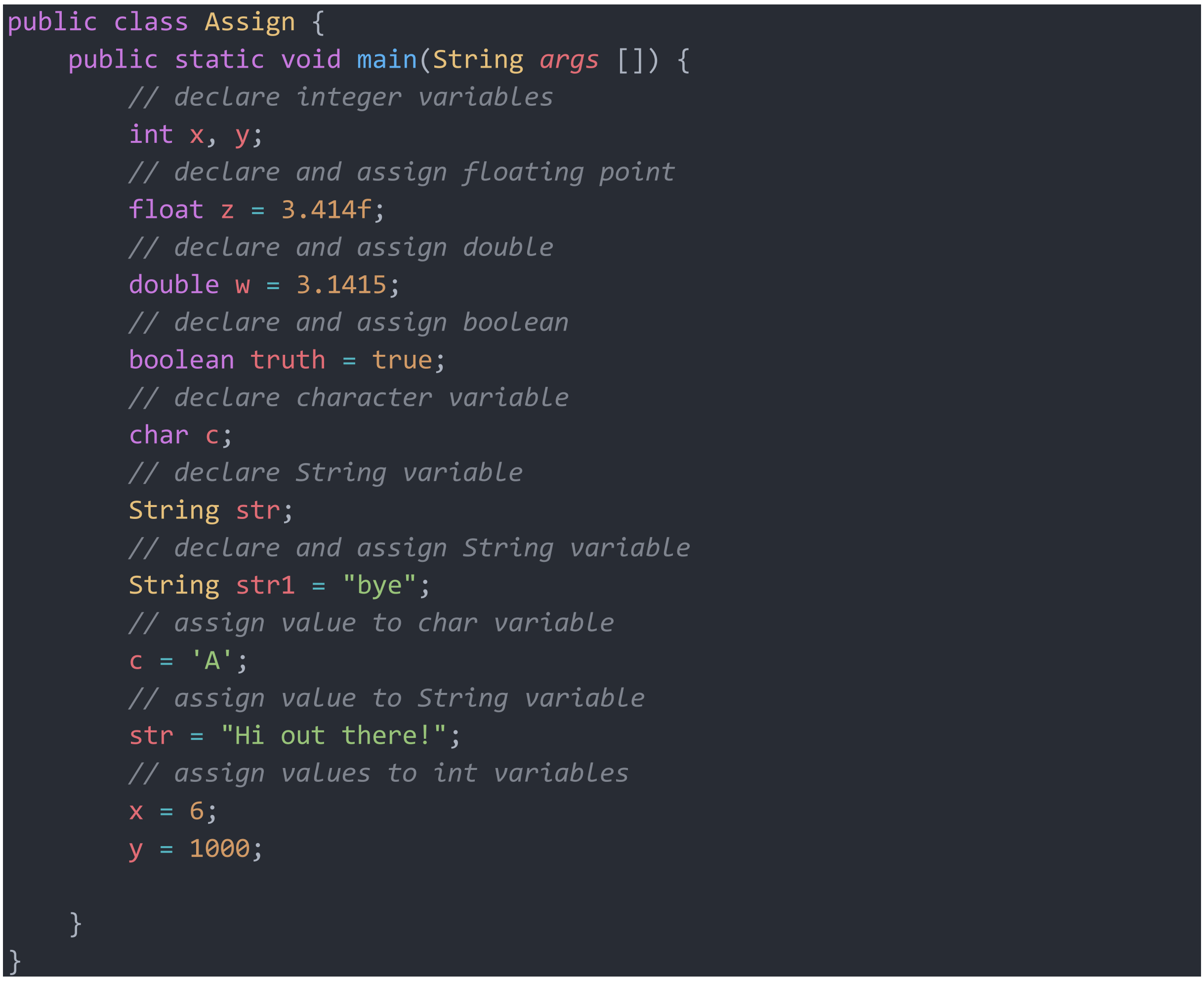
Nama : Mochammad khaerul ilman

Kelas : Prak PBO – C

Nim : 1227050073

Latihan Praktikum 2a

Percobaan 1 : Pendeklarasian variabel dan pemberian nilai



```
public class Assign {
    public static void main(String args []) {
        // declare integer variables
        int x, y;
        // declare and assign floating point
        float z = 3.414f;
        // declare and assign double
        double w = 3.1415;
        // declare and assign boolean
        boolean truth = true;
        // declare character variable
        char c;
        // declare String variable
        String str;
        // declare and assign String variable
        String str1 = "bye";
        // assign value to char variable
        c = 'A';
        // assign value to String variable
        str = "Hi out there!";
        // assign values to int variables
        x = 6;
        y = 1000;

    }
}
```

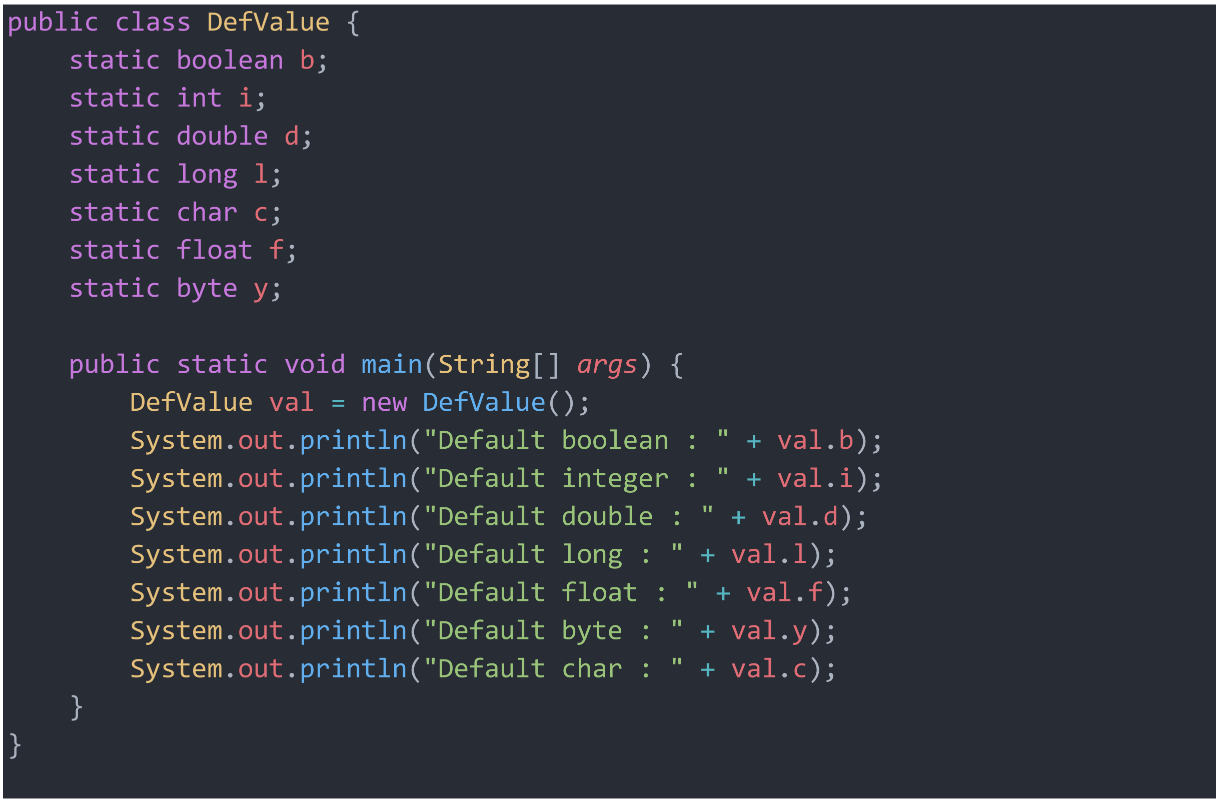
Catatan kesalahan :

