)a); // int <-- float` Mencetak/ menampilkan nilai dari variable a menggunakan tipe data float, sebelumnya variable a memiliki tipe data integer.

`System.out.println((double)b); // int <-- double` Mencetak/menampilkan nilai dari variable b menggunakan tipe data double, sebelumnya variable b memiliki tipe data integer.

`System.out.println((int)d); // float <-- int` Mencetak/menampilkan nilai dari variable d menggunakan tipe data integer, sebelumnya variable d memiliki tipe data float.

`System.out.println((double)e); // float <-- double` Mencetak/menampilkan nilai dari variable e menggunakan tipe data double, sebelumnya variable e memiliki tipe data float.

`System.out.println((int)g); // char <-- int (ASCII)` Mencetak/menampilkan nilai dari variable g menggunakan tipe data integer yang menggunakan standar pengkodean, sebelumnya variable g memiliki tipe data char.

`System.out.println((float)g); // char <-- float (ASCII)` Mencetak/menampilkan nilai dari variable g menggunakan tipe data float yang menggunakan standar pengkodean, sebelumnya variable g memiliki tipe data char.

`System.out.println((double)g); // char <-- double (ASCII)` Mencetak/menampilkan nilai dari variable g menggunakan tipe data double yang menggunakan standar pengkodean, sebelumnya variable g memiliki tipe data char.

`System.out.println((float)k); // double <-- float` Mencetak/menampilkan nilai variable k menggunakan tipe data float, sebelumnya variable k memiliki tipe data double.

- Outputnya

```

C:\Program Files\Java\jdk-11.0.2\bin> Command Prompt
D:\PBO java\Tugas>javac Casting1.java
D:\PBO java\Tugas>java Casting1
5.0
6.0
2
3.200000047683716
53
53.0
53.0
3
3.14

```

- Listing program Casting2

```
File Edit View Go Run Terminal Help Casting2.java - Tugas - Visual Studio Code
EXPLORER
TUGAS
  AsgDll.class
  AsgDll.java
  Asign.class
  Asign.java
  ASIGNL.class
  ASIGNL.java
  BacaAkar.class
  BacaAkar.java
  BacaData.class
  BacaData.java
  Casting1.class
  Casting1.java
  Casting2.java
  AsgDll.java

Casting2.java
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2022
5  Waktu    : 00.06 WITA
6  */
7
8  /*Casting menggunakan tipe data Class*/
9  public class Casting2 {
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         int a=8,b=9;
16         float d=2.f,e=3.2f;
17         char g='5';
18         double k=3.14;
19         String n="67",m="45",l="100";
20
21         a = Integer.parseInt(n); /*Konversi String ke Integer*/
22         k = Double.parseDouble(m); /*Konversi String ke Double*/
23         d = Float.parseFloat(l); /*Konversi String ke Float*/
24         System.out.println("a : "+a+"\nk : "+k+"\nd : "+d);
25
26         n = String.valueOf(b); /*Konversi Integer ke String*/
27         m = String.valueOf(g); /*Konversi Karakter ke String*/
28         l = String.valueOf(e); /*Konversi Float ke String*/
29         System.out.println("n : "+n+"\nm : "+m+"\nl : "+l);
30         k = Double.valueOf(a).intValue(); /*Konversi Integer keDouble*/
31         double c = Integer.valueOf(b).doubleValue();
32         System.out.println("k : "+k+"\nc : "+c+"\nl : "+l);
33     }
34 }
35
```

- Keyword

Public class Casting2{ sebuah kode publik yang dapat diakses oleh semua kelas dengan nama kelas Casting2.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int a= 8, b=9;` tipe data integer dengan variabel a dan masukkan nilainya 8, dan variabel b dengan masukkan nilai 9.

`char g='5';` tipe data char dengan variabel g dan masukkan nilai 5.

`double k= 3.14;` tipe data double dengan variabel k yang memiliki nilai 3.14. double merupakan tipe data yang menampung bilangan pecahan.

`String n="67",m="45", l="100";` Variable **n**, **m**, dan **l** menyimpan nilai **"67"**, **"45"**, dan **"100"** bertipe data **String** yang menampung kumpulan karakter seperti kata dan kalimat dan ditulis diantar tanda kutip dua.

`a = Integer.parseInt(n);` Nilai variable a adalah nilai yang disimpan variable n yang akan diubah dari tipe data string ke integer.

`k = Double.parseDouble(m);` Nilai variable k adalah nilai yang disimpan variable m yang akan diubah dari tipe data string ke double.

`d = Float.parseFloat(l);` Nilai variable d adalah nilai yang disimpan variable l yang akan diubah dari tipe data string ke double.

`System.out.println("a : "+a+"\nk : "+k+"\nd : "+d);` Akan mencetak/menampilkan statement **"a : "**, **"k"**, dan **"d"** yang akan menampilkan nilai dari variable a, k, dan d setelah diubah, `\n` untuk new line.

`n = String.valueOf(b);` Nilai variable n adalah nilai yang disimpan variable b yang akan diubah dari tipe data integer ke string.

`m = String.valueOf(g);` Nilai variable m adalah nilai yang disimpan variable g yang akan diubah dari tipe data char ke string.

`l = String.valueOf(e);` Nilai variable l adalah nilai yang disimpan variable e yang akan diubah dari tipe data float ke string.

`System.out.println("n : "+n+"\nm : "+m+"\nl : "+l);` Akan mencetak dan menampilkan statement **"n : "**, **"m"**, dan **"l"** yang akan menampilkan nilai dari variable n, m, dan l setelah diubah, `\n` untuk new line.

`k = Double.valueOf(a).intValue();` Nilai variable k adalah nilai yang disimpan variable a yang akan diubah dari tipe data integer ke double.

`double c = Integer.valueOf(b).doubleValue();` Nilai variable c yang memiliki tipe data double dan nilai variable c adalah nilai yang disimpan variable b yang akan diubah dari tipe data double ke integer.

`System.out.println("k : "+k+"\nc : "+c+"\nl : "+l);` Akan mencetak/menampilkan statement **"k : "**, **"c"**, dan **"l"** yang akan menampilkan nilai dari variable k, c, dan l setelah diubah, `\n` untuk new line.

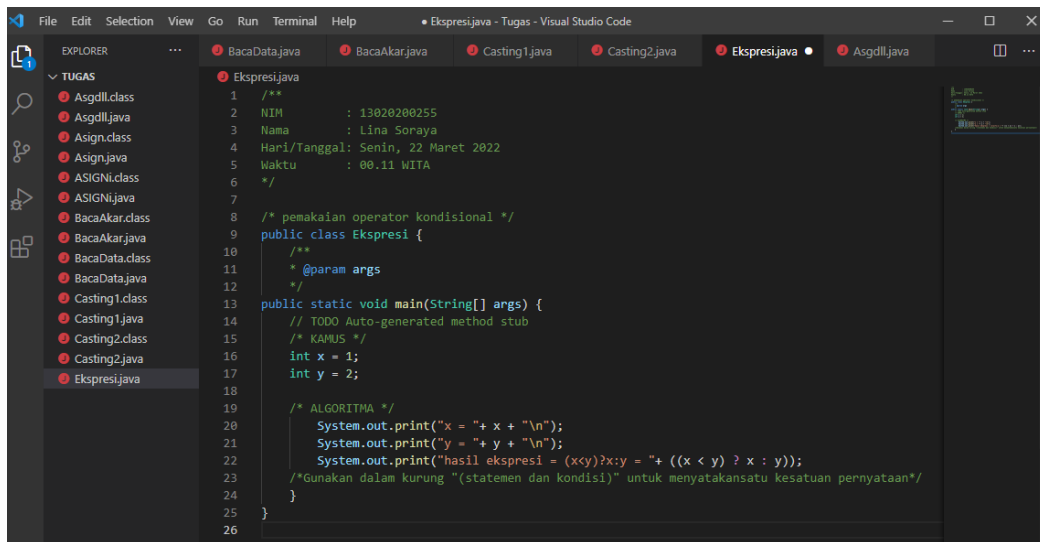
- Outputnya

```
Command Prompt

D:\PBO java\Tugas>javac Casting2.java

D:\PBO java\Tugas>java Casting2
a : 67
k : 45.0
d : 100.0
n : 9
m : 5
l : 3.2
k : 67.0
c : 9.0
l : 3.2
```

- Listing program Casting2



```
File Edit Selection View Go Run Terminal Help • Ekspresijava - Tugas - Visual Studio Code

EXPLORER
TUGAS
  Asgdll.class
  Asgdll.java
  Asign.class
  Asign.java
  ASIGNi.class
  ASIGNi.java
  BacaAkar.class
  BacaAkar.java
  BacaData.class
  BacaData.java
  Casting1.class
  Casting1.java
  Casting2.class
  Casting2.java
  Ekspresijava

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2022
5  Waktu    : 00.11 WITA
6  */
7
8  /* pemakaian operator kondisional */
9  public class Ekspresi {
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* KAHUS */
16         int x = 1;
17         int y = 2;
18
19         /* ALGORITMA */
20         System.out.print("x = " + x + "\n");
21         System.out.print("y = " + y + "\n");
22         System.out.print("hasil ekspresi = (x<y)?x:y = " + ((x < y) ? x : y));
23         /*Gunakan dalam kurung "(statemen dan kondisi)" untuk menyatakansatu kesatuan pernyataan*/
24     }
25 }
26
```

- Keyword

Public class Casting2{ sebuah kode publik yang dapat diakses oleh semua kelas dengan nama kelas Casting2.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

int x=1; int y=2; tipe data integer dengan variabel x dan masukkan nilainya 1, dan variabel y dengan masukkan nilai 2.

System.out.print("x = " + x + "\n"); Akan mencetak dan menampilkan statement "x : " yang akan menampilkan nilai dari variable x, \n untuk new line.

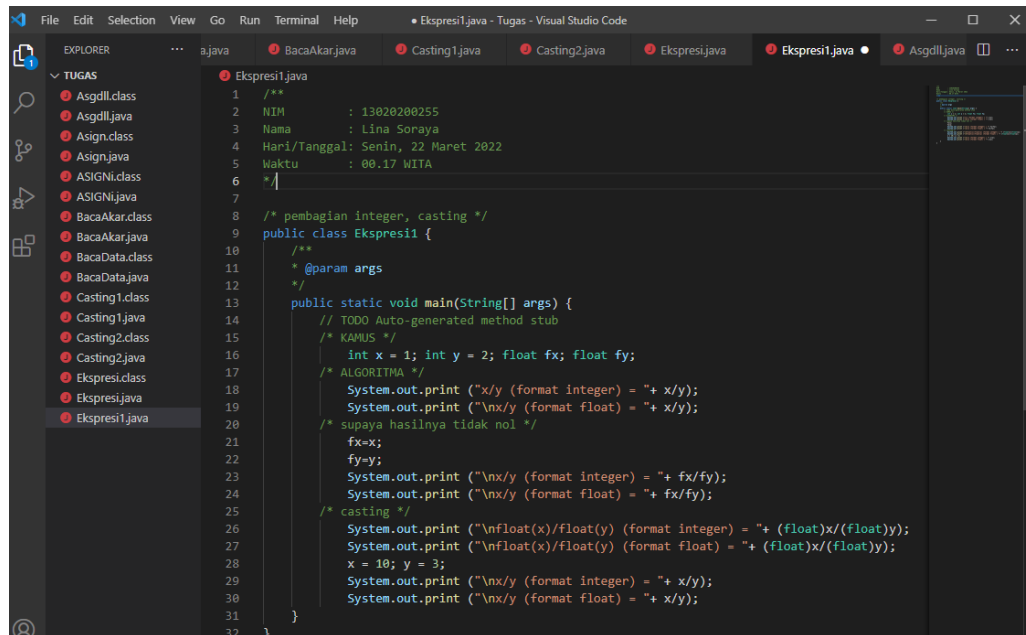
System.out.print("y = " + y + "\n"); Akan mencetak dan menampilkan statement "y : " yang akan menampilkan nilai dari variable y, \n untuk new line.

`System.out.print("hasil ekspresi = (x<y)?x:y = " + ((x < y) ?(x : y));` Akan mencetak dan menampilkan statement “hasil ekspresi : = (x<y)?x:y = ” yang akan menampilkan nilai dari hasil ekspresi `((x < y) ?(x : y))`, \n untuk new line.

- **Outputnya**

```
Command Prompt
D:\PBO java\Tugas>javac Ekspresi.java
D:\PBO java\Tugas>java Ekspresi
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1
D:\PBO java\Tugas>
```

- **Listing Program Ekspresi1**



```
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2022
5  Waktu    : 00.17 WITA
6  */
7
8  /* pembagian integer, casting */
9  public class Ekspresi1 {
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* KAMUS */
16         int x = 1; int y = 2; float fx; float fy;
17         /* ALGORITMA */
18         System.out.print ("x/y (format integer) = " + x/y);
19         System.out.print ("\nx/y (format float) = " + x/y);
20         /* supaya hasilnya tidak nol */
21         fx=x;
22         fy=y;
23         System.out.print ("\nx/y (format integer) = " + fx/fy);
24         System.out.print ("\nx/y (format float) = " + fx/fy);
25         /* casting */
26         System.out.print ("\nfloat(x)/float(y) (format integer) = " + (float)x/(float)y);
27         System.out.print ("\nfloat(x)/float(y) (format float) = " + (float)x/(float)y);
28         x = 10; y = 3;
29         System.out.print ("\nx/y (format integer) = " + x/y);
30         System.out.print ("\nx/y (format float) = " + x/y);
31     }
32 }
```

- **Keyword**

`Public class Casting2{` sebuah kode publik yang dapat diakses oleh semua kelas dengan nama kelas Casting2.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int x=1; int y=2;` tipe data integer dengan variabel X dan memiliki nilai 1, variabel y memiliki nilai 2.

`float fx; float fy;` tipe data float dengan pendeklarasian variabel fx dan fy tanpa adanya masukkan nilai.

`System.out.print ("x/y (format integer) = "+ x/y);` Mencetak dan menampilkan statement "x/y (format integer) = " dan menampilkan nilai dari x atau y.

`System.out.print ("\nx/y (format float) = "+ x/y);` Mencetak dan menampilkan statement "x/y (format float) = " dan menampilkan nilai dari x atau y.

`/* supaya hasilnya tidak nol */`

`fx=x;` Variable fx akan menampilkan nilai dari variable x.

`fy=y;` Variable fy akan menampilkan nilai dari variable y.

`System.out.print ("\nx/y (format integer) = "+ fx/fy);` Mencetak dan menampilkan statement "x/y (format integer) = " dan menampilkan nilai dari fx atau fy. \n sebagai new line.

`System.out.print ("\nx/y (format float) = "+ fx/fy);` Mencetak dan menampilkan statement "x/y (format float) = " dan menampilkan nilai dari fx atau fy. \n sebagai new line.

`/* casting */`

`System.out.print ("\nfloat(x)/float(y) (format integer) = "+ (float)x/(float)y);` Mencetak dan menampilkan statement "float(x)/float(y) (format integer) = " dan menampilkan nilai dari float(x) atau float(y) menggunakan tipe data float.

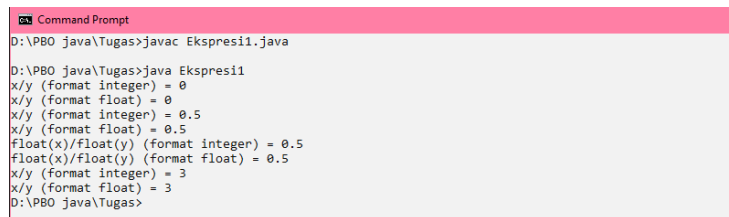
`System.out.print ("\nfloat(x)/float(y) (format float) = "+ (float)x/(float)y);` Mencetak dan menampilkan statement "float(x)/float(y) (format float) = " dan menampilkan nilai dari float(x) atau float(y) menggunakan tipe data float.

`x = 10; y = 3;` variable x menyimpan nilai 10 dan variable y menyimpan variable 3.

`System.out.print ("\nx/y (format integer) = "+ x/y);` Mencetak dan menampilkan statement "x/y (format integer) = " dan menampilkan nilai dari x/y.