Program benar tidak ada kesalahan program

Kesimpulan program :

Program ini mendeklarasikan dan menginisialisasi berbagai jenis variabel, termasuk integer, floating point, double, boolean, char, dan String

Percobaan 2 : Nilai default variabel

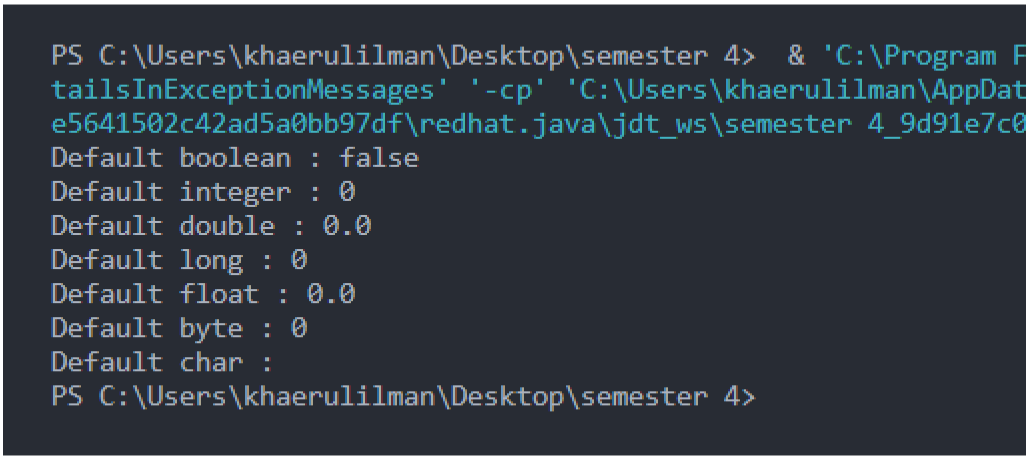


```
public class DefValue {
    static boolean b;
    static int i;
    static double d;
    static long l;
    static char c;
    static float f;
    static byte y;

    public static void main(String[] args) {
        DefValue val = new DefValue();
        System.out.println("Default boolean : " + val.b);
        System.out.println("Default integer : " + val.i);
        System.out.println("Default double : " + val.d);
        System.out.println("Default long : " + val.l);
        System.out.println("Default float : " + val.f);
        System.out.println("Default byte : " + val.y);
        System.out.println("Default char : " + val.c);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

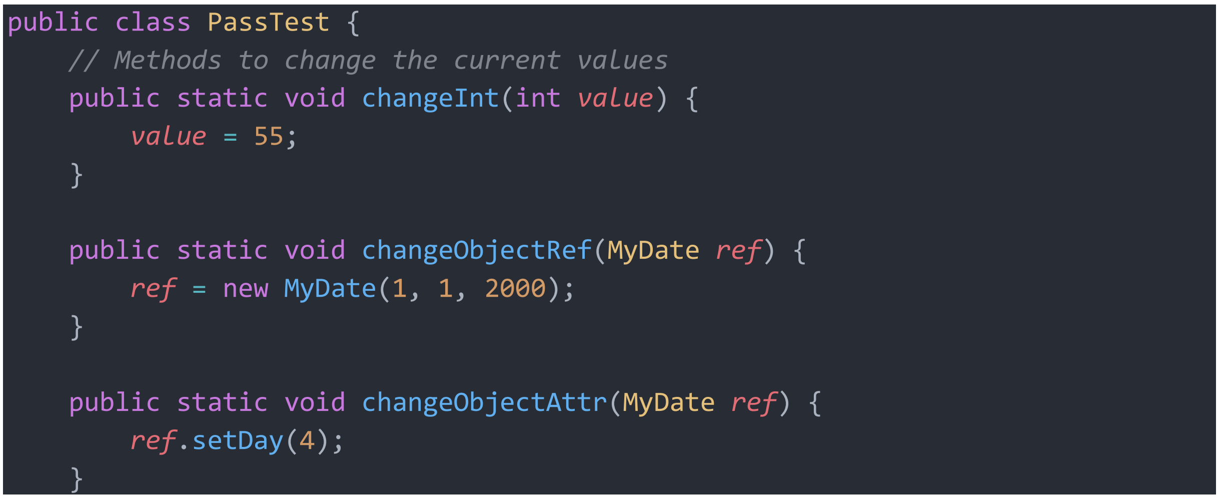


```
PS C:\Users\khaerulilman\Desktop\semester 4>  & 'C:\Program F
tailsInExceptionMessages' '-cp' 'C:\Users\khaerulilman\AppDat
e5641502c42ad5a0bb97df\redhat.java\jdt_ws\semester 4_9d91e7c0
Default boolean : false
Default integer : 0
Default double : 0.0
Default long : 0
Default float : 0.0
Default byte : 0
Default char :
PS C:\Users\khaerulilman\Desktop\semester 4>
```

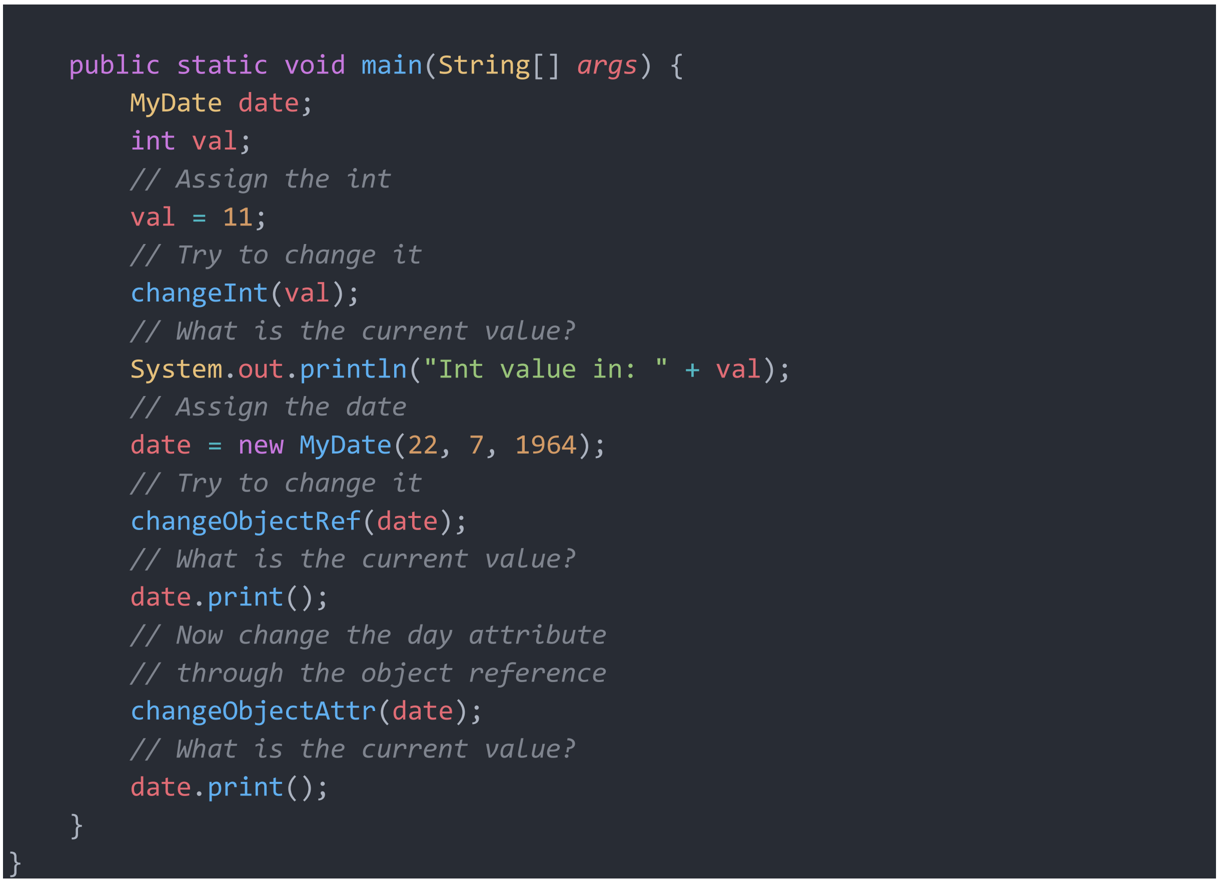
Kesimpulan program :

Program DefValue ini menunjukkan nilai default dari tipe data primitif dalam Java ketika variabel-variabel tersebut tidak diinisialisasi secara eksplisit. Program ini mendeklarasikan beberapa variabel statis dengan tipe data primitif seperti boolean, integer, double, long, char, float, dan byte.

Percobaan 3 : Menggambarkan pass by value