`System.out.print ("\nx/y (format float) = "+ x/y);` Mencetak dan menampilkan statement "float(x)/float(y) (format float) = " dan menampilkan nilai dari float(x) atau float(y) menggunakan tipe data float.

- **Outputnya**



```
D:\PBO java\Tugas>javac Ekspres11.java
D:\PBO java\Tugas>java Ekspres11
x/y (format integer) = 0
x/y (format float) = 0
x/y (format integer) = 0.5
x/y (format float) = 0.5
float(x)/float(y) (format integer) = 0.5
float(x)/float(y) (format float) = 0.5
x/y (format integer) = 3
x/y (format float) = 3
D:\PBO java\Tugas>
```


- Listing program Hello

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2024
5  Waktu    : 00.54 WITA
6  */
7
8  public class Hello {
9
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* menuliskan hello ke layar */
16         System.out.print("Hello");
17         /* menuliskan hello dan ganti baris*/
18         System.out.print("\nHello ");
19         /* menuliskan hello dan ganti baris*/
20         System.out.println("World");
21         System.out.println("Welcome");
22     }
23 }

```

- Keyword

Public class Hello { sebuah kode publik yang dapat diakses oleh semua kelas dengan nama kelas Hello.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

/* menuliskan hello ke layar */

System.out.print("Hello"); Mencetak dan menampilkan statement “Hello”.

/* menuliskan hello dan ganti baris*/

System.out.print("\nHello "); Mencetak dan menampilkan statement “Hello”, \n untuk new line atau ganti baris

/* menuliskan hello dan ganti baris*/

System.out.println("World");

System.out.println("Welcome"); Mencetak dan menampilkan statement “World” dan statement “Welcome”.

- Outputnya

```

D:\PBO java\Tugas>javac Hello.java

D:\PBO java\Tugas>java Hello
Hello
Hello World
Welcome

```

- Listing program Incr

```

1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Senin, 22 Maret 2022
5   * Waktu    : 01.12 WITA
6   */
7
8  /* Efek dari operator ++ */
9  public class Incr {
10
11     /**
12     * @param args
13     */
14     public static void main(String[] args) {
15         // TODO Auto-generated method stub
16         /* Kamus */
17         int i, j;
18
19         /* Program */
20         i = 3;
21         j = 1++;
22
23         System.out.println ("Nilai i : " + (++i) + "\nNilai j : " + j);
24     }
25 }
26

```

- Keyword

/* Efek dari operator ++ */

public class Incr { { sebuah kode publik yang dapat diakses oleh semua kelas dengan nama kelas Incr.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

Int i,j; tipe data integer dengan yang memiliki dua variabel yaitu variabel i dan j.

i=3; j=i++; variabel memiliki masukkan nilai 3 dan variabel j memiliki masukkan nilai yang akan bertambah satu.

System.out.println("Nilai: " + (++i) "\nNilai j: " + j); adalah sebuah pernyataan (statement) yang memanggil *method println* dimana fungsi dari *method* ini adalah untuk menampilkan "Nilai i:" yang akan terus bertambah satu sebelum pemrosesan dan menampilkan statemen "Nilai j: " kemudian akan menampilkan nilai dari variabel j. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator* (pemisah baris). Sedangkan simbol **\n (slash n)** pada *method* memiliki fungsi yang sama yaitu perintah untuk menampilkan baris baru.

- Outputnya

```

C:\> Command Prompt

D:\PBO java\Tugas>javac Incr.java

D:\PBO java\Tugas>java Incr
Nilai i : 5
Nilai j : 3

```

- Listing program Oper1

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2021
5  Waktu    : 01.14 WITA
6  */
7
8  /* pemakaian beberapa operator terhadap bit */
9  public class Oper1 {
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* KAIRUS */
16         int n = 10; /* 1010 */
17         int x = 1; /* 1 */
18         int y = 2; /* 10 */
19         /* ALGORITMA */
20         System.out.println("n = " + n);
21         System.out.println("x = " + x);
22         System.out.println("y = " + y);
23         System.out.println("n & 8 = " + (n & 8)); /* 1010 AND 1000 */
24         System.out.println("x & 8 = " + (x & 8)); /* 1 AND 0111 */
25         System.out.println("y << 2 = " + (y << 2)); /* 10 ==> 1000 = 8 */
26         System.out.println("y >> 3 = " + (y >> 3)); /* 10 ==> 0000 = 0 */
27     }
28 }

```

- Keyword

Public class Oper1 { adalah pendeklarasian kelas dengan nama kelas **Oper1** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. **Public static void main(String[] args) {** adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Public static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

Int n= 10; sebuah tipe data integer dengan variabel n dan memiliki nilai 10.

Int x=1; sebuah tipe data integer dengan variabel x dan memiliki nilai 1.

Int y =2 ; sebuah tipe data integer dengan variabel y dan memiliki nilai 2.

System.out.println ("n = "+ n);

System.out.println ("x = "+ x);

System.out.println ("y = "+ y); Akan mencetak dan menampilkan statement “n : “, “x”, dan “y” yang akan menampilkan nilai dari variable n, x, dan y.

System.out.println ("n & 8 = "+ (n & 8)); /* 1010 AND 1000 */ Akan mencetak dan menampilkan statement "n & 8 = " dan akan mencetak dan menampilkan hasil dari operator bitwise dari n & 8, dengan jika 1010 AND 1000 akan menghasilkan nilai biner baru yang diubah ke bentuk desimal.

`System.out.println ("x & ~ 8 = " + (x & ~8)); /* 1 AND 0111 */` Akan mencetak dan menampilkan statement "y & ~ 8 = " dan akan mencetak dan menampilkan hasil dari operator bitwise dari y & ~ 8, jika 1 Komplemen AND 0111 Menghasilkan nilai biner yang terbalik dari hasilnya.

`System.out.println ("y << 2 = " + (y << 2)); /* 10 ==> 1000 = 8 */` Akan mencetak dan menampilkan statement "y << 2 = " dan akan mencetak dan menampilkan hasil dari operasi bitwise dari y << 2, jika 1 -==> 1000 = 8, jika iya akan mencetak nilai biner yang digeser ke kiri.

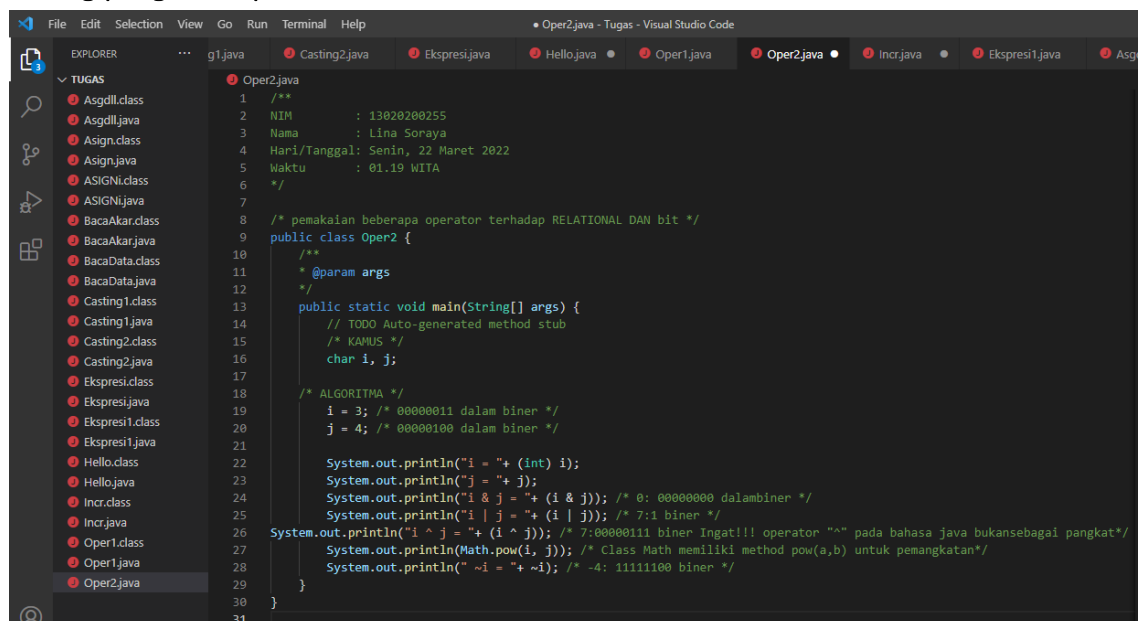
`System.out.println ("y >> 3 = " + (y >> 3)); /* 10 ==> 0000 = 0 */` Akan mencetak dan menampilkan statement "y >> 3 = " dan akan mencetak dan menampilkan hasil dari operator bitwise dari y >> 3, jika 1 ==> 0000 = 0 akan mencetak nilai biner yang digeser ke kanan.

- **Outputnya**



```
Command Prompt
D:\PBO java\tugas>javac Oper1.java
D:\PBO java\tugas>java Oper1
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0
```

- **Listing program Oper2**



```
Oper2.java - Visual Studio Code
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2022
5  Waktu    : 01.19 WITA
6  */
7
8  /* pemakaian beberapa operator terhadap RELATIONAL DAN bit */
9  public class Oper2 {
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* KAMUS */
16         char i, j;
17
18         /* ALGORITMA */
19         i = 3; /* 00000011 dalam biner */
20         j = 4; /* 00000100 dalam biner */
21
22         System.out.println("i = " + (int) i);
23         System.out.println("j = " + j);
24         System.out.println("i & j = " + (i & j)); /* 0: 00000000 dalam biner */
25         System.out.println("i | j = " + (i | j)); /* 7: 01111011 dalam biner */
26         System.out.println("i ^ j = " + (i ^ j)); /* 7: 00000111 biner Ingat!!! operator "^" pada bahasa java bukan sebagai pangkat */
27         System.out.println(Math.pow(i, j)); /* Class Math memiliki method pow(a,b) untuk pemangkatan */
28         System.out.println("~i = " + ~i); /* -4: 11111100 biner */
29     }
30 }
31
```

- **Keywordnya**

`Public class Oper2 {` adalah pendeklarasian kelas dengan nama kelas **Oper2** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`char i, j;` Variable `i` dan `j` bertipe data **char** yang digunakan untuk menyimpan satu digit karakter dan karakter itu harus ditulis diantara tanda kutip.

`i = 3;` `/* 00000011 dalam biner */` variable `i` menyimpan nilai 3, dalam biner adalah 0000001.

`j = 4;` `/* 00000100 dalam biner */` Variable `j` menyimpan niali 4, dalam biner adalah 00000100

`System.out.println("i = " + (int) i);`

`System.out.println("j = " + j);` Mencetak dan menampilkan statement "`i =`" dan statement "`j =`", dan mencetak dan menampilkan nilai dari variable `i` yang bertipe dasar integer dan nilai dari `j`

`System.out.println("i & j = " + (i & j));` Akan mencetak dan menampilkan statement "`i & j =`" dan akan mencetak dan menampilkan hasil dari operator bitwise dari `i & j`, dengan jika 0 AND 00000000 akan menghasilkan nilai biner baru yang diubah ke bentuk desimal.

`System.out.println("i | j = " + (i | j));` Akan mencetak dan menampilkan statement "`i | j =`" dan akan mencetak dan menampilkan hasil dari operator bitwise dari `i | j`, dengan 7 OR 00000111 jika kedua bit 0 akan bernilai 0 juga, selain itu nilai bit akan di set menjadi 1.

`System.out.println("i ^ j = " + (i ^ j));` Akan mencetak dan menampilkan statement "`i ^ j =`" dan akan mencetak dan menampilkan hasil dari operator bitwise dari `i ^ j`, dengan 7 XOR 00000111 jika salah satu dar dua variable memiliki nilai 1 akan bernilai 1 juga, tetapi jika nilai kedua variable sama akan menghasilkan 0.

`System.out.println(Math.pow(i, j));` Mencetak dan menampilkan pangkat dari variable `i` dan variable `j`. Fungsi `Math.pow` digunakan untuk membuat pangkat.

`System.out.println("~i = " + ~i);` Mencetak dan menampilkan statement komplemen variable `i` "`~i =`" dan akan mencetak dan menampilkan nilai yang disimpan di komplemen variable `i`.

- Outputnya

```

Command Prompt
D:\PBO java\Tugas>javac Oper2.java

D:\PBO java\Tugas>java Oper2
i = 3
j = 4
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4

```

- Listing program Oper3

```

Oper3.java - Tugas - Visual Studio Code
EXPLORER
TUGAS
  Asgdl.class
  Asgdl.java
  Asgn.class
  Asgn.java
  ASIGNi.class
  ASIGNi.java
  BacaAkar.class
  BacaAkar.java
  BacaData.class
  BacaData.java
  Casting1.class
  Casting1.java
  Casting2.class
  Casting2.java
  Ekspresi.class
  Ekspresi.java
  Ekspresi1.class
  Ekspresi1.java
  Hello.class
  Hello.java
  Incr.class
  Incr.java
  Oper3.java
  Oper1.java
  Oper2.java
  Incr.java
  Ekspresi1.java
  Asgdl.java

Oper3.java
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2022
5  Waktu    : 01.33 WITA
6  */
7
8  public class Oper3 {
9      /**
10     * @param args
11     */
12     public static void main(String[] args) {
13         // TODO Auto-generated method stub
14
15         /* Algoritma */
16         if (true && true){ System.out.println(true && true); }/* true = true and true */
17         if (true & true) { System.out.println(true & false); } /* true & true */
18         if (true) { System.out.println(true); } /* true */
19         if (true || true){ System.out.println(true); } /* true = true or true */
20         if (true|false) { System.out.println(true|false); } /* true|false */
21     }
22 }
23
24
25

```

- Keyword

Public class Oper3 { adalah pendeklarasian kelas dengan nama kelas **Oper3** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. **Public static void main(String[] args) {** adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

if (true && true){

System.out.println(true && true); }/* terdapat percabangan if yang dimana terdapat kondisi Operator Logika AND. Mencetak dan menampilkan nilai true jika kedua operand bernilai true, selain itu operator akan menghasilkan false.

if (true & true) {

System.out.println(true & false); } terdapat percabangan if yang dimana terdapat kondisi Operator AND. Mencetak dan menampilkan nilai true dengan mengavaluasi

secara terpisah kemudian membandingkan hasilnya. Jika keduanya benar maka akan mengembalikan true. Tetapi jika salah atau keduanya salah akan menghasilkan false.

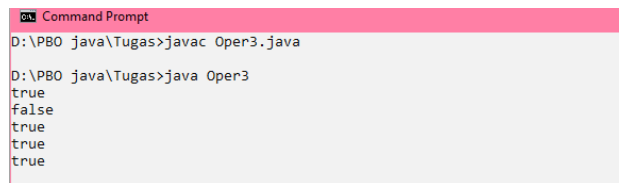
`if (true) { System.out.println(true); }` terdapat percabangan if yang dimana terdapat kondisi true. Mencetak dan menampilkan nilai true karena pada percabangan menampilkan kondisi true.

`if (true || true){ System.out.println(true); }` terdapat percabangan if yang dimana terdapat kondisi Operator Logika OR.

Mencetak dan menampilkan nilai true jika salah satu operand bernilai true.

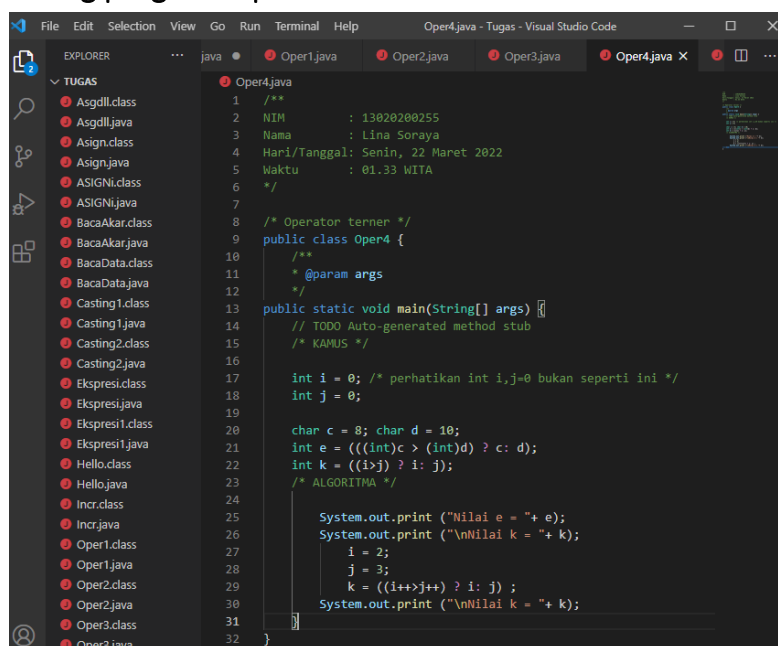
`if (true|false) { System.out.println(true|false); } /* true|false */` terdapat percabangan if yang dimana terdapat kondisi Operator OR. Mencetak dan menampilkan nilai true atau false. Operator | membandingkan operator kiri dan kanan apapun kondisinya.

- **Outputnya**



```
Command Prompt
D:\PBO java\Tugas>javac Oper3.java
D:\PBO java\Tugas>java Oper3
true
false
true
true
true
```

- **Listing program Oper4**



```
Oper4.java - Tugas - Visual Studio Code
EXPLORER
TUGAS
  Asgdl1.class
  Asgdl1.java
  Assign.class
  Assign.java
  ASIGNi.class
  ASIGNi.java
  BacaAkar.class
  BacaAkar.java
  BacaData.class
  BacaData.java
  Casting1.class
  Casting1.java
  Casting2.class
  Casting2.java
  Ekspresi.class
  Ekspresi.java
  Ekspresi1.class
  Ekspresi1.java
  Hello.class
  Hello.java
  Incr.class
  Incr.java
  Oper1.class
  Oper1.java
  Oper2.class
  Oper2.java
  Oper3.class
  Oper3.java
  Oper4.java

Oper4.java
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 22 Maret 2022
5  Waktu    : 01.33 WITA
6  */
7
8  /* Operator terner */
9  public class Oper4 {
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* KAMUS */
16
17         int i = 0; /* perhatikan int i,j=0 bukan seperti ini */
18         int j = 0;
19
20         char c = 0; char d = 10;
21         int e = (((int)c > (int)d) ? c: d);
22         int k = ((i>j) ? i: j);
23         /* ALGORITMA */
24
25         System.out.print ("Nilai e = "+ e);
26         System.out.print ("\nNilai k = "+ k);
27         i = 2;
28         j = 3;
29         k = ((i++>j++) ? i: j) ;
30         System.out.print ("\nNilai k = "+ k);
31
32     }
```

- **Keyword**

`Public class Oper3 {` adalah pendeklarasian kelas dengan nama kelas **Oper3** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*

`int i= 0; int j:0;` sebuah tipe data integer dengan variabel *i* yang memiliki nilai 0, dan variabel *j* yang juga memiliki nilai 0.

`char c= 8; char d= 10;` Variable **c** dan **d**, yang memiliki nilai 8 dan 10 bertipe data **char** yang digunakan untuk menyimpan satu digit karakter dan karakter itu harus ditulis diantara tanda kutip.

`int e = (((int)c > (int)d) ? c: d);` menentukan nilai variable *e* menggunakan operator `?` : dari variable *c* dan *d*, lalu memasukkan hasilnya dalam variable *e*.

`int k = ((i>j) ? i: j);` menentukan nilai variable *k* menggunakan operator `?` : dari variable *j* dan *i*, lalu memasukkan hasilnya dalam variable *k*.

`System.out.print ("Nilai e = "+ e);` Mencetak dan menampilkan statement "Nilai e = ", lalu mencetak dan menampilkan nilai dari variable *e*.


`System.out.print ("\nNilai k = "+ k);` Mencetak dan menampilkan statement "Nilai k = ", lalu mencetak dan menampilkan nilai dari variable *k*.

`i = 2; j = 3;` Variable *i* dan *j* menyimpan nilai 2 dan 3.

`k = ((i++>j++) ? i: j);` menentukan nilai variable *k* menggunakan operator `?` : dari variable *j* dan *i*, lalu memasukkan hasilnya dalam variable *k*. nilai *i* dan *j* akan bertambah 1 setelah pemrosesan.

`System.out.print ("\nNilai k = "+ k);` Mencetak dan menampilkan statement "Nilai k = ", lalu mencetak dan menampilkan nilai dari variable *k*.

- **Outputnya**



```
Command Prompt
D:\PBO\java\Tugas>javac Oper4.java
D:\PBO\java\Tugas>java Oper4
Nilai e = 10
Nilai k = 0
Nilai k = 4
D:\PBO\java\Tugas>
```


- Listing program Operator

```

1  *
2  M      : 13020200255
3  ma     : Lina Soraya
4  ri/Tanggal: Senin, 22 Maret 2022
5  ktu    : 02.36 WITA
6
7
8  Contoh pengoperasian variabel bertipe dasar */
9  blic class Oprator {
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* Kamus */
16         boolean Bool1, Bool2, TF ; int i,j, hsl ;
17         float x,y,res;
18         /* algoritma */
19         System.out.println("Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output");
20         Bool1 = true;
21         Bool2 = false;
22         TF = Bool1 && Bool2 ; /* Boolean AND */
23         System.out.print("\n TRUE/FALSE FOR BOOLEAN AND : "+TF);
24         TF = Bool1 || Bool2 ; /* Boolean OR */
25         System.out.print("\n TRUE/FALSE FOR BOOLEAN OR : "+TF);
26         TF = ! Bool1 ; /* NOT */
27         System.out.print("\n TRUE/FALSE FOR BOOLEAN NOT : "+TF);
28         TF = Bool1 ^Bool2; /* XOR */
29         System.out.print("\n TRUE/FALSE FOR BOOLEAN XOR : "+TF);
30         System.out.print("\n operasi numerik");
31
32         /* operasi numerik */
33         i = 5; j = 2 ;
34         hsl = i+j; System.out.print("\n Hasil : "+hsl);
35         hsl = i - j; System.out.print("\n Hasil : "+hsl);
36         hsl = i / j; System.out.print("\n Hasil : "+hsl);
37         hsl = i * j; System.out.print("\n Hasil : "+hsl);
38         hsl = i /j ; /* pembagian bulat */ System.out.print("\n Hasil : "+hsl);
39         hsl = i%j ; /* sisa. modulo */
40         System.out.print("\n Hasil : "+hsl);
41         System.out.print("\n operasi numerik");
42
43         /* operasi numerik */
44         x = 5 ; y = 5 ;
45         res = x + y; System.out.print("\n RESULT : "+res);
46         res = x - y; System.out.print("\n RESULT: "+res);
47         res = x / y; System.out.print("\n RESULT : "+res);
48         res = x * y;
49         System.out.print("\n HRESULT : \t"+res);
50         System.out.print("\n operasi relasional numerik");
51
52         /* operasi relasional numerik */
53         TF = (i==j); System.out.print("\n TRUE/FALSE : "+TF);
54         TF = (i!=j); System.out.print("\n TRUE/FALSE : "+TF);
55         TF = (i < j); System.out.print("\n TRUE/FALSE : "+TF);
56         TF = (i > j); System.out.print("\n TRUE/FALSE : "+TF);
57         TF = (i <= j); System.out.print("\n TRUE/FALSE : "+TF);
58         TF = (i >= j); System.out.print("\n TRUE/FALSE : \t"+TF);
59         System.out.print("\n operasi relasional numerik");
60
61         /* operasi relasional numerik */
62         TF = (x != y); System.out.print("\n TRUE/FALSE : "+TF);
63         TF = (x < y); System.out.print("\n TRUE/FALSE : "+TF);
64         TF = (x > y); System.out.print("\n TRUE/FALSE : "+TF);
65         TF = (x <= y); System.out.print("\n TRUE/FALSE : "+TF);
66         TF = (x >= y); System.out.print("\n TRUE/FALSE : "+TF);
67         System.out.println(TF);
68     }
69 }

```

- Keyword

/*Contoh pengoperasian variabel bertipe dasar*/

Public class Oprator{ adalah pendeklarasian kelas dengan nama kelas **Oprator** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. **Public static void main(String[] args) {** adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static**

yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*

`boolean Bool1, Bool2, TF` ; Variable Bool1, Bool2, dan Tf memiliki tipe data boolean, yang digunakan untuk menampung nilai True atau false.

`int i,j, hsl` ; variable i, j, dan hsl bertipe data integer yang merupakan tipe data yang berbentuk bilangan bulat.

`float x,y,res;` Variable x, y, res bertipe data float merupakan tipe data untuk menyatakan pecahan, tipe data float mempunyai 4 bytes dan 32 bit dengan keakuratan ng 6-7 digit.

`System.out.println("Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output");` Menampilkan statement "Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output."

`Bool1 = true;` Nilai pada variable Bool1 akan bernilai true.

`Bool2 = false;` Nilai pada variable Bool2 akan bernilai false.

`TF = Bool1 && Bool2 ; /* Boolean AND */` Nilai dari variable TF akan dibandingkan menggunakan operator AND. Operator mengembalikan nilai true jika kedua operand bernilai sama, selain itu operator akan menghasilkan false.

`System.out.print("\n TRUE/FALSE FOR BOOLEAN AND : "+TF);` Mencetak dan menampilkan "TRUE/FALSE FOR BOOLEAN AND : ", dan mencetak dan menampilkan nilai dari variable TF

`TF = Bool1 || Bool2 ; /* Boolean OR */` Nilai dari variable TF akan dibandingkan menggunakan operator OR. Operator mengembalikan nilai true jika salah satu atau kedua kondisi bernilai benar. Jika tidak maka akan mengembalikan nilai false.

`System.out.print("\n TRUE/FALSE FOR BOOLEAN OR : "+TF);` Mencetak dan menampilkan "TRUE/FALSE FOR BOOLEAN OR : ", dan mencetak dan menampilkan nilai dari variable TF

`TF = ! Bool1 ; /* NOT */` Nilai dari variable TF akan menghasilkan nilai true apa bila kondisi logika berkebalikan atau tidak terpenuhi, jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE FOR BOOLEAN NOT : "+TF);` Mencetak dan menampilkan "TRUE/FALSE FOR BOOLEAN NOT : ", dan mencetak dan menampilkan nilai dari variable TF.

`TF = Bool1 ^Bool2; /* XOR */` Nilai dari variable TF menghasilkan nilai true jika memiliki kebenaran 1, tetapi jika kebenaran 0 maka akan menghasilkan false.

`System.out.print("\n TRUE/FALSE FOR BOOLEAN XOR : "+TF);` Mencetak dan menampilkan "TRUE/FALSE FOR BOOLEAN XOR : ", dan mencetak dan menampilkan nilai dari variable TF.

`System.out.print("\n operasi numerik");` Mencetak dan menampilkan statement "Operasi Numerik". \n untuk new line.

`i = 5;`

`j = 2 ;` Variabe i menyimpan nilai 5 dan variable j menyimpan nilai 2

`hsl = i+j;` nilai variable hsl didapatkan dari operator variable i + variable j.

`System.out.print("\n Hasil : "+hsl);` Mencetak dan menampilkan "Hasil : ", lalu mencetak dan menampilkan nilai dari variable hsl, \n untuk new line.

`hsl = i - j;` nilai variable hsl didapatkan dari operator variable i - variable j.

`System.out.print("\n Hasil : "+hsl);` Mencetak dan menampilkan "Hasil : ", lalu mencetak dan menampilkan nilai dari variable hsl, \n untuk new line.

`hsl = i / j;` nilai variable hsl didapatkan dari operator variable i/ variable j. tanda / merupakan pembagian.

`System.out.print("\n Hasil : "+hsl);` Mencetak dan menampilkan "Hasil : ", lalu mencetak dan menampilkan nilai dari variable hsl, \n untuk new line.

`hsl = i * j;` nilai variable hsl didapatkan dari operator variable i* variable j. tanda * merupakan perkalian.

`System.out.print("\n Hasil : "+hsl);` Mencetak dan menampilkan "Hasil : ", lalu mencetak dan menampilkan nilai dari variable hsl, \n untuk new line.

`hsl = i /j ; /* pembagian bulat */` nilai variable hsl didapatkan dari operator variable i/ variable j. tanda / merupakan pembagian.

`System.out.print("\n Hasil : "+hsl);` Mencetak dan menampilkan "Hasil : ", lalu mencetak dan menampilkan nilai dari variable hsl, \n untuk new line.

`hsl = i%j ; /* sisa. modulo */` Nilai dari variable hsl didapatkan dari operator variable i% variable j. tanda % merupakan sisa.

`System.out.print("\n Hasil : "+hsl);` Mencetak dan menampilkan "Hasil : ", lalu mencetak dan menampilkan nilai dari variable hsl, \n untuk new line.

`System.out.print("\n operasi numerik");` Mencetak dan menampilkan "Operasi Numerk ", \n untuk new line.

`x = 5 ;`

`y = 5 ;` variable x menyimpan nilai 5 dan variable y menyimpan nilai 5.

`res = x + y;` nilai dari variable res didapatkan dari operator variabel x + variable y.

`System.out.print("\n RESULT : "+res);` Mencetak dan menampilkan "Result: ", lalu mencetak dan menampilkan nilai dari variable res, \n untuk new line.

`res = x - y;` nilai dari variable res didapatkan dari operator variable x – variable y

`System.out.print("\n RESULT: "+res);` Mencetak dan menampilkan "Result: ", lalu mencetak dan menampilkan nilai dari variable res, \n untuk new line.

`res = x / y;` nilai dari variable res didapatkan dari operator variable x / variable y, tanda / untuk pembagian.

`System.out.print("\n RESULT : "+res);` Mencetak dan menampilkan "Result: ", lalu mencetak dan menampilkan nilai dari variable res, \n untuk new line.

`res = x * y;` nilai dari variable res didapatkan dari operator variable x * variable y, tanda * digunakan untuk perkalian.

`System.out.print("\n HRESULT : \t"+res);` Mencetak dan menampilkan "Result: ", lalu mencetak dan menampilkan nilai dari variable res, \n untuk new line.

`System.out.print("\n operasi relasional numerik");` Mencetak dan menampilkan "Operasi relasional numerik : ", \n untuk new line.

`TF = (i==j);` Nilai dari TF didapatkan dari operator perbandingan antara i = j, hasil nilai true didapatkan jika kedua kondisi sama, tetapi jika salah satu kondisi berbeda maka akan bernilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (i!=j);` Nilai dari TF didapatkan dari operator perbandingan antara i != j, hasil nilai true didapatkan jika kedua kondisi tidak sama, tetapi jika kedua kondisi sama maka akan bernilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (i < j);` Nilai dari variable TF didapatkan dari operator perbandingan i < j, hasil nilai true didapatkan jika benar kondisi i kurang dari j, jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (i > j);` Nilai dari variable TF didapatkan dari operator perbandingan i > j, hasil nilai true didapatkan jika benar kondisi i lebih besar dari j, jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (i <= j);` Nilai dari variable TF didapatkan dari operator perbandingan i <= j, hasil nilai true didapatkan jika benar kondisi i kurang dari sama dengan j, jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF=(i >= j);` Nilai dari variable TF didapatkan dari operator perbandingan $i \geq j$, hasil nilai true didapatkan jika benar kondisi i lebih besar dari sama dengan j , jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : \t"+TF);` Mencetak dan menampilkan statement "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`System.out.print("\n operasi relasional numerik");` Mencetak dan menampilkan statement "Operasi relasional numerik", \n untuk new line.

`TF = (x != y);` Nilai TF didapatkan dari $x \neq y$ akan menghasilkan nilai true apa bila kondisi logika berkebalikan atau tidak terpenuhi, jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan statement "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (x < y);` Nilai dari variable TF didapatkan dari operator perbandingan $x < y$, hasil nilai true didapatkan jika benar kondisi x kurang dari y , jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan statement "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (x > y);` Nilai dari variable TF didapatkan dari operator perbandingan $x > y$, hasil nilai true didapatkan jika benar kondisi x lebih dari y , jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan statement "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (x <= y);` Nilai dari variable TF didapatkan dari operator perbandingan $x \leq y$, hasil nilai true didapatkan jika benar kondisi x lebih kurang dari sama dengan y , jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan statement "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

`TF = (x >= y);` Nilai dari variable TF didapatkan dari operator perbandingan $i \geq j$, hasil nilai true didapatkan jika benar kondisi i lebih besar dari sama dengan j , jika tidak maka akan menghasilkan nilai false.