```
public class PassTest {
    // Methods to change the current values
    public static void changeInt(int value) {
        value = 55;
    }

    public static void changeObjectRef(MyDate ref) {
        ref = new MyDate(1, 1, 2000);
    }

    public static void changeObjectAttr(MyDate ref) {
        ref.setDay(4);
    }
```

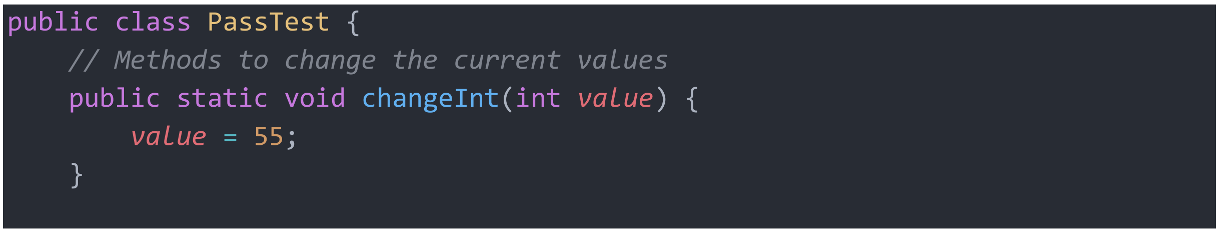


```java
    public static void main(String[] args) {
        MyDate date;
        int val;
        // Assign the int
        val = 11;
        // Try to change it
        changeInt(val);
        // What is the current value?
        System.out.println("Int value in: " + val);
        // Assign the date
        date = new MyDate(22, 7, 1964);
        // Try to change it
        changeObjectRef(date);
        // What is the current value?
        date.print();
        // Now change the day attribute
        // through the object reference
        changeObjectAttr(date);
        // What is the current value?
        date.print();
    }
}
```

Catatan kesalahan :

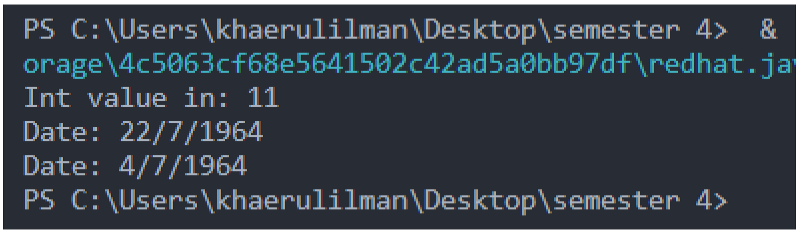
- tidak ada class MyDate
- tidak ada method public MyDate(int day, int month, int year)
- tidak ada method setDay(int day)
- tidak ada method print()

code setelah diperbaiki :



```java
public class PassTest {
    // Methods to change the current values
    public static void changeInt(int value) {
        value = 55;
    }
    public static void changeObjectRef(MyDate ref) {
        ref = new MyDate(1, 1, 2000);
    }

    public static void changeObjectAttr(MyDate ref) {
        ref.setDay(4);
    }

    public static void main(String[] args) {
        MyDate date;
        int val;
        // Assign the int
        val = 11;
        // Try to change it
        changeInt(val);
        // What is the current value?
        System.out.println("Int value in: " + val);
        // Assign the date
        date = new MyDate(22, 7, 1964);
        // Try to change it
        changeObjectRef(date);
        // What is the current value?
        date.print();
        // Now change the day attribute
        // through the object reference
        changeObjectAttr(date);
        // What is the current value?
        date.print();
    }
}

class MyDate {
    private int day;
    private int month;
    private int year;

    public MyDate(int day, int month, int year) {
        this.day = day;
        this.month = month;
        this.year = year;
    }

    public void setDay(int day) {
        this.day = day;
    }

    public void print() {
        System.out.println("Date: " + day + "/" + month + "/" + year);
    }
}
```

Output :



```
PS C:\Users\khaerulilman\Desktop\semester 4>  & 
orage\4c5063cf68e5641502c42ad5a0bb97df\redhat.ja
Int value in: 11
Date: 22/7/1964
Date: 4/7/1964
PS C:\Users\khaerulilman\Desktop\semester 4>
```

Kesimpulan program :

Program PassTest adalah contoh program Java yang mengilustrasikan konsep pengiriman parameter ke metode Java.

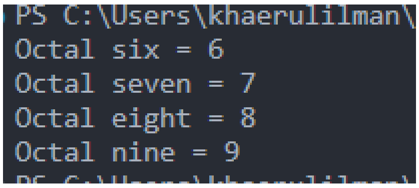
Percobaan 4 : Menggunakan bilangan basis 8 (Octal)

```java
public class Octal {
    public static void main(String[] args) {
        int six = 06;
        int seven = 07;
        int eight = 010;
        int nine = 011;
        System.out.println("Octal six = " + six);
        System.out.println("Octal seven = " + seven);
        System.out.println("Octal eight = " + eight);
        System.out.println("Octal nine = " + nine);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

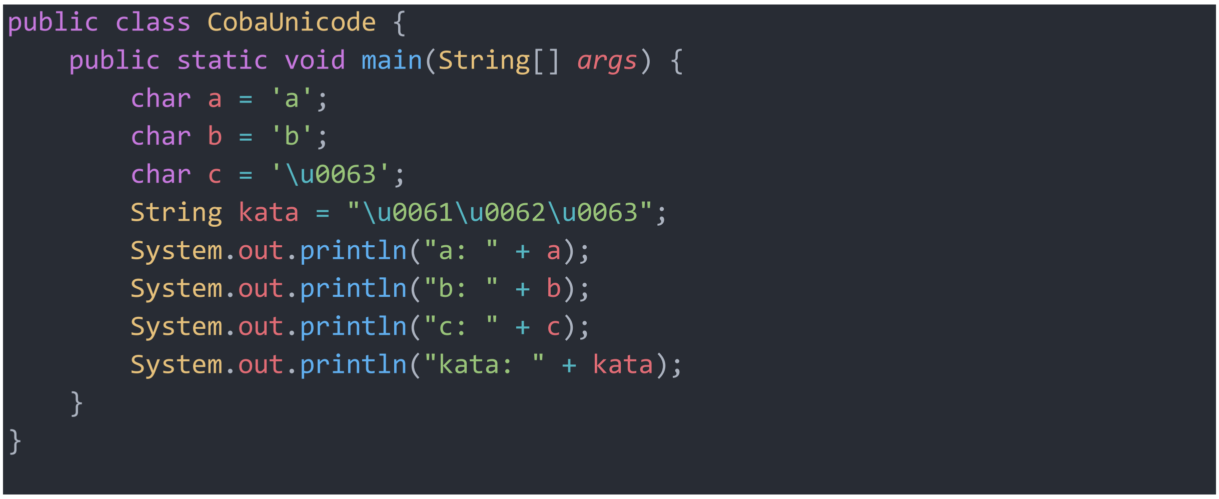


```
PS C:\Users\khaerulilman\
Octal six = 6
Octal seven = 7
Octal eight = 8
Octal nine = 9
PS C:\Users\khaerulilman\
```

Kesimpulan program :

contoh program Java yang menunjukkan cara menggunakan literal oktal dalam Java.