`System.out.print("\n TRUE/FALSE : "+TF);` Mencetak dan menampilkan statement "True/False: ", lalu mencetak dan menampilkan nilai dari variable TF, \n untuk new line.

- Outputnya

```

Command Prompt
D:\PBO java\Tugas>javac Oprator.java

D:\PBO java\Tugas>java Oprator
Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output
TRUE/FALSE FOR BOOLEAN AND : false
TRUE/FALSE FOR BOOLEAN OR : true
TRUE/FALSE FOR BOOLEAN NOT : false
TRUE/FALSE FOR BOOLEAN XOR : true
operasi numerik
Hasil : 7
Hasil : 3
Hasil : 2
Hasil : 10
Hasil : 2
Hasil : 1
operasi numerik
RESULT : 10.0
RESULT : 0.0
RESULT : 1.0
HRESULT : 25.0
operasi relasional numerik
TRUE/FALSE : false
TRUE/FALSE : true
TRUE/FALSE : false
TRUE/FALSE : true
TRUE/FALSE : false
TRUE/FALSE : true
operasi relasional numerik
TRUE/FALSE : false
TRUE/FALSE : false
TRUE/FALSE : false
TRUE/FALSE : true
TRUE/FALSE : true

```

2. Kode program

- Listing 1

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 23 Maret 2022
5  Waktu    : 04.34 WITA
6  */
7
8  import java.io.BufferedReader;
9  import java.io.IOException;
10 import java.io.InputStreamReader;
11 import javax.swing.*;
12 public class BacaString {
13     /**
14      * @param args
15      * @throws IOException
16      */
17     public static void main(String[] args) throws IOException {
18         // TODO Auto-generated method stub
19         /* Kamus */
20         String str;
21         BufferedReader datAIn = new BufferedReader(new
22             InputStreamReader(System.in));
23         /* Program */
24         System.out.print("\nBaca string dan Integer: \n");
25         System.out.print("masukkan sebuah string: ");
26         str= datAIn.readLine();
27         System.out.print ("String yang dibaca : "+ str);
28     }
29 }

```

- Keyword

`import java.io.BufferedReader;` impor scanner agar diketahui method-method yang digunakan pada Scanner.

`import java.io.IOException;`

`import java.io.InputStreamReader;`

`import javax.swing.*;`

`public class BacaString {` adalah pendeklarasian kelas dengan nama kelas **BacaString** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*

`String str;` tipe data string dengan nama str.

`BufferedReader datAIn = new BufferedReader (new InputStreamReader(System.in));`

`Public static void main (string[] args){` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan.

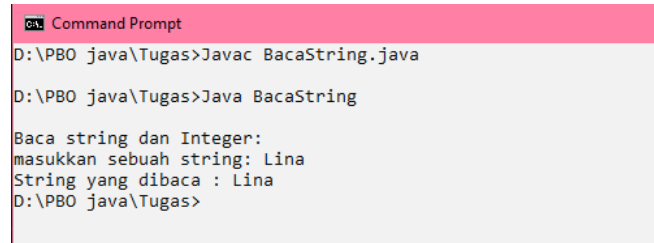
`System.out.print ("\nBaca string dan Integer: \n");` akan mengeluarkan atau memanggil "Baca string dan Integer". Diikuti dengan tanda \n yang berarti baris baru.

`System.out.print("masukkan sebuah string: ");` adalah sebuah pernyataan (statement) yang memanggil *method print* dimana fungsi dari *method* ini adalah untuk menampilkan kata "masukan sebuah string: ".

`str= datAIn.readLine();` variabel str dengan rumus sebuah fungsi.

`System.out.print ("String yang dibaca : "+ str);` adalah sebuah pernyataan (statement) yang memanggil *method print* dimana fungsi dari *method* ini adalah untuk menampilkan kata "String yang dibaca: " dari str.

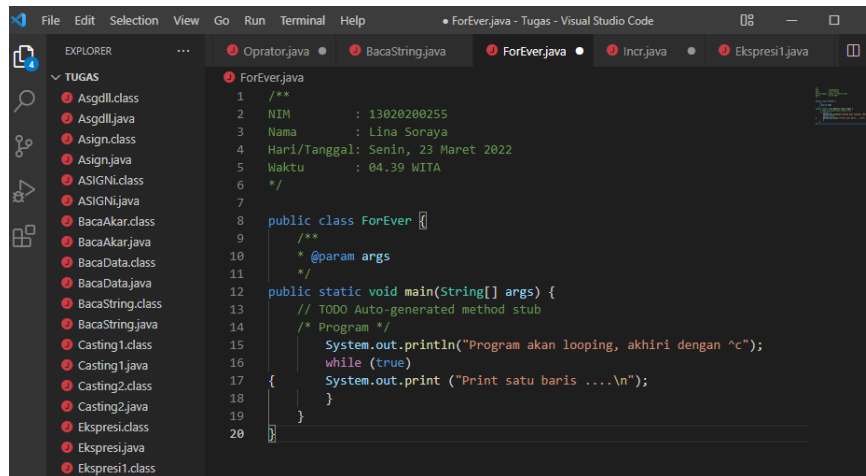
- **Output**



```
C:\ Command Prompt
D:\PBO java\Tugas>javac BacaString.java
D:\PBO java\Tugas>java BacaString

Baca string dan Integer:
masukkan sebuah string: Lina
String yang dibaca : Lina
D:\PBO java\Tugas>
```

- Listing program 2



- Keyword

Public class ForEver { adalah pendeklarasian kelas dengan nama kelas **For Ever** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

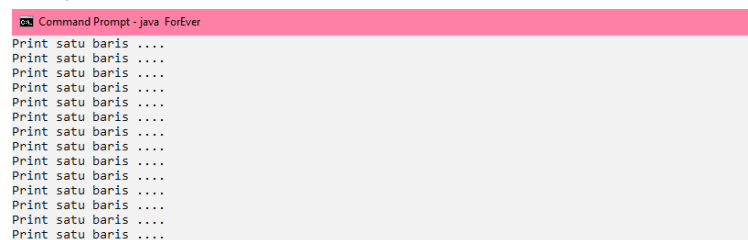
Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan.

`System.out.println("Program akan looping, akhiri dengan ^c");` adalah sebuah pernyataan (statement) yang memanggil *method* **println** dimana fungsi dari *method* ini adalah untuk menampilkan “Program akanlooping”. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris).

while (true) jika benar maka

`System.out.print ("Print satu baris\n");` akan menampilkan "Print satu baris..." sampai diakhiri dengan `^c`.

- Output



- Listing 3

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 23 Maret 2022
5  Waktu    : 04.44 WITA
6  */
7
8  /* contoh pemakaian IF satu kasus */
9  /* membaca nilai integer, menuliskan nilainya jika positif */
10 import java.util.Scanner;
11 public class If1 {
12     /**
13      * @param args
14      */
15     public static void main(String[] args) {
16         // TODO Auto-generated method stub
17         /* Kamus */
18         Scanner masukan=new Scanner(System.in);
19         int a;
20         /* Program */
21         System.out.print ("Contoh IF satu kasus \n");
22         System.out.print ("Ketikkan suatu nilai integer : ");
23         a = masukan.nextInt();
24         if (a >= 0)
25             System.out.print ("\nNilai a positif "+ a);
26     }
27 }

```

- Keyword

* contoh pemakaian IF satu kasus */

/* membaca nilai integer, menuliskan nilainya jika positif */

import java.util.Scanner; import kelas Scanner untuk mengenali method=method yang ada pada kelas Scanner.

public class If1 { adalah pendeklarasian kelas dengan nama kelas **If1** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan.

Scanner masukan=new Scanner(System.in);

int a; tipe data integer yang berisi variabel a tanpa masukkan nilai. Integer adalah tipe data yang menampung bilangan bulat.

System.out.print ("Contoh IF satu kasus \n"); menampilkan apa yang ada di dalam *method*.

System.out.print ("Ketikkan suatu nilai integer : "); menampilkan isi yang ada di dalam *method*.

a = masukan.nextInt(); variabel a dengan masukkan nexInt

if (a >= 0) jika a lebih besar sama dengan nol, maka

System.out.print ("\nNilai a positif "+ a); akan dikeluarkan Nilai a positif, dari a yang dimasukkan.

}

}

- Output

```

Command Prompt

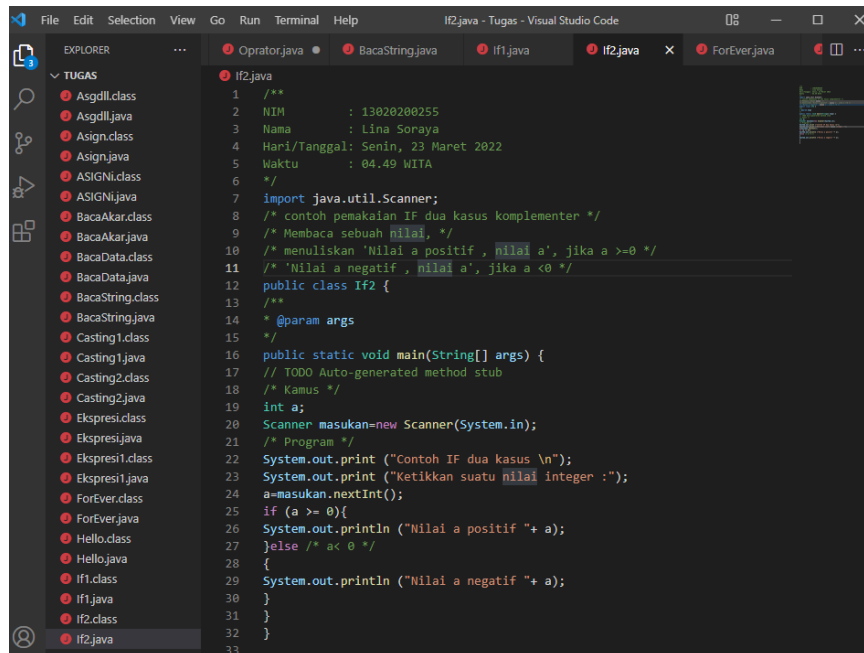
D:\PBO java\Tugas>javac If1.java

D:\PBO java\Tugas>java If1
Contoh IF satu kasus
Ketikkan suatu nilai integer : 22

Nilai a positif 22
D:\PBO java\Tugas>

```

- Listing 4



- Keyword

`import java.util.Scanner;`

`/* contoh pemakaian IF dua kasus komplementer */`

`/* Membaca sebuah nilai, */`

`/* menuliskan 'Nilai a positif , nilai a', jika a >=0 */`

`/* 'Nilai a negatif , nilai a', jika a <0 */`

`public class If2 {` keyword `public` digunakan untuk mengakses seluruh kelas. Keyword `class` digunakan untuk mendefinisikan dan mengimplementasikan objek, dan lalu diikuti dengan pemberian nama kelas yaitu "If2".

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu

parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int a;` bilangan dengan tipe data integer yang memiliki variabel `a` tanpa masukkan nilai.

`Scanner masukan=new Scanner(System.in);` kode untuk menginstansi objek Scanner

`System.out.print ("Contoh IF dua kasus \n");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan “Contoh IF dua kasus”, diikuti dengan `\n` untuk perintah baris baru.

`System.out.print ("Ketikkan suatu nilai integer :");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan kata “Ketikkan suatu nilai integer”.

`a=masukan.nextInt();` variabel `a` yang menyimpan masukan dari keyboard.

`if (a >= 0){` jika nilai dari variabel `a` lebih besar sama dengan nol, maka akan di eksekusi dan

`System.out.println ("Nilai a positif "+ a);` ditampilkan “Nilai a positif” dari variabel `a`.

}

`else /*a< 0*/` jika `a` lebih kecil dari 0, maka

{

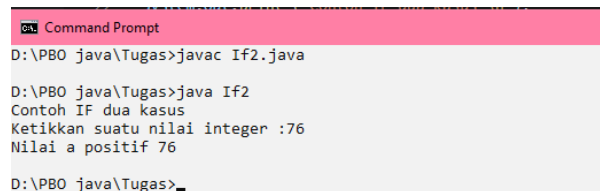
`System.out.println ("Nilai a negatif "+ a);` akan ditampilkan “Nilai a negatif” dari variabel `a` yang diinputkan.

}

}

}

- **Output**



```
Command Prompt
D:\PBO java\Tugas>javac If2.java

D:\PBO java\Tugas>java If2
Contoh IF dua kasus
Ketikkan suatu nilai integer :76
Nilai a positif 76

D:\PBO java\Tugas>
```

- Listing 5

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Senin, 23 Maret 2022
5  Waktu    : 04.53 WITA
6  */
7  import java.util.Scanner;
8  /* contoh pemakaian IF tiga kasus */ /* Membaca sebuah nilai, */
9  /* menuliskan 'Nilai a positif, nilai a', jika a >0 */
10 /* 'Nilai Nol, nilai a', jika a = 0 */
11 /* 'Nilai a negatif, nilai a', jika a <0 */
12 public class If3 {
13     /**
14     * @param args
15     */
16     public static void main(String[] args) {
17         // TODO Auto-generated method stub
18         /* Kamus */
19         int a;
20         Scanner masukan=new Scanner(System.in);
21         /* Program */
22         System.out.print ("Contoh IF tiga kasus \n");
23         System.out.print ("Ketikkan suatu nilai integer :");
24         a=masukan.nextInt();
25         if (a > 0){
26             System.out.println ("Nilai a positif "+ a);
27         }else if (a == 0){
28             System.out.println ("Nilai Nol "+ a);
29         }else /* a < 0 */ {
30             System.out.println ("Nilai a negatif "+ a);
31         }
32     }
33 }
34

```

- Keyword

`import java.util.Scanner;`
 /* contoh pemakaian IF tiga kasus */
 /* Membaca sebuah nilai, */
 /* menuliskan 'Nilai a positif, nilai a', jika a >0 */
 /* 'Nilai Nol, nilai a', jika a = 0 */
 /* 'Nilai a negatif, nilai a', jika a <0 */

`public class If3 {` keyword public digunakan untuk mengakses seluruh kelas. Keyword class digunakan untuk mendefinisikan dan mengimplementasikan objek, dan lalu diikuti dengan pemberian nama kelas yaitu "If3".

`Public static void main(String[] args) {` public dapat diartikan bahwa argumen main () merupakan method yang bersifat dapat diakses dan dapat dipanggil dimana saja. Void merupakan tipe data yang menyatakan bahwa kode deklarasi tidak memerlukan return artinya method main tidak memerlukan nilai return. String menyatakan kumpulan char atau karakter yang terdapat huruf, angka, dan beberapa simbol. String args[] merupakan array dengan tipe data string . mengandung array. Tanpa string arg [] program tidak dapat dieksekusi.

`int a;` sebuah bilangan dengan tipe data integer dan memiliki variabel a tanpa masukkan nilai.

`Scanner masukan=new Scanner(System.in);` kode program yang bertujuan untuk menginstansi objek pada Scanner.

`System.out.print ("Contoh IF tiga kasus \n");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan "Contoh IF tiga kasus". Diikuti dengan `\n` untuk perintah baris baru.

`System.out.print ("Ketikkan suatu nilai integer :");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan kata "Ketikan suatu nilai integer".

`a=masukan.nextInt();` Variabel *a* yang memiliki dan menyimpan sebuah kode program yang akan di jalankan pada saat di panggil pada cmd.

`if (a > 0){` jika nilai dari variabel *a* lebih besar dari nol, maka

`System.out.println ("Nilai a positif "+ a);` akan ditampilkan "Nilai a positif" dari variabel *a* yang dimasukkan.

`}else if (a == 0){` tapi, jika nilai dari variabel *a* sama dengan nol, maka

`System.out.println ("Nilai Nol "+ a);` cmd akan menampilkan "Nilai Nol" dari masukkan variabel *a*.

`}else /*a > 0*/{` jika *a* lebih besar dari nol, maka akan dieksekusi dan menampilkan

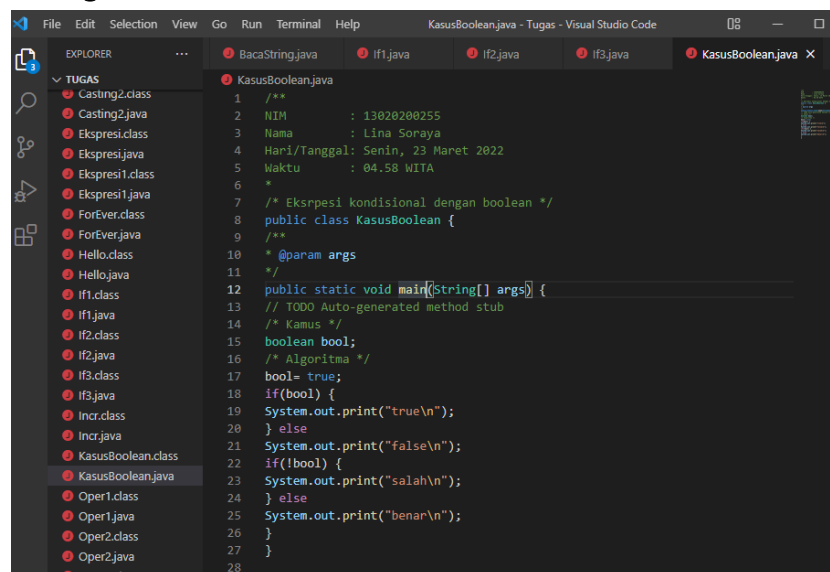
`System.out.println ("Nilai a negatif "+ a);` menampilkan 'Nilai a negatif'.

- **Output 6**