Percobaan 5 : Menggunakan unicode

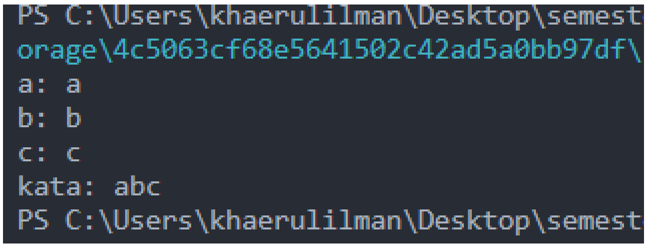


```java
public class CobaUnicode {
    public static void main(String[] args) {
        char a = 'a';
        char b = 'b';
        char c = '\u0063';
        String kata = "\u0061\u0062\u0063";
        System.out.println("a: " + a);
        System.out.println("b: " + b);
        System.out.println("c: " + c);
        System.out.println("kata: " + kata);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

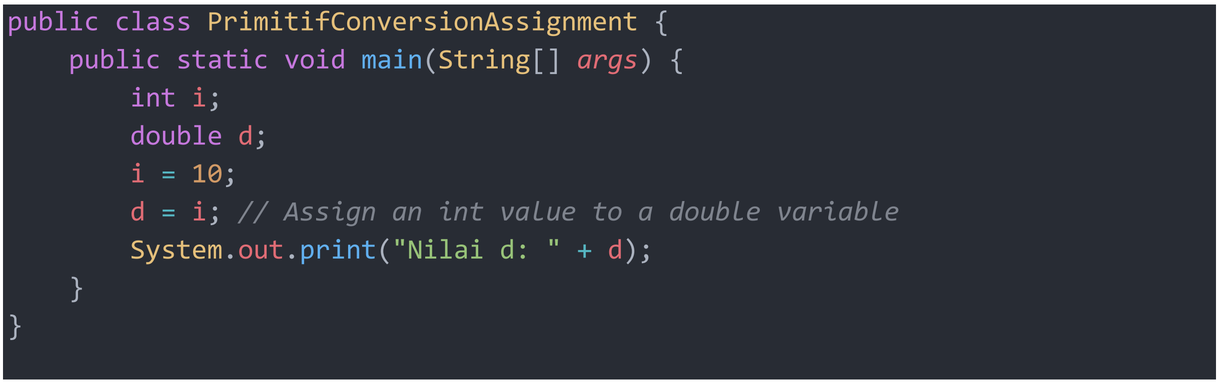


```
PS C:\Users\khaerulilman\Desktop\semest
orage\4c5063cf68e5641502c42ad5a0bb97df\
a: a
b: b
c: c
kata: abc
PS C:\Users\khaerulilman\Desktop\semest
```

Kesimpulan program :

contoh program Java ini menunjukkan penggunaan karakter Unicode dalam Java.

Percobaan 6 : Primitive conversion - assignment

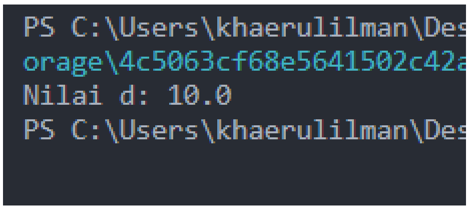


```java
public class PrimitifConversionAssignment {
    public static void main(String[] args) {
        int i;
        double d;
        i = 10;
        d = i; // Assign an int value to a double variable
        System.out.print("Nilai d: " + d);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

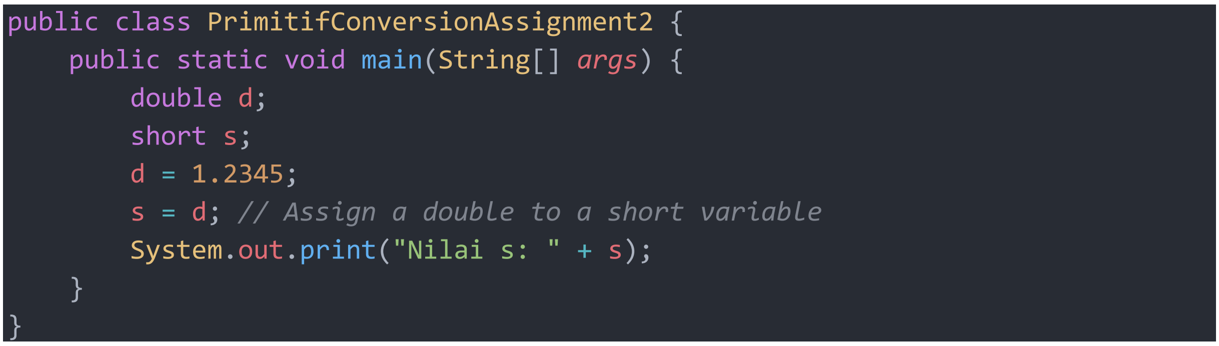


```
PS C:\Users\khaerulilman\Des
orage\4c5063cf68e5641502c42a
Nilai d: 10.0
PS C:\Users\khaerulilman\Des
```

Kesimpulan program :

contoh program Java ini menunjukkan konversi tipe data primitif secara otomatis dalam Java.

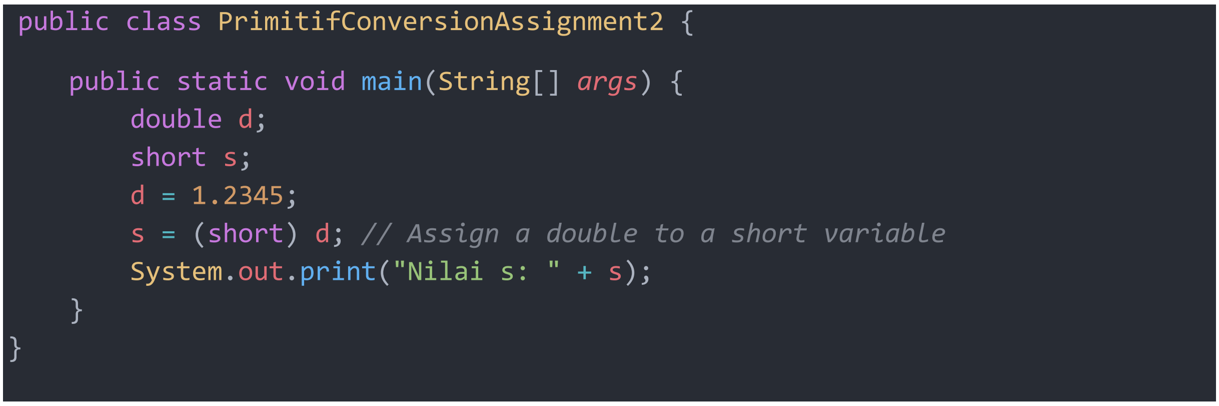
Percobaan 7 : Primitive conversion – assignment



```java
public class PrimitifConversionAssignment2 {
    public static void main(String[] args) {
        double d;
        short s;
        d = 1.2345;
        s = d; // Assign a double to a short variable
        System.out.print("Nilai s: " + s);
    }
}
```

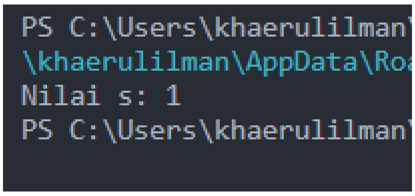
Catatan kesalahan :

- mencoba memberi value variabel d (double) pada variabel s (short) , maka d harus di konversi terlebih dahulu



```java
public class PrimitifConversionAssignment2 {

    public static void main(String[] args) {
        double d;
        short s;
        d = 1.2345;
        s = (short) d; // Assign a double to a short variable
        System.out.print("Nilai s: " + s);
    }
}
```

Output :

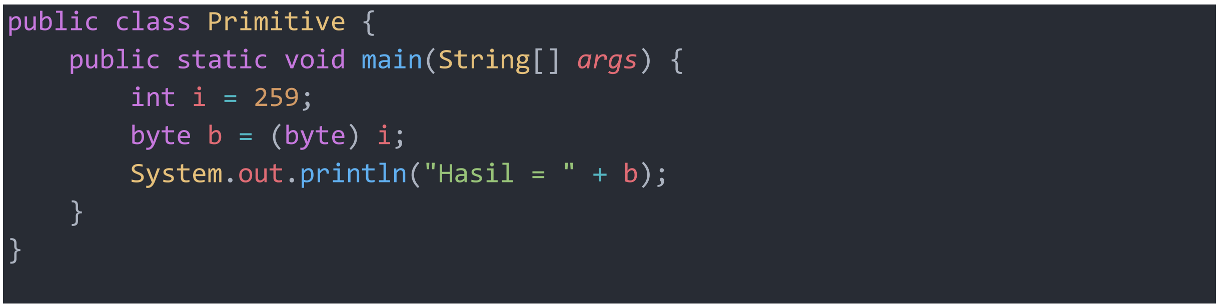


```
PS C:\Users\khaerulilman
\khaerulilman\AppData\Roa
Nilai s: 1
PS C:\Users\khaerulilman
```

Kesimpulan program :

contoh program Java yang menunjukkan cara melakukan konversi tipe data primitif secara eksplisit dalam Java. Dalam program ini, variabel d diinisialisasi dengan nilai 1.2345, dan kemudian nilai variabel d dicoba di-assign ke variabel s yang memiliki tipe data short. Karena tipe data short memiliki rentang nilai yang lebih kecil dari tipe data double, konversi ini harus dilakukan secara eksplisit dengan melakukan casting (short) pada variabel d.

Percobaan 8 : Primitive conversion – assignment

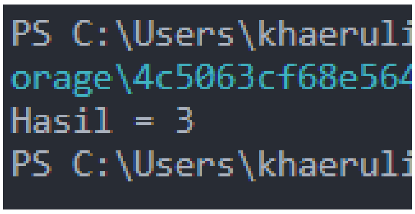


```
public class Primitive {
    public static void main(String[] args) {
        int i = 259;
        byte b = (byte) i;
        System.out.println("Hasil = " + b);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

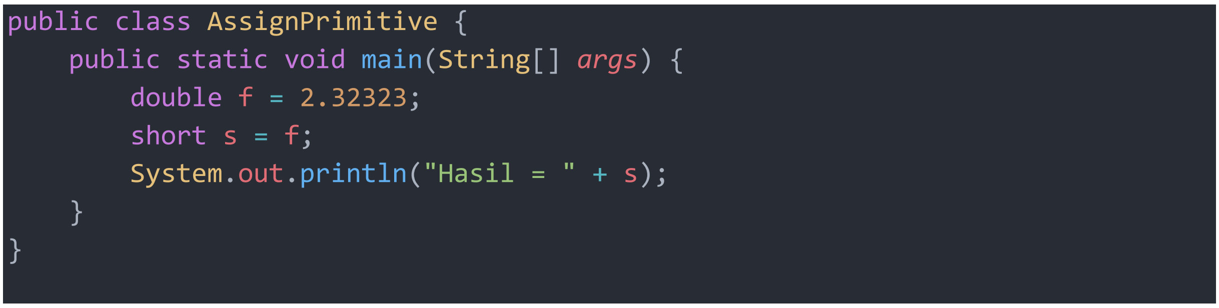


```
PS C:\Users\khaeruli
orage\4c5063cf68e564
Hasil = 3
PS C:\Users\khaeruli
```

Kesimpulan :

cara konversi tipe data primitif dalam Java dengan menggunakan casting. Dalam program ini, sebuah nilai integer (i) diinisialisasi dengan nilai 259, yang melebihi rentang nilai yang dapat disimpan dalam tipe data byte. Kemudian, nilai i dicoba di-assign ke variabel byte (b) dengan menggunakan casting (byte)

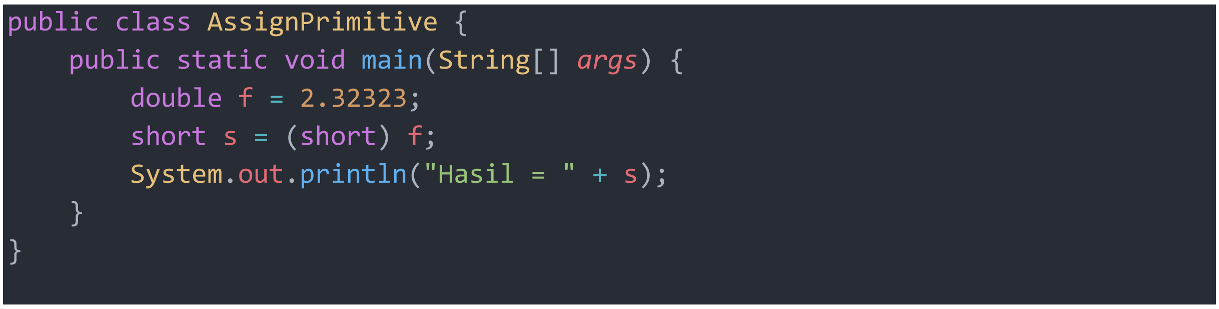
Percobaan 9 : Primitive conversion – assignment



```
public class AssignPrimitive {
    public static void main(String[] args) {
        double f = 2.32323;
        short s = f;
        System.out.println("Hasil = " + s);
    }
}
```

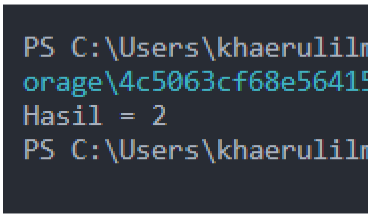
Catatan kesalahan :

- mencoba memberi value variabel f (double) pada variabel s (short) , maka f harus di konversi/casting terlebih dahulu