```
Command Prompt
D:\PBO java\Tugas>javac If3.java
D:\PBO java\Tugas>java If3
Contoh IF tiga kasus
Ketikkan suatu nilai integer :-76
Nilai a negatif -76
```

- **Listing**



```
KasusBoolean.java
1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Senin, 23 Maret 2022
5   * Waktu    : 04.58 WITA
6   *
7   * Eksrpesi kondisional dengan boolean */
8  public class KasusBoolean {
9      /**
10     * @param args
11     */
12     public static void main(String[] args) {
13         // TODO Auto-generated method stub
14         /* Kasus */
15         boolean bool;
16         /* Algoritma */
17         bool= true;
18         if(bool) {
19             System.out.print("true\n");
20         } else
21             System.out.print("false\n");
22         if(!bool) {
23             System.out.print("salah\n");
24         } else
25             System.out.print("benar\n");
26     }
27 }
28
```

- **Keyword**

`/* Eksrpesi kondisional dengan boolean */`

`public class KasusBoolean {` keyword `public` digunakan untuk mengakses seluruh kelas. Keyword `class` digunakan untuk mendefinisikan dan mengimplementasikan objek, dan lalu diikuti dengan pemberian nama kelas yaitu "KasusBoolean".

`Public static void main(String[] args) {` `public` dapat diartikan bahwa argumen `main ()` merupakan method yang bersifat dapat diakses dan dapat dipanggil dimana saja. `Void` merupakan tipe data yang menyatakan bahwa kode deklarasi tidak memerlukan return artinya method `main` tidak memerlukan nilai return. `String` menyatakan kumpulan char atau karakter yang terdapat huruf, angka, dan beberapa simbol. `String args[]` merupakan array dengan tipe data `string` . mengandung array. Tanpa `string arg []` program tidak dapat dieksekusi.

`boolean bool;` sebuah inisialisasi parameter.

`bool= true;` nilai dari `Boolean` sama dengan `true`, atau nilai `true` menuju ke `Boolean`.

`if(bool) {` kondisi dimana jika `Boolean` bernilai `true` maka akan ditampilkan `true` bernilai benar.

`System.out.print("true\n");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan inputan dari "Ketikan suatu nilai integer", yang diikuti dengan `\n` yang menandakan adanya baris baru.

`} else` jika nilai `Boolean` bukan `true`, maka

`System.out.print("false\n");` akan ditampilkan "false"

`if(!bool) {` kondisi dimana tidak sama dengan `Boolean` dan tidak bernilai `true`.

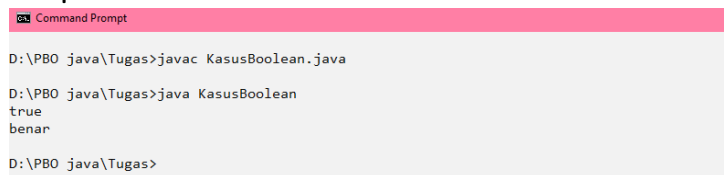
`System.out.print("salah\n");` maka akan ditampilkan "salah".

`} else` tapi jika kondisi sama dengan `Boolean` dan bernilai `true`

`System.out.print("benar\n");` maka akan dieksekusi dan ditampilkan "benar".

}

- **Output**



```
Command Prompt
D:\PBO java\Tugas>javac KasusBoolean.java
D:\PBO java\Tugas>java KasusBoolean
true
benar
D:\PBO java\Tugas>
```

- Listing 7

```

1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Senin, 23 Maret 2022
5   * Waktu    : 04.49 WITA
6   */
7
8  import java.util.Scanner;
9  public class KasusSwitch {
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO: Auto-generated method stub
15         /* Kamus */
16         char cc;
17         Scanner masukan=new Scanner(System.in);
18         /* Program */
19         System.out.print ("Ketikkan sebuah huruf, akhiri dengan RETURN \n");
20         cc=masukan.next().charAt(0);
21         switch (cc) {
22             case 'a': { System.out.print (" Yang anda ketik adalah a \n"); break;}
23             case 'u': { System.out.print (" Yang anda ketik adalah u \n"); break;}
24             case 'e': { System.out.print (" Yang anda ketik adalah e \n"); break;}
25             case 'i': { System.out.print (" Yang anda ketik adalah i \n"); break;}
26             case 'o': { System.out.print (" Yang anda ketik adalah o\n"); break;}
27             default:
28                 System.out.print (" Yang anda ketik adalah huruf mati \n");
29             }
30         }
31     }

```

- Keyword

`import java.util.Scanner;`

`public class KasusSwitch {` adalah pendeklarasian kelas dengan nama kelas **KasusSwitsch** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`char cc;` sebuah tipe data char dengan variabel cc. Tipe data char dalam bahasa Java dipakai untuk menampung 1 digit karakter, baik itu berupa huruf, angka maupun karakter lain seperti ^, %, dan #.

`Scanner masukan=new Scanner(System.in);` kode yang menginstansi objek Scanner.

`System.out.print ("Ketikkan sebuah huruf, akhiri dengan RETURN \n");` adalah sebuah pernyataan (statement) yang memanggil *method print* dimana fungsi dari *method* ini adalah untuk menampilkan inputan “Ketikan sebuah huruf, akhiri dengan RETURN”, yang diikuti dengan `\n` yang menandakan adanya baris baru.

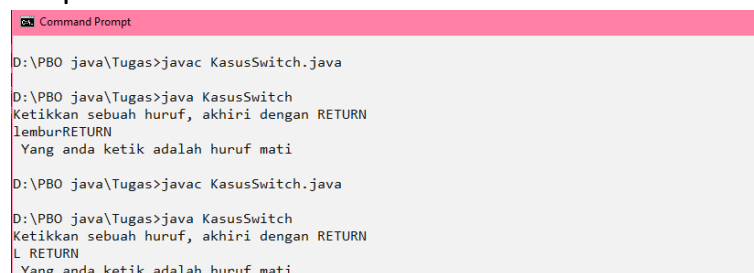
`cc=masukan.next().charAt(0);` variabel cc dengan sebuah kode program `masukan.next().charAt(0)` yang nantinya akan dieksekusi ketika dipanggil.

```

switch (cc) { perintah operasi struktur percabangan Swich-Case untuk hasil inputan
yang memenuhi kondisi akan menampilkan rentetean inputan yang berada da;am case.
case 'a': {
System.out.print (" Yang anda ketik adalah a \n"); menampilkan “Yang anda ketik
adalah masukkan dari case a.
break;
}
case 'u': {
System.out.print (" Yang anda ketik adalah u \n"); menampilkan “Yang anda ketik
adalah masukkan dari case u.
break;
}
case 'e': {
System.out.print (" Yang anda ketik adalah e \n"); menampilkan “Yang anda ketik
adalah masukkan dari case e.
break;
}
case 'i': {
System.out.print (" Yang anda ketik adalah i \n"); menampilkan “Yang anda ketik adalah
masukkan dari case i.
break;
}
case 'o': {
System.out.print (" Yang anda ketik adalah o \n"); menampilkan “Yang anda ketik
adalah masukkan dari case o.
break;
}
default: jika kondisi memenuhi maka akan menampilkan output dibawah.
System.out.print (" Yang anda ketik adalah huruf mati \n");

```

- **Output**



```

Command Prompt

D:\PBO java\Tugas>javac KasusSwitch.java

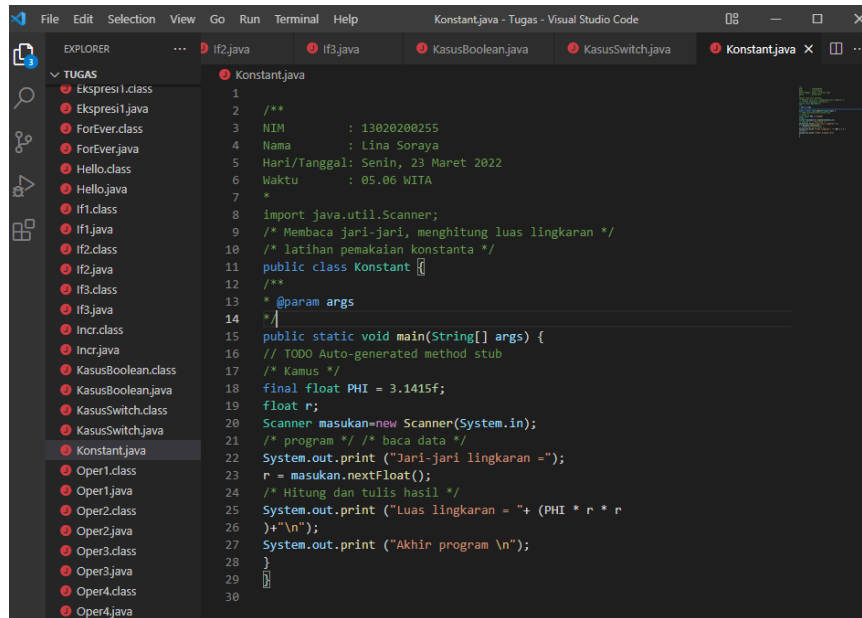
D:\PBO java\Tugas>java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN
lemburRETURN
Yang anda ketik adalah huruf mati

D:\PBO java\Tugas>javac KasusSwitch.java

D:\PBO java\Tugas>java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN
L RETURN
Yang anda ketik adalah huruf mati

```

- **Listing 8**



```
1
2  /**
3   * NIM      : 13020200255
4   * Nama     : Lina Soraya
5   * Hari/Tanggal: Senin, 23 Maret 2022
6   * Waktu    : 05.06 WITA
7   */
8  import java.util.Scanner;
9  /* Membaca jari-jari, menghitung luas lingkaran */
10 /* latihan pemakaian konstanta */
11 public class Konstant {
12     /**
13      * @param args
14      */
15     public static void main(String[] args) {
16         // TODO Auto-generated method stub
17         /* Kamus */
18         final float PHI = 3.1415f;
19         float r;
20         Scanner masukan=new Scanner(System.in);
21         /* program */ /* baca data */
22         System.out.print ("Jari-jari lingkaran =");
23         r = masukan.nextFloat();
24         /* Hitung dan tulis hasil */
25         System.out.print ("Luas lingkaran = "+ (PHI * r * r
26         )+"\n");
27         System.out.print ("Akhir program \n");
28     }
29 }
30
```

- **Keyword**

`import java.util.Scanner;`

`/* Membaca jari-jari, menghitung luas lingkaran */`

`/* latihan pemakaian konstanta */`

`public class Konstant {` adalah pendeklarasian kelas dengan nama kelas **Konstan** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`final float PHI = 3.1415f;` deklarasi variabel PHI dengan masukkan nilai 3.1415f. kde f di belakang angka menandakan bahwa nilai tersebut adalah nilai dari bilangan float.

`float r;` deklarasi tipe data float dengan variabel r. float adalah tipe data yang menyimpan bilangan pecahan.

`Scanner masukan=new Scanner(System.in);` scanner dengan nama masukan untuk membuat objek scanner yang baru dengan nama System.in

`/* program */ /* baca data */`

`System.out.print ("Jari-jari lingkaran =");` menampilkan keluaran dari masukan "Jari-jari lingkaran".

`r = masukan.nextFloat();` variabel r dengan hitungan nilai didalam float.

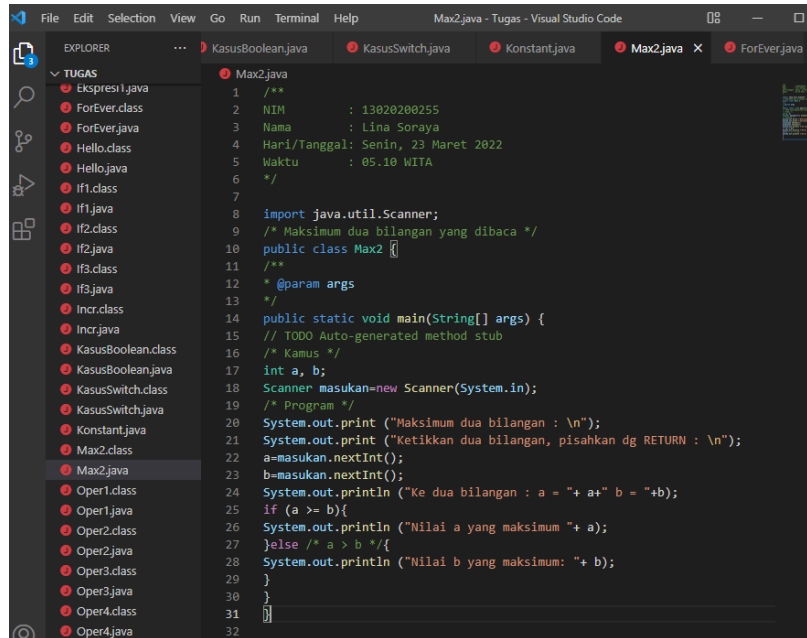
```
/* Hitung dan tulis hasil */
```

`System.out.print ("Luas lingkaran = "+ (PHI * r * r)+"\n");` menampilkan output dari inputan "Luas lingkaran" dari nilai PHI x r^2 dengan memasukkan dari nilai r tadi.

`System.out.print ("Akhir program \n");` dan akan ditampilkan "akhir program".