```
public class AssignPrimitive {
    public static void main(String[] args) {
        double f = 2.32323;
        short s = (short) f;
        System.out.println("Hasil = " + s);
    }
}
```

output :



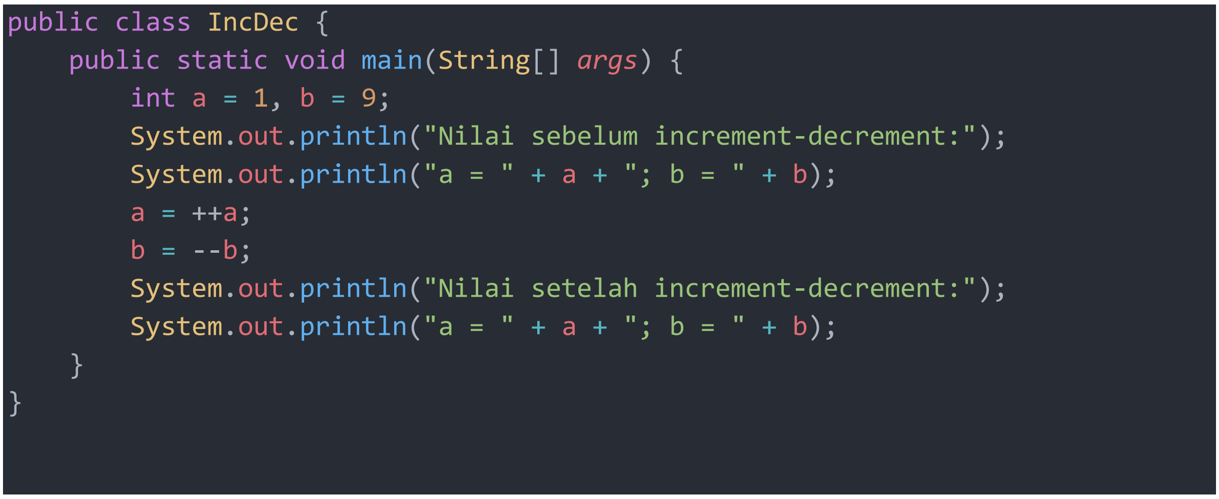
```
PS C:\Users\khaerulilm
orage\4c5063cf68e56415
Hasil = 2
PS C:\Users\khaerulilm
```

Kesimpulan :

konversi tipe data primitif dari double ke short menggunakan casting. Dalam program ini, sebuah nilai double (f) diinisialisasi dengan nilai 2.32323, kemudian nilai f dicoba di-assign ke variabel short (s) dengan menggunakan casting (short).

Latihan Praktikum 2b

Percobaan 10 : Unary operator 1 – Increment Decrement

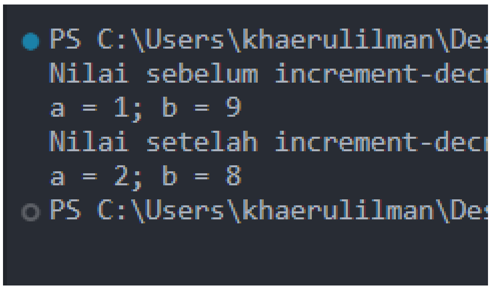


```
public class IncDec {
    public static void main(String[] args) {
        int a = 1, b = 9;
        System.out.println("Nilai sebelum increment-decrement:");
        System.out.println("a = " + a + "; b = " + b);
        a = ++a;
        b = --b;
        System.out.println("Nilai setelah increment-decrement:");
        System.out.println("a = " + a + "; b = " + b);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

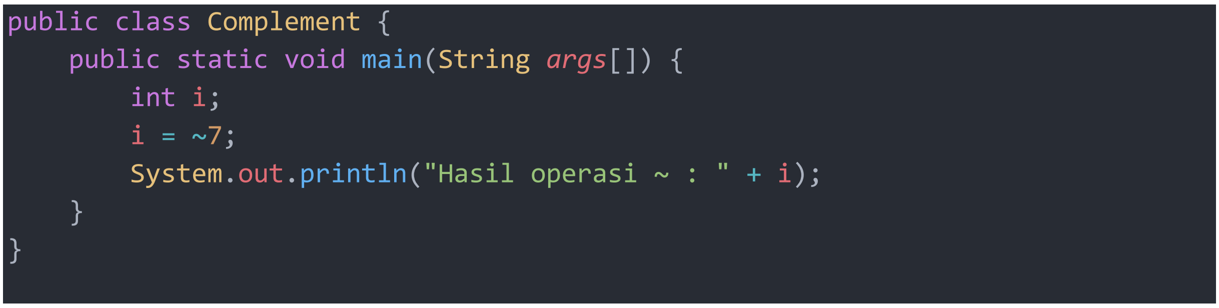


```
● PS C:\Users\khaerulilman\De
  Nilai sebelum increment-dec
  a = 1; b = 9
  Nilai setelah increment-dec
  a = 2; b = 8
○ PS C:\Users\khaerulilman\De
```

Kesimpulan :

Program ini menunjukkan penggunaan operator increment (++) dan decrement (--).

Percobaan 11 : Unary operator 2 – Complement

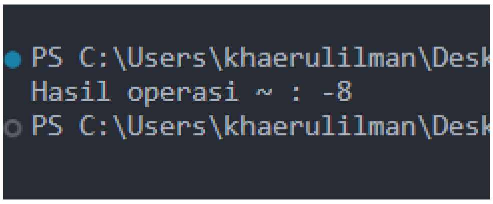


```java
public class Complement {
    public static void main(String args[]) {
        int i;
        i = ~7;
        System.out.println("Hasil operasi ~ : " + i);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

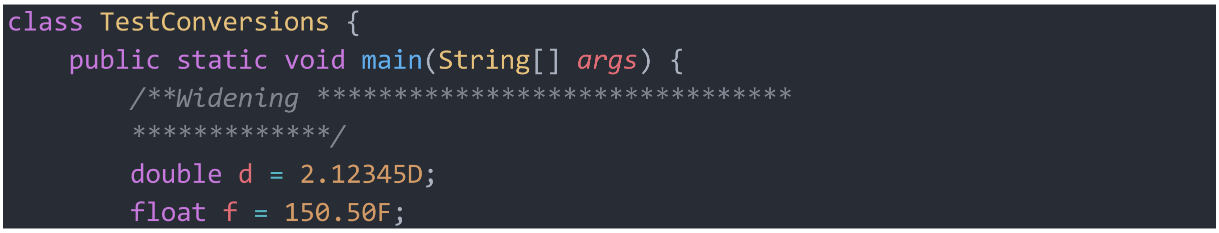