```
}  
}
```

- Listing 9



```
1  /**  
2  : 13020200255  
3  Nama : Lina Soraya  
4  Hari/Tanggal: Senin, 23 Maret 2022  
5  Waktu : 05.10 WITA  
6  */  
7  
8  import java.util.Scanner;  
9  /* Maksimum dua bilangan yang dibaca */  
10 public class Max2 {  
11     /**  
12     * @param args  
13     */  
14     public static void main(String[] args) {  
15         // TODO Auto-generated method stub  
16         /* Kamus */  
17         int a, b;  
18         Scanner masukan=new Scanner(System.in);  
19         /* Program */  
20         System.out.print ("Maksimum dua bilangan : \n");  
21         System.out.print ("Ketikkan dua bilangan, pisahkan dg RETURN : \n");  
22         a=masukan.nextInt();  
23         b=masukan.nextInt();  
24         System.out.println ("Ke dua bilangan : a = "+ a+" b = "+b);  
25         if (a >= b){  
26             System.out.println ("Nilai a yang maksimum "+ a);  
27         }else /* a > b */{  
28             System.out.println ("Nilai b yang maksimum: "+ b);  
29         }  
30     }  
31 }  
32
```

- Keyword

`import java.util.Scanner;`

`/* Maksimum dua bilangan yang dibaca */`

`public class Max2 {` adalah pendeklarasian kelas dengan nama kelas **Max** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Public static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int a, b;` sebuah pendeklarasian variabel a dan b tanpa masukkan nilai.

`Scanner masukan=new Scanner(System.in);` scanner dengan nama masukan untuk membuat objek scanner yang baru dengan nama `System.in`

`/* Program */`

`System.out.print ("Maksimum dua bilangan : \n");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan inputan dari “Maksimum dua bilangan”, yang diikuti dengan `\n` yang menandakan adanya baris baru.

`System.out.print ("Ketikkan dua bilangan, pisahkan dg RETURN : \n");` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan “Ketikkan dua bilangan, pisahkan dengan RETURN”, yang diikuti dengan `\n` yang menandakan adanya baris baru.

`a=masukan.nextInt();` menginputkan nilai a lalu disimpan.

`b=masukan.nextInt();` menginputkan nilai b.

`System.out.println ("Ke dua bilangan : a = "+ a+" b = "+b);` adalah sebuah pernyataan (statement) yang memanggil *method* **print** dimana fungsi dari *method* ini adalah untuk menampilkan inputan dari “Ke dua bilangan: a”.

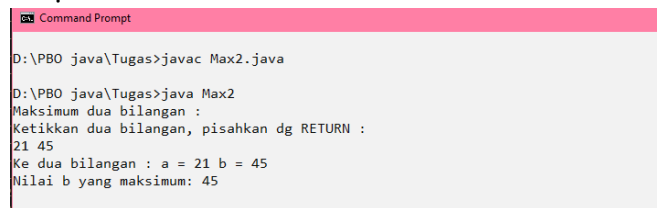
`if (a >= b){` jika nilai dari variabel a lebih besar sma dengan nilai variabel b.

`System.out.println ("Nilai a yang maksimum "+ a);` maka yang akan ditampilkan dan yang akan menjadi nilai maksimum adalah a.

`}else /*a > b*/ {` tapi jika nilai a lebih bessar daripada b.

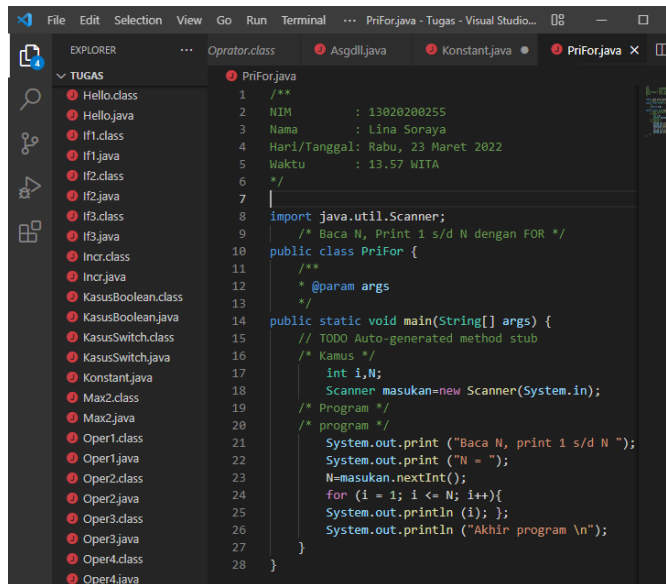
`System.out.println ("Nilai b yang maksimum: "+ b);` yang akan ditampilkan adalah nilai dari variabel b yang merupakan nilai maksimum.

- **Output**



```
Command Prompt
D:\PBO java\Tugas>javac Max2.java
D:\PBO java\Tugas>java Max2
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
21 45
Ke dua bilangan : a = 21 b = 45
Nilai b yang maksimum: 45
```

- **Listing 10**



```
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Rabu, 23 Maret 2022
5  Waktu    : 13.57 WITA
6  */
7
8  import java.util.Scanner;
9  /* Baca N, Print 1 s/d N dengan FOR */
10 public class PriFor {
11     /**
12     * @param args
13     */
14     public static void main(String[] args) {
15         // TODO Auto-generated method stub
16         /* Kamus */
17         int i,N;
18         Scanner masukan=new Scanner(System.in);
19         /* Program */
20         /* program */
21         System.out.print ("Baca N, print 1 s/d N ");
22         System.out.print ("N = ");
23         N=masukan.nextInt();
24         for (i = 1; i <= N; i++){
25             System.out.println (i);
26             System.out.println ("Akhir program \n");
27         }
28     }
```

- **Keyword**

`import java.util.Scanner;`

`/* Baca N, Print 1 s/d N dengan FOR */`

`public class PriFor {` adalah pendeklarasian kelas dengan nama kelas **PriFor** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int i,N;` pendeklarasian tipe data integer dengan variabel i dan N.

`Scanner masukan=new Scanner(System.in);` scanner dengan nama masukan untuk membuat objek scanner yang baru dengan nama System.in

`System.out.print ("Baca N, print 1 s/d N ");` mencetak "Baca N, print 1 s/d N".

`System.out.print ("\n N = ");` dengan sebuah keyword `\b` menandakan bahwa ada baris baru. Statement ini akan menampilkan "N".

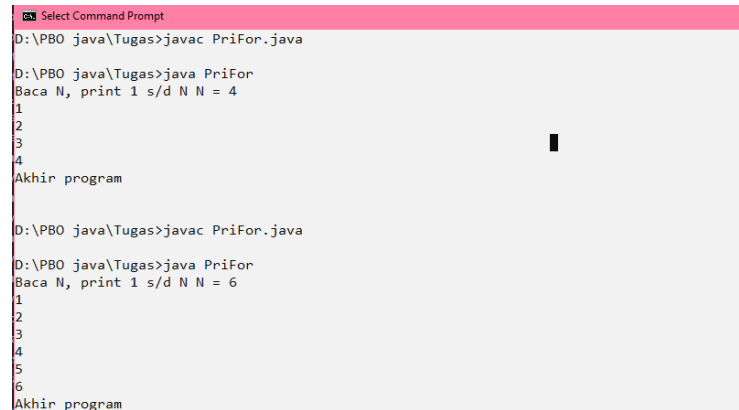
`N=masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel N dan menyimpan masukan.

`for (i = 1; i <= N; i++){` sebuah perulangan dimana pengekskusion pertama dilakukan pada nilai i=1, apakah nilai i sama dengan 1? Jika iya maka lanjut ke espresi selanjutnya apakah i<N? Kemudian nilai i kan terus bertambah 1 sampai proses perulangan selesai.

`System.out.println (i); }` ditampilkan nilai dari i.

`System.out.println ("Akhir program \n");` menampilkan “Akhir program”.

- **Outputnya**



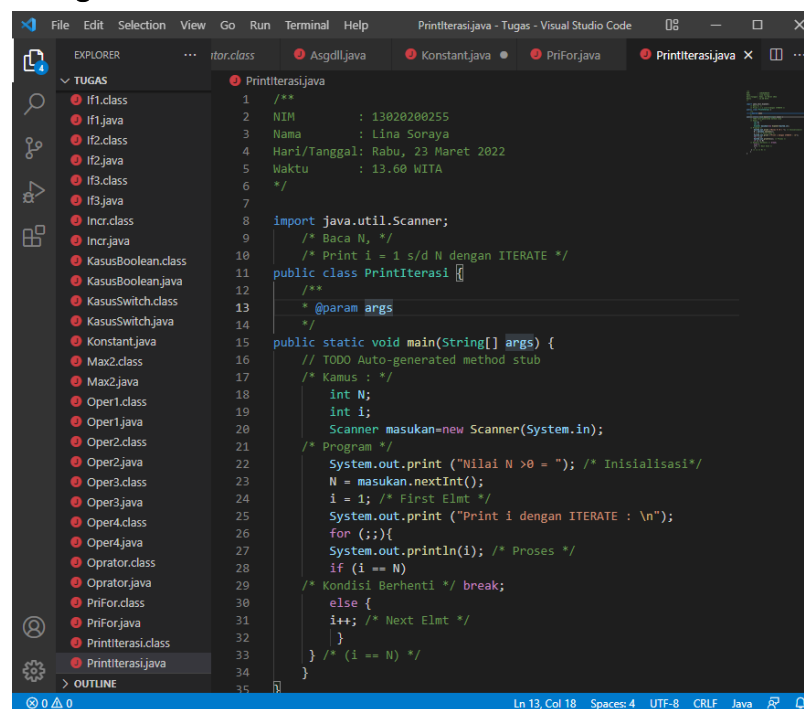
```
Select Command Prompt
D:\PBO java\Tugas>javac PriFor.java

D:\PBO java\Tugas>java PriFor
Baca N, print 1 s/d N N = 4
1
2
3
4
Akhir program

D:\PBO java\Tugas>javac PriFor.java

D:\PBO java\Tugas>java PriFor
Baca N, print 1 s/d N N = 6
1
2
3
4
5
6
Akhir program
```

- **Listing 11**



```
File Edit Selection View Go Run Terminal Help PrintIterasi.java - Tugas - Visual Studio Code
EXPLORER
TUGAS
  If1.class
  If1.java
  If2.class
  If2.java
  If3.class
  If3.java
  Incr.class
  Incr.java
  KasusBoolean.class
  KasusBoolean.java
  KasusSwitch.class
  KasusSwitch.java
  Konstant.java
  Konstant.java
  Max2.class
  Max2.java
  Oper1.class
  Oper1.java
  Oper2.class
  Oper2.java
  Oper3.class
  Oper3.java
  Oper4.class
  Oper4.java
  Operator.class
  Operator.java
  PriFor.class
  PriFor.java
  PrintIterasi.class
  PrintIterasi.java
  > OUTLINE

PrintIterasi.java
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Rabu, 23 Maret 2022
5  Waktu    : 13.00 WITA
6  */
7
8  import java.util.Scanner;
9
10 /* Baca N, */
11 /* Print i = 1 s/d N dengan ITERATE */
12 public class PrintIterasi {
13     /**
14      * @param args
15      */
16     public static void main(String[] args) {
17         // TODO Auto-generated method stub
18         /* Kamus : */
19         int N;
20         int i;
21         Scanner masukan=new Scanner(System.in);
22
23         /* Program */
24         System.out.print ("Nilai N >0 = "); /* Inisialisasi*/
25         N = masukan.nextInt();
26         i = 1; /* First Elmt */
27         System.out.print ("Print i dengan ITERATE : \n");
28         for (;){
29             System.out.println(i); /* Proses */
30             if (i == N)
31                 /* Kondisi Berhenti */ break;
32             else {
33                 i++; /* Next Elmt */
34             }
35         } /* (i == N) */
36     }
37 }
```

- **Tujuan**

Bertujuan untuk menampilkan hasil pemograman dari ITERASI dimana kondisi awal dari program akan diperiksa, jika kondisi tidak sma dengan yang diawal maka akan dilakukan ekseusi selanjutnya.

- **Keyword**

```
import java.util.Scanner;
/* Baca N, */
/* Print i = 1 s/d N dengan ITERATE */
```

`public class PrintIterasi {` adalah pendeklarasian kelas dengan nama kelas **PrintIterasi** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. `Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int N;` variabel N dengan tipe data integer.

`int i;` variabel I dengan tipe data integer juga

`Scanner masukan=new Scanner(System.in);`

`System.out.print ("Nilai N >0 = ");` memanggil print dari masukan yang ada di method.

`N = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel N dan menyimpan masukan.

`i = 1;` variabel i dengan masukan nilai 1.

`System.out.print ("Print i dengan ITERATE : \n");` memanggil print dari masukan yang ada di method.

`for (;;) {`

`System.out.println(i);` memanggil print dari masukan yang ada di method yaitu nilai dari i

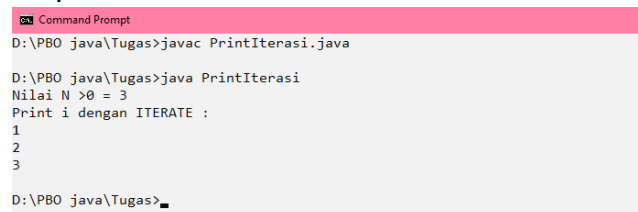
`if (i == N)` jika nilai i sama dengan nilai N maka program akan keluar dari perulangan.

`break;`

`else {` jika nilai i tidak sama dengan nilai N maka program akan terus melakukan perulangan.

`i++;` dan nilai i akan terus bertambah 1 sampai batak maks perulangan yang telah ditentukan.

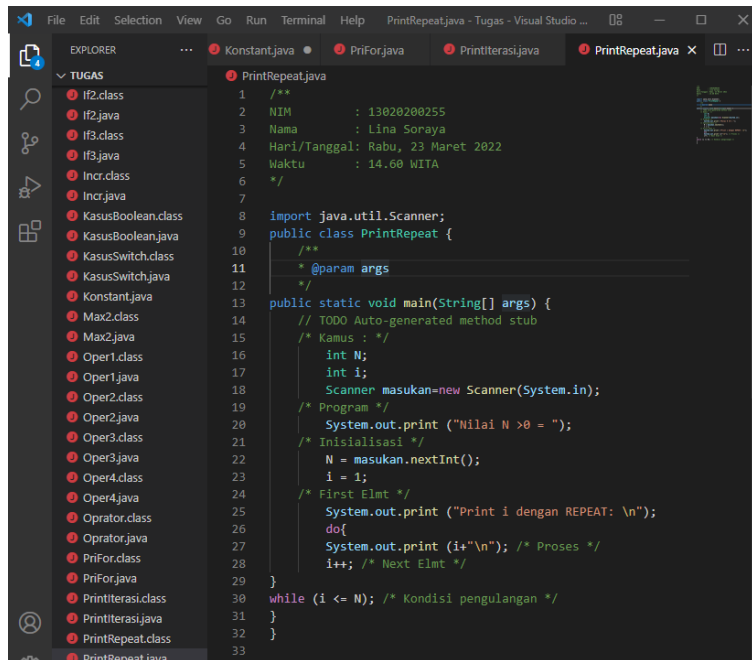
- **Output**



```
Command Prompt
D:\PBO java\Tugas>javac PrintIterasi.java

D:\PBO java\Tugas>java PrintIterasi
Nilai N >0 = 3
Print i dengan ITERATE :
1
2
3
D:\PBO java\Tugas>
```

- Listing 12



```

1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Rabu, 23 Maret 2022
5   * Waktu    : 14.00 WITA
6   */
7
8  import java.util.Scanner;
9  public class PrintRepeat {
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* Kamus : */
16         int N;
17         int i;
18         Scanner masukan=new Scanner(System.in);
19         /* Program */
20         System.out.print ("Nilai N >0 = ");
21         /* Inisialisasi */
22         N = masukan.nextInt();
23         i = 1;
24         /* First Elmt */
25         System.out.print ("Print i dengan REPEAT: \n");
26         do{
27             System.out.print (i+"\n"); /* Proses */
28             i++; /* Next Elmt */
29         }
30         while (i <= N); /* Kondisi pengulangan */
31     }
32 }
33

```

- Keyword

`import java.util.Scanner;`

`public class PrintRepeat {` adalah pendeklarasian kelas dengan nama kelas **PrintRepeat** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. `Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int N; int i;` pendeklarasian variabel N dan I, dengan tipe data integer tanpa masukkan nilai.

`Scanner masukan=new Scanner(System.in);`

`System.out.print ("Nilai N >0 = ");` menampilkan apa yang dipanggil pada method.

`N = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel N dan menyimpan masukan.

`i = 1;` sebuah assigmen atau sebuah variabel I dengan masukkan nilai 1.

`System.out.print ("Print i dengan REPEAT: \n");` memanggil print dan menampilkan "Print I dengan REPEAT" sesuai isi method.

do{ Perulangan do-while. Pada do while biasanya kondisi terletak pada bagian bawah yang artinya do while akan melakukan terlebih dahulu perulangan, kemudian akan membaca kondisi (syarat) untuk berakhirnya perulangan.

System.out.print (i+"\n"); Menampilkan output dari inputan.

i++; dan nilai dari variabel **i** akan terus bertambah satu sampai **i** bernilai **N**.

}

while (i <= N); sebuah perulangan.

}

}

- **Output**

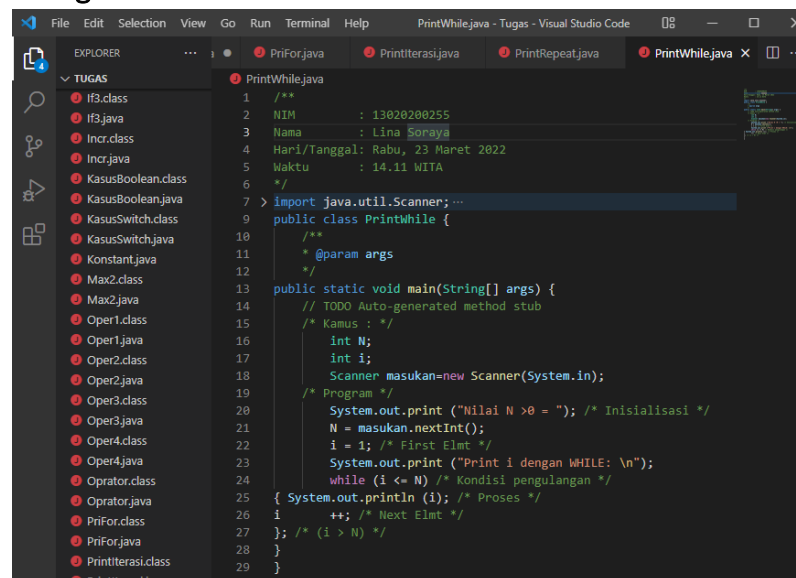