```
PS C:\Users\khaerulilman\Desk
Hasil operasi ~ : -8
PS C:\Users\khaerulilman\Desk
```

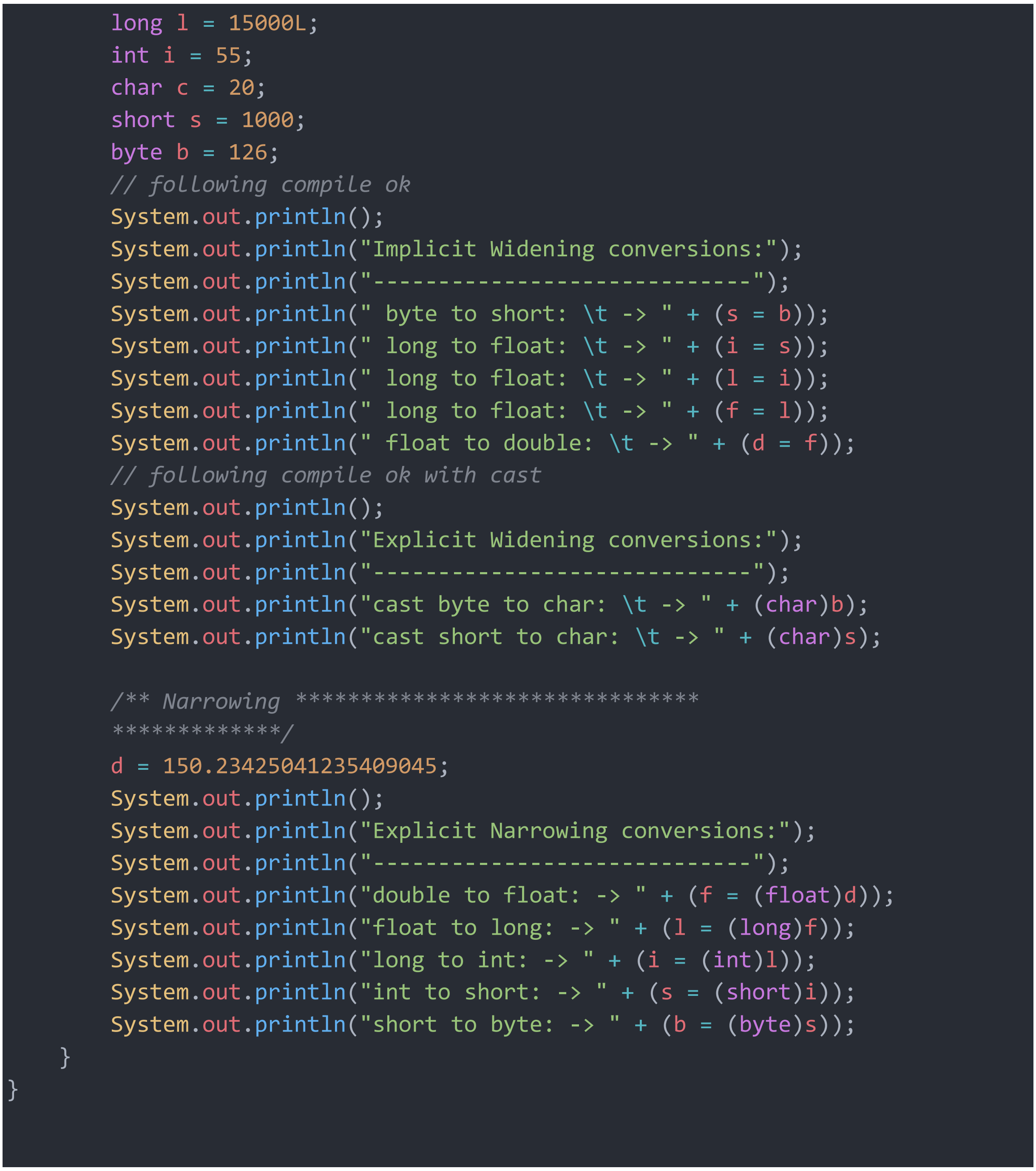
Kesimpulan :

menunjukkan penggunaan operator bitwise complement (~). Dalam program ini, sebuah variabel integer i diinisialisasi dengan hasil dari operasi bitwise complement dari nilai 7.

Percobaan 12 : Unary operator 3 – Type Cast



```java
class TestConversions {
    public static void main(String[] args) {
        /**Widening *******************************
        *************/
        double d = 2.12345D;
        float f = 150.50F;
```

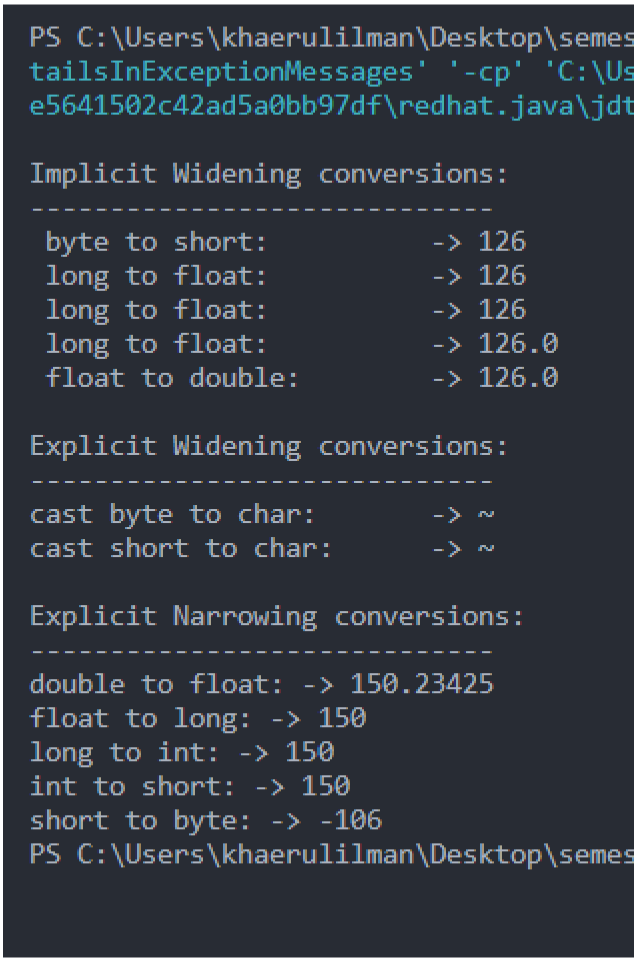


```
        long l = 15000L;
        int i = 55;
        char c = 20;
        short s = 1000;
        byte b = 126;
        // following compile ok
        System.out.println();
        System.out.println("Implicit Widening conversions:");
        System.out.println("-----------------------------");
        System.out.println(" byte to short: \t -> " + (s = b));
        System.out.println(" long to float: \t -> " + (i = s));
        System.out.println(" long to float: \t -> " + (l = i));
        System.out.println(" long to float: \t -> " + (f = l));
        System.out.println(" float to double: \t -> " + (d = f));
        // following compile ok with cast
        System.out.println();
        System.out.println("Explicit Widening conversions:");
        System.out.println("-----------------------------");
        System.out.println("cast byte to char: \t -> " + (char)b);
        System.out.println("cast short to char: \t -> " + (char)s);

        /** Narrowing *******************************
        *************/
        d = 150.234250412354O9045;
        System.out.println();
        System.out.println("Explicit Narrowing conversions:");
        System.out.println("-----------------------------");
        System.out.println("double to float: -> " + (f = (float)d));
        System.out.println("float to long: -> " + (l = (long)f));
        System.out.println("long to int: -> " + (i = (int)l));
        System.out.println("int to short: -> " + (s = (short)i));
        System.out.println("short to byte: -> " + (b = (byte)s));
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :



```
PS C:\Users\khaerulilman\Desktop\semes
tailsInExceptionMessages' '-cp' 'C:\Us
e5641502c42ad5a0bb97df\redhat.java\jdt

Implicit Widening conversions:
-----------------------------
 byte to short:          -> 126
 long to float:          -> 126
 long to float:          -> 126
 long to float:          -> 126.0
 float to double:        -> 126.0

Explicit Widening conversions:
-----------------------------
cast byte to char:       -> ~
cast short to char:      -> ~

Explicit Narrowing conversions:
-----------------------------
double to float: -> 150.23425
float to long: -> 150
long to int: -> 150
int to short: -> 150
short to byte: -> -106
PS C:\Users\khaerulilman\Desktop\semes
```

Kesimpulan :

Kesi konversi tipe data secara implisit (widening) dan eksplisit (narrowing). Dalam program ini, berbagai jenis konversi tipe data dilakukan antara tipe data numerik primitif seperti byte, short, int, long, float, dan double, serta char.mpulan :