```
D:\PBO java\Tugas>javac PrintRepeat.java

D:\PBO java\Tugas>java PrintRepeat
Nilai N >0 = 4
Print i dengan REPEAT:
1
2
3
4

D:\PBO java\Tugas>
```

- **Listing 13**



```
1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Rabu, 23 Maret 2022
5   * Waktu    : 14.11 WITA
6   */
7  > import java.util.Scanner; ...
9  public class PrintWhile {
10     /**
11      * @param args
12      */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* Kamus : */
16         int N;
17         int i;
18         Scanner masukan=new Scanner(System.in);
19         /* Program */
20         System.out.print ("Nilai N >0 = "); /* Inisialisasi */
21         N = masukan.nextInt();
22         i = 1; /* First Elmt */
23         System.out.print ("Print i dengan WHILE: \n");
24         while (i <= N) /* Kondisi pengulangan */
25         { System.out.println (i); /* Proses */
26           i ++; /* Next Elmt */
27         }; /* (i > N) */
28     }
29 }
```

- **Keyword**

import java.util.Scanner; impor kelas Scanner.

/* Baca N, Print i = 1 s/d N dengan WHILE */

public class PrintWhile { adalah pendeklarasian kelas dengan nama kelas **PrintWhile** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int N; int i;` pendeklarasian variabel N dan I, dengan tipe data integer tanpa masukkan nilai.

`Scanner masukan=new Scanner(System.in);` kode untuk menginstansi objek.

`System.out.print ("Nilai N >0 = ");` menampilkan hasil dari inputan yaitu "Nilai N>0"

`N = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel N dan menyimpan masukan.

`i = 1;` sebuah assignment, yaitu tipe data integer dengan variabel i dan masukkan nilai 1.

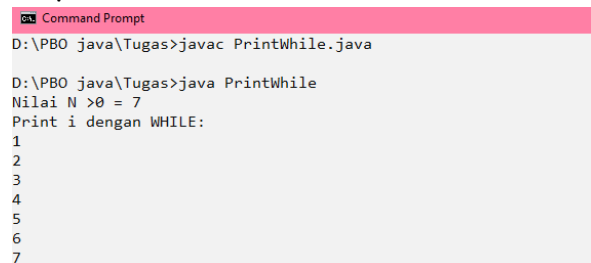
`System.out.print ("Print i dengan WHILE: \n");` menampilkan apa yang dipanggil pada method.

`while (i <= N)` sebuah perulangan dengan kondisi apakah $i \leq N$.

`{ System.out.println (i);` mencetak nilai dari i

`i++;` dan nilai i akan terus bertambah 1 sampai batak maks perulangan yang telah ditentukan.

- **Output**



```
Command Prompt
D:\PBO java\Tugas>javac PrintWhile.java

D:\PBO java\Tugas>java PrintWhile
Nilai N >0 = 7
Print i dengan WHILE:
1
2
3
4
5
6
7
```

- Listing 14

```

1  /**
2   NIM      : 13020200255
3   Nama     : Lina Soraya
4   Hari/Tanggal: Rabu, 23 Maret 2022
5   Waktu    : 14.13 WITA
6   */
7  import java.util.Scanner;
8  /* Baca N, */
9  /* Print i = 1 s/d N dengan while (ringkas) */
10 public class PrintWhile1 {
11     /**
12      * @param args
13      */
14     public static void main(String[] args) {
15         // TODO Auto-generated method stub
16         /* Kamus : */
17         int N;
18         int i = 1;
19         Scanner masukan=new Scanner(System.in);
20         /* Program */
21         System.out.print ("Nilai N >0 = ");
22         N = masukan.nextInt();
23         System.out.print ("Print i dengan WHILE (ringkas): \n");
24         while (i <= N){
25             System.out.println (i++);
26         } /* (i > N)*/
27     }
28 }

```

- Keyword

`import java.util.Scanner;` mengimpor scanner agar diketahui method=method yang ada pada Scanner.

`/* Baca N, */`

`/* Print i = 1 s/d N dengan while (ringkas) */`

`public class PrintWhile1 {` adalah pendeklarasian kelas dengan nama kelas **PrintWhile1** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int N; int i = 1;` pendeklarasian variabel N dan I, dengan masukkan nilai i=1.

`Scanner masukan=new Scanner(System.in);` menginstansi objek.

`System.out.print ("Nilai N >0 = ");` menampilkan hasil dari inputan method.

`N = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel N dan menyimpan masukan.

`System.out.print ("Print i dengan WHILE (ringkas): \n");` menampilkan output dari inputan.

`while (i <= N){` sebuah perulangan dengan kondisi apakah i<=N.

`System.out.println (i++);` menampilkan inputan dan nilai i akan terus bertambah 1 sampai batak maks perulangan yang telah ditentukan.

- Output

```

Command Prompt

D:\PBO java\Tugas>javac PrintWhile1.java

D:\PBO java\Tugas>java PrintWhile1
Nilai N >0 = 2
Print i dengan WHILE (ringkas):
1
2

D:\PBO java\Tugas>_

```

- Listing 15

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Rabu, 23 Maret 2022
5  Waktu    : 14.18 WITA
6  */
7  import java.util.Scanner;
8  /* contoh baca nilai x, */
9  /* Jumlahkan nilai yang dibaca dengan ITERATE */
10 public class PrintXinterasi {
11     /**
12     * @param args
13     */
14     public static void main(String[] args) {
15         // TODO Auto-generated method stub
16         /* Kamus : */
17         int Sum=0;
18         int x;
19         Scanner masukan=new Scanner(System.in);
20         /* Program */
21         System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");
22         x = masukan.nextInt(); /* First Elmt */
23         if (x == 999){
24             System.out.print ("Kasus kosong \n");
25         }else{ /* Minimal ada satu data yang dijumlahkan
26             */
27             Sum = x; /* Inisialisasi; invariant !! */
28             for (;;){
29                 System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");
30                 x = masukan.nextInt(); /* Next Elmt */
31                 if(x==999)
32                     break;
33                 else{
34                     Sum = Sum + x; /* Proses */
35                 }
36             }
37             System.out.println("Hasil penjumlahan = "+ Sum);
38             /* Terminasi */
39         }
40     }
41 }

```

- Keyword

import java.util.Scanner; mengimpor scanner agar diketahui method=method yang ada pada Scanner.

/* contoh baca nilai x, */

/* Jumlahkan nilai yang dibaca dengan ITERATE */

public class PrintXinterasi { adalah pendeklarasian kelas dengan nama kelas PrintXinterasi dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan. **Public static void main(String[] args) {** adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan.

`int Sum=0; int x;` pendeklarasian tipe data integer dengan variabel sum dan memiliki nilai 0, serta variabel x tanpa masukkan nilai.

`Scanner masukan=new Scanner(System.in);` kode untuk menginstansi objek.

`System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");` perintah menampilkan inputan.

`x = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel x dan menyimpan masukkan.

`if (x == 999){` jika nilai i sama dengan 999 maka akan melakukan pengecekan dan menampilkan

`System.out.print ("Kasus kosong \n");` ditampilkan "kasus kosong".

`}else{` jika nilai i tidak sama dengan 999 maka minimal akan satu data yang ditambahkan.

`Sum = x`

`for (;){`

`System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");` perintah menampilkan apa yang dideklarasikan.

`x = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel x dan menyimpan masukkan.

`if(x==999)` jika inputan tidak sama dengan 999 dan diakhiri dengan 999, maka

`break;`

`else{`

`Sum = Sum + x;`

`System.out.println("Hasil penjumlahan = "+ Sum);` ditampilkan nilai hasil dari penjumlahan variabel sum.

- **Output**

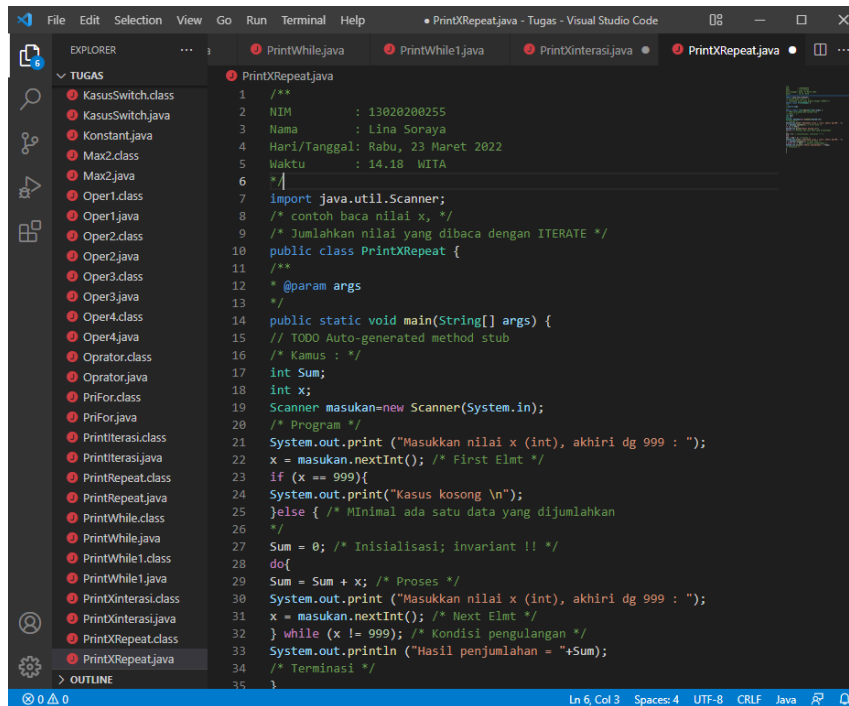
```
Command Prompt

D:\PBO java\Tugas>javac PrintXinterasi.java

D:\PBO java\Tugas>java PrintXinterasi
Masukkan nilai x (int), akhiri dg 999 : 20999
Masukkan nilai x (int), akhiri dg 999 : 20 999
Masukkan nilai x (int), akhiri dg 999 : Hasil penjumlahan = 21019

D:\PBO java\Tugas>
```

- Listing 16



- Keyword

`import java.util.Scanner;` mengimpor scanner agar diketahui method=method yang ada pada Scanner.

`/* contoh baca nilai x, */`

`/* Jumlahkan nilai yang dibaca dengan ITERATE */`

`public class PrintXRepeat{` adalah pendeklarasian kelas dengan nama kelas **PrintXRepeat** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int Sum; int x;` pendeklarasian tipe data integer dengan variabel sum dan memiliki nilai 0, serta variabel x tanpa memasukkan nilai.

`Scanner masukan=new Scanner(System.in);` kode untuk menginstansi objek.

`System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");` perintah menampilkan apa yang dideklarasikan.

`x = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel x dan menyimpan masukan.

if (x == 999){ }{ jika nilai i sama dengan 999 maka akan melakukan pengecekan dan menampilkan

System.out.print("Kasus kosong \n"); ditampilkan "kasus kosong".

}else { }{ jika nilai i tidak sama dengan 999 maka akan melakukan pengecekan dan menampilkan

Sum = 0; variabel sum dengan nilai 0.

do{ }{ perulangan do-while. . Pada do while biasanya kondisi terletak pada bagian bawah yang artinya do while akan melakukan terlebih dahulu perulangan, kemudian akan membaca kondisi (syarat) untuk berakhirnya perulangan.

Sum = Sum + x; variabel sum dengan memasukkan nilai yaitu sum + x.

System.out.print ("Masukkan nilai x (int), akhiri dg 999 : "); perintah menampilkan inputan.

x = masukan.nextInt(); memasukkan nilai dari hasil inputan ke variabel x dan menyimpan masukan.

} while (x != 999); }{ perulangan dengan kondisi apakah variabel x tidak sama dengan 999.

System.out.println ("Hasil penjumlahan = "+Sum); hasil perulangan diatas akan ditampilkan disini.

• Output

```
Command Prompt
D:\PBO java\Tugas>java PrintXRepeat
Masukkan nilai x (int), akhiri dg 999 : 1 999
Masukkan nilai x (int), akhiri dg 999 : Hasil penjumlahan = 1
D:\PBO java\Tugas>
```

• Listing 17

```
PrintXWhile.java - Tugas - Visual Studio Code
EXPLOLER
TUGAS
  Konstant.java
  Max2.class
  Max2.java
  Oper1.class
  Oper1.java
  Oper2.class
  Oper2.java
  Oper3.class
  Oper3.java
  Oper4.class
  Oper4.java
  Operator.class
  Operator.java
  PriFor.class
  PriFor.java
  PrintIterasi.class
  PrintIterasi.java
  PrintRepeat.class
  PrintRepeat.java
  PrintWhile.class
  PrintWhile.java
  PrintWhile1.class
  PrintWhile1.java
  PrintXInterasi.class
  PrintXInterasi.java
  PrintXRepeat.class
  PrintXRepeat.java
  PrintXWhile.class
  PrintXWhile.java
  PrintXWhile1.java
  PrintXWhile1.java

PrintXWhile.java
1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Rabu, 23 Maret 2022
5  Waktu    : 14.28 WITA
6  */
7  import java.util.Scanner;
8  /* contoh baca nilai x, */
9  /* Jumlahkan nilai yang dibaca dengan WHILE */
10 public class PrintXWhile {
11     /**
12     * @param args
13     */
14     public static void main(String[] args) {
15         // TODO Auto-generated method stub
16         /* Kamus : */
17         int Sum;
18         int x;
19         Scanner masukan = new Scanner(System.in);
20         /* Program */
21         Sum = 0; /* Inisialisasi */
22         System.out.print("Masukkan nilai x (int), akhiri dg 999 : ");
23         x = masukan.nextInt(); /* First Elmt */
24         while (x != 999) /* Kondisi berhenti */ {
25             Sum = Sum + x; /* Proses */
26             System.out.print("Masukkan nilai x (int), akhiri dg 999 : ");
27             x = masukan.nextInt(); /* First Elmt */
28         }
29         System.out.println("Hasil penjumlahan = "+ Sum); /*
30         Terminasi */
31     }
32 }
```

- **Keyword**

`import java.util.Scanner;` mengimpor scanner agar diketahui method=method yang ada pada Scanner.

`/* contoh baca nilai x, */`

`/* Jumlahkan nilai yang dibaca dengan WHILE */`

`public class PrintXWhile {` adalah pendeklarasian kelas dengan nama kelas **PrintXWhile** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

`int Sum; int x; ;` pendeklarasian tipe data integer dengan variabel sum dan memiliki nilai 0, serta variabel x tanpa masukan nilai.

`Scanner masukan=new Scanner(System.in);` kode untuk menginstansi objek.

`Sum = 0;` variabel sum dengan nilai 0.

`System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");` perintah menampilkan inputan.

`x = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel x dan menyimpan masukan.

`while (x != 999) {` perulangan dengan kondisi apakah variabel x tidak sama dengan 999

`Sum = Sum + x;` variable sum dengan masukan nilai yaitu sum + x.

`System.out.print("Masukkan nilai x (int), akhiri dg 999 : ");` perintah menampilkan inputan.

`x = masukan.nextInt();` memasukkan nilai dari hasil inputan ke variabel x dan menyimpan masukan.

`}`

`System.out.println("Hasil penjumlahan = "+ Sum);` ditampilkan nilai hasil dari penjumlahan variabel sum.

- **Output**

```
Command Prompt

>.\PBO java\Tugas>javac PrintXWhile.java

>.\PBO java\Tugas>java PrintXWhile
Masukkan nilai x (int), akhiri dg 999 : 12 999
Masukkan nilai x (int), akhiri dg 999 : Hasil penjumlahan = 12

>.\PBO java\Tugas>_
```

- Listing 18

```

1  /**
2  NIM      : 13020200255
3  Nama     : Lina Soraya
4  Hari/Tanggal: Rabu, 23 Maret 2022
5  Waktu    : 14.32 WITA
6  */
7  import java.util.Scanner;
8  /* Contoh program yang mengandung prosedur dan fungsi */
9  /* prosedur/fungsi */
10 public class SubProgram {
11     /**
12     * @param args
13     */
14     public static int maxab (int a, int b){
15         /* mencari maksimum dua bilangan bulat */
16         return ((a >= b) ? a : b);
17     }
18     public static void tukar (int a, int b)
19     { /* menukar dua bilangan bulat */
20         int temp;
21         temp = a;
22         a = b;
23         b = temp;
24         System.out.println ("Ke dua bilangan setelah tukar: a = "+ a +" b = "+ b);
25     }
26     /** Program Utama */
27     public static void main(String[] args) {
28         // TODO Auto-generated method stub
29         /* Membaca dua bilangan integer */
30         /* Menuliskan maksimum dua bilangan yang dibaca
31         dg memanggil fungsi */
32         /* Menukar kedua bilangan dengan 'prosedur' */
33         int a, b;
34         Scanner masukan=new Scanner(System.in);
35         System.out.print ("Maksimum dua bilangan \n");
36         System.out.print ("Ketikkan dua bilangan, pisahkan dg RETURN : \n");
37         a = masukan.nextInt();
38         b = masukan.nextInt();
39         System.out.println ("Ke dua bilangan : a = "+ a +" b = "+ b);
40         System.out.println ("Maksimum = " + (maxab(a, b)));
41         System.out.print("Tukar kedua bilangan... \n");
42         tukar (a, b);
43     }
44 }

```

- Tujuan

Program yang bertujuan untuk menampilkan hasil inputan yang menggunakan prosedur dan fungsi.

- Keyword

import java.util.Scanner; mengimport scanner agar diketahui method=method yang ada pada Scanner.

/* Contoh program yang mengandung prosedur dan fungsi */

/* prosedur/fungsi */

public class SubProgram { adalah pendeklarasian kelas dengan nama kelas **SubProgram** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

public static int maxab (int a, int b){

/* mencari maksimum dua bilangan bulat */

return ((a >= b) ? a : b); mengembalikan nilai dari variabel.

}

`public static void tukar (int a, int b)` method untuk memanggil nilai tukar dari variabel a dan b.

`{ /* menukar dua bilangan bulat */`

`int temp;` variabel temp tipe data integer.

`temp = a;` nilai dari variabel temp yaitu a.

`a = b;` nilai a sama dengan b

`b = temp;` maka nilai b sama dengan temp

`System.out.println ("Ke dua bilangan setelah tukar: a = "+ a +" b = "+ b);` menampilkan hasil inputan.

`}`

`/** Program Utama **/`

`Public static void main(String[] args) {` adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Public static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*

`Scanner masukan=new Scanner(System.in);` kode untuk menginstansi objek

`System.out.print ("Maksimum dua bilangan \n");` menampilkan "Maksimum dua bilangan"

`System.out.print ("Ketikkan dua bilangan, pisahkan dg RETURN : \n");` menampilkan "Ketikkan dua bilangan, pisahkan dengan RETURN".

`a = masukan.nextInt();`

`b = masukan.nextInt();` deklarasi nilai a dan b adalah sebuah inputan

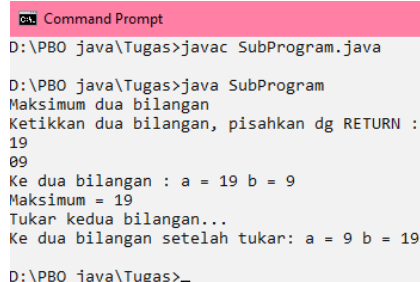
`System.out.println ("Ke dua bilangan : a = "+ a +" b = "+ b);`

`System.out.println ("Maksimum = " + (maxab (a, b)));`

`System.out.print("Tukar kedua bilangan ");` menampilkan semua yang diinputkan.

`tukar (a, b);` fungsi tukar variabel a dengan variabel b.

- **Output**

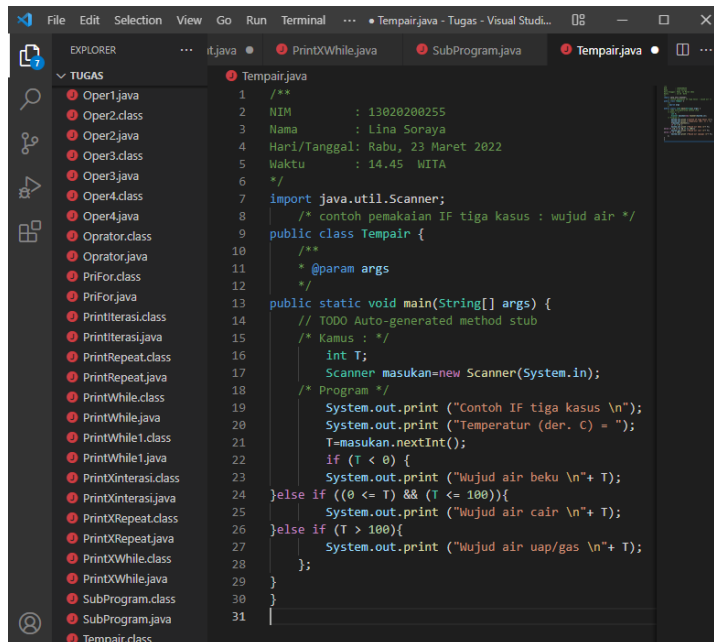


```
Command Prompt
D:\PBO java\Tugas>javac SubProgram.java

D:\PBO java\Tugas>java SubProgram
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
19
09
Ke dua bilangan : a = 19 b = 9
Maksimum = 19
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a = 9 b = 19

D:\PBO java\Tugas>_
```

- Listing 19



```

1  /**
2   * NIM      : 13020200255
3   * Nama     : Lina Soraya
4   * Hari/Tanggal: Rabu, 23 Maret 2022
5   * Waktu    : 14.45 WITA
6   */
7  import java.util.Scanner;
8  /* contoh pemakaian IF tiga kasus : wujud air */
9  public class Tempair {
10     /**
11     * @param args
12     */
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         /* Kamus : */
16         int T;
17         Scanner masukan=new Scanner(System.in);
18         /* Program */
19         System.out.print ("Contoh IF tiga kasus \n");
20         System.out.print ("Temperatur (der. C) = ");
21         T=masukan.nextInt();
22         if (T < 0) {
23             System.out.print ("Wujud air beku \n"+ T);
24         }else if ((0 <= T) && (T <= 100)){
25             System.out.print ("Wujud air cair \n"+ T);
26         }else if (T > 100){
27             System.out.print ("Wujud air uap/gas \n"+ T);
28         }
29     }
30 }
31

```

- Tujuan

Program ini berfungsi untuk menampilkan hasil dari program yang menggunakan perintah IF.

- Keyword

import java.util.Scanner; mengimpor scanner agar diketahui method=method yang ada pada Scanner.

/* contoh pemakaian IF tiga kasus : wujud air */

public class Tempair { adalah pendeklarasian kelas dengan nama kelas **Tempair** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifer yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

int T; variabel T dengan tipe data integer.

Scanner masukan = new Scanner(System.in); kode program untuk menginstansi objek.

System.out.print ("Contoh IF tiga kasus \n");

System.out.print ("Temperatur (der. C) = "); memanggil print dari method yang dimasukkan.

T=masukan.nextInt(); masukkan nilai variabel T

if (T < 0) Perulangan dengan kondisi apakah T < 0. Jika ia akan ditampilkan pada statement dibawah ini.

```
{  
System.out.print ("Wujud air beku \n"+ T);  
}
```

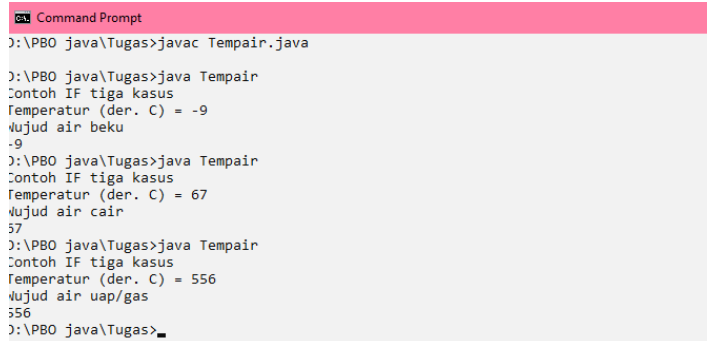
else if ((0 <= T) && (T <= 100)){ perulangan dengan kondisi apakah 0 lebih kecil sama dengan T, dan lebih kecil sma dengan 100? Jika iya akan ditampilkan di statement pemanggilan.

```
System.out.print ("Wujud air cair \n"+ T);  
}
```

else if (T > 100){ jika lebih besar dari seratus , maka

System.out.print ("Wujud air uap/gas \n"+ T); ditampilkan wujud air uap gas dari T

- **Output**



```
Command Prompt  
D:\PBO java\Tugas>javac Tempair.java  
  
D:\PBO java\Tugas>java Tempair  
Contoh IF tiga kasus  
temperatur (der. C) = -9  
wujud air beku  
-9  
D:\PBO java\Tugas>java Tempair  
Contoh IF tiga kasus  
temperatur (der. C) = 67  
wujud air cair  
67  
D:\PBO java\Tugas>java Tempair  
Contoh IF tiga kasus  
temperatur (der. C) = 556  
wujud air uap/gas  
556  
D:\PBO java\Tugas>
```

3. Kasus : Buat Flowchart dan Program menggunakan bahasa java untuk Konversi Waktu (Jam:Menit:Detik) dari masukan detik!

Menampilkan Waktu dalam format jam:menit:detik. Spesifikasi :

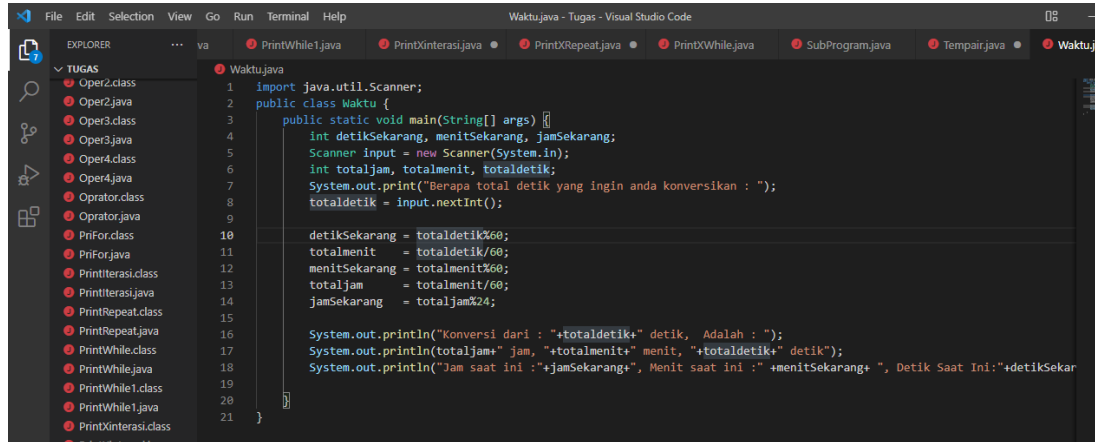
- a. mendapatkan total detik melalui masukan keyboard (misalnya. 1203183086)
- b. mendapatkan detik saat ini dari totalDetik % 60 (misal 1203183086 detik % 60 = 26)
- c. mendapatkan detik ssat ini dari totalDetik dengan membagi totalDetik dengan 60 (misal 1203183086 detik /60 = 20053051 menit)
- d. mendapatkan menit saat ini dari totalMenit % 60 (misalnya 20053051 menit % 60 = 31 menit saat ini)
- e. mendapatkan total jam totalJam dengan membagi totalMenit dengan 60 (misal 20053051 menit/60 = 334217 jam)
- f. mendapatkan jam saat ini dari totalJam % 24 (misal 334217 jam % 24 = 17 jam saa ini)

Kerangka Program

1. Masukkan total detik
2. Hitung detikSekarang = totalDetik %60

3. Hitung totalMenit = totalDetik/60
4. Hitung menitSekarang = totalMenit%60
5. Hitung totalJam = totalMenit / 60
6. Hitung jamSekarang = totalJam % 24
7. Tampil waktu (Jam:Menit:Detik)

- Listing



```

1  import java.util.Scanner;
2  public class Waktu {
3      public static void main(String[] args) {
4          int detikSekarang, menitSekarang, jamSekarang;
5          Scanner input = new Scanner(System.in);
6          int totaljam, totalmenit, totaldetik;
7          System.out.print("Berapa total detik yang ingin anda konversikan : ");
8          totaldetik = input.nextInt();
9
10         detikSekarang = totaldetik%60;
11         totalmenit = totaldetik/60;
12         menitSekarang = totalmenit%60;
13         totaljam = totalmenit/60;
14         jamSekarang = totaljam%24;
15
16         System.out.println("Konversi dari : "+totaldetik+" detik, Adalah : ");
17         System.out.println(totaljam+" jam, "+totalmenit+" menit, "+totaldetik+" detik");
18         System.out.println("Jam saat ini : "+jamSekarang+", Menit saat ini : "+menitSekarang+" , Detik Saat Ini: "+detikSekarang);
19     }
20 }
  
```

- Keyword

import java.util.Scanner; mengimpor scanner agar diketahui method=method yang ada pada Scanner.

public class Waktu { adalah pendeklarasian kelas dengan nama kelas **Waktu** dan menggunakan kode **Public** yang artinya kelas tersebut dapat diakses oleh kelas lain dimanapun. Pembuatan nama kelas harus sama dengan nama file yang akan disimpan.

Public static void main(String[] args) { adalah deklarasi method utama (*main method*), yaitu method utama yang akan dipanggil ketika program java dijalankan. **Publis static** yaitu sebuah modifier yang menentukan dari mana *method* dapat diakses. **Void** yaitu return type yang berarti kosong atau tidak memiliki keluaran. **Main** yaitu nama fungsi yang dapat dipanggil dari tempat lain pada program yang kita tulis. **Args** yaitu parameters, bagian ini opsional, adalah nilai masukan yang perlu disertakan saat memanggil *method*.

int detikSekarang, menitSekarang, jamSekarang;

Scanner input = new Scanner(System.in); kode untuk menginstansi objek

int totaljam, totalmenit, totaldetik;

System.out.print("Berapa total detik yang ingin anda konversikan : "); perintah menampilkan inputan.

totaldetik = input.nextInt();

detikSekarang = totaldetik%60;

totalmenit = totaldetik/60;

menitSekarang = totalmenit%60;

totaljam = totalmenit/60;

```

jamSekarang = totaljam%24;
System.out.println("Konversi dari : "+totaldetik+" detik, Adalah : ");
System.out.println(totaljam+" jam, "+totalmenit+" menit, "+totaldetik+" detik");
System.out.println("Jam saat ini : "+jamSekarang+", Menit saat ini : "+menitSekarang+",
Detik Saat Ini:"+detikSekarang);

```

adalah sebuah pernyataan (statement) yang memanggil *method* **println**. **Println** berarti setelah nilai ditampilkan secara otomatis java akan menambahkan *line separator*(pemisah baris).

- Ouput

```

Command Prompt
D:\PBO java\Tugas>javac Waktu.java
D:\PBO java\Tugas>java Waktu
Berapa total detik yang ingin anda konversikan : 12900
Konversi dari : 12900 detik, Adalah :
3 jam, 215 menit, 12900 detik
Jam saat ini :3, Menit saat ini :35, Detik Saat Ini:0

```

- Flowchart