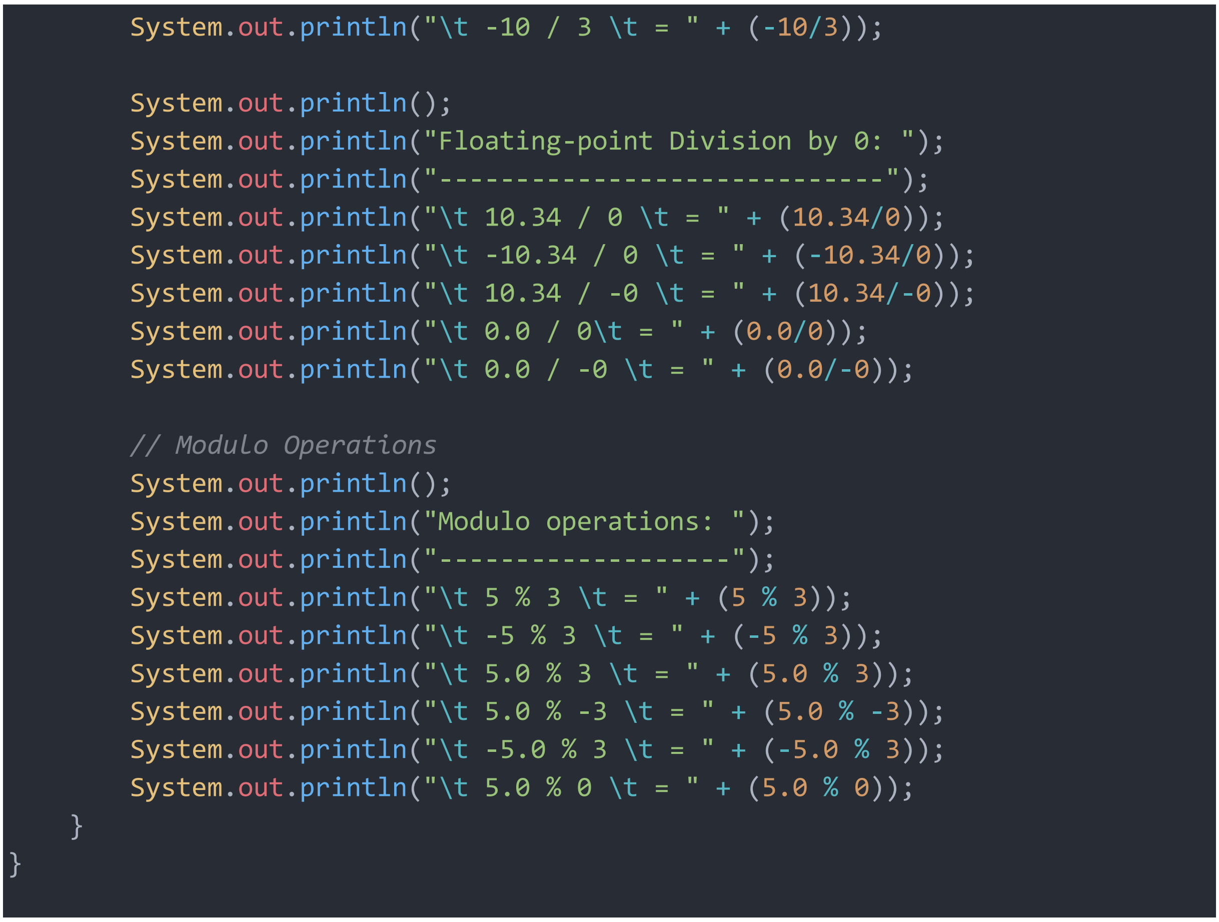
Percobaan 13 : Arithmetic operator

```
    public static void main(String[] args) {
        // Java Division Operations
        System.out.println();
        System.out.println("Integer Division - results truncated:");
        System.out.println("------------------------------------");
        System.out.println("\t 10 / 3 \t = " + (10/3));
        System.out.println("\t 10 / -3 \t = " + (10/-3));
        System.out.println("\t -10 / 3 \t = " + (-10/3));

        System.out.println();
        System.out.println("Floating-point Division by 0: ");
        System.out.println("-----------------------------");
        System.out.println("\t 10.34 / 0 \t = " + (10.34/0));
        System.out.println("\t -10.34 / 0 \t = " + (-10.34/0));
        System.out.println("\t 10.34 / -0 \t = " + (10.34/-0));
        System.out.println("\t 0.0 / 0\t = " + (0.0/0));
        System.out.println("\t 0.0 / -0 \t = " + (0.0/-0));

        // Modulo Operations
        System.out.println();
        System.out.println("Modulo operations: ");
        System.out.println("-------------------");
        System.out.println("\t 5 % 3 \t = " + (5 % 3));
        System.out.println("\t -5 % 3 \t = " + (-5 % 3));
        System.out.println("\t 5.0 % 3 \t = " + (5.0 % 3));
        System.out.println("\t 5.0 % -3 \t = " + (5.0 % -3));
        System.out.println("\t -5.0 % 3 \t = " + (-5.0 % 3));
        System.out.println("\t 5.0 % 0 \t = " + (5.0 % 0));
    }
```

Catatan kesalahan :

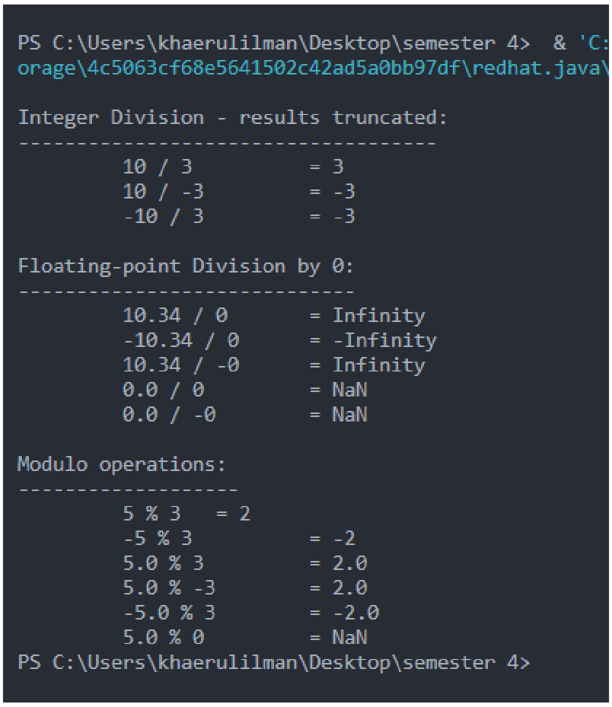
- tidak memberi deklarasi class sebelum memulai public static void main(String[] args)

```
class Arithmetic {
    public static void main(String[] args) {
        // Java Division Operations
        System.out.println();
        System.out.println("Integer Division - results truncated:");
        System.out.println("------------------------------------");
        System.out.println("\t 10 / 3 \t = " + (10/3));
        System.out.println("\t 10 / -3 \t = " + (10/-3));
```



```java
        System.out.println("\t -10 / 3 \t = " + (-10/3));

        System.out.println();
        System.out.println("Floating-point Division by 0: ");
        System.out.println("-----------------------------");
        System.out.println("\t 10.34 / 0 \t = " + (10.34/0));
        System.out.println("\t -10.34 / 0 \t = " + (-10.34/0));
        System.out.println("\t 10.34 / -0 \t = " + (10.34/-0));
        System.out.println("\t 0.0 / 0\t = " + (0.0/0));
        System.out.println("\t 0.0 / -0 \t = " + (0.0/-0));

        // Modulo Operations
        System.out.println();
        System.out.println("Modulo operations: ");
        System.out.println("-------------------");
        System.out.println("\t 5 % 3 \t = " + (5 % 3));
        System.out.println("\t -5 % 3 \t = " + (-5 % 3));
        System.out.println("\t 5.0 % 3 \t = " + (5.0 % 3));
        System.out.println("\t 5.0 % -3 \t = " + (5.0 % -3));
        System.out.println("\t -5.0 % 3 \t = " + (-5.0 % 3));
        System.out.println("\t 5.0 % 0 \t = " + (5.0 % 0));
    }
}
```

Output :

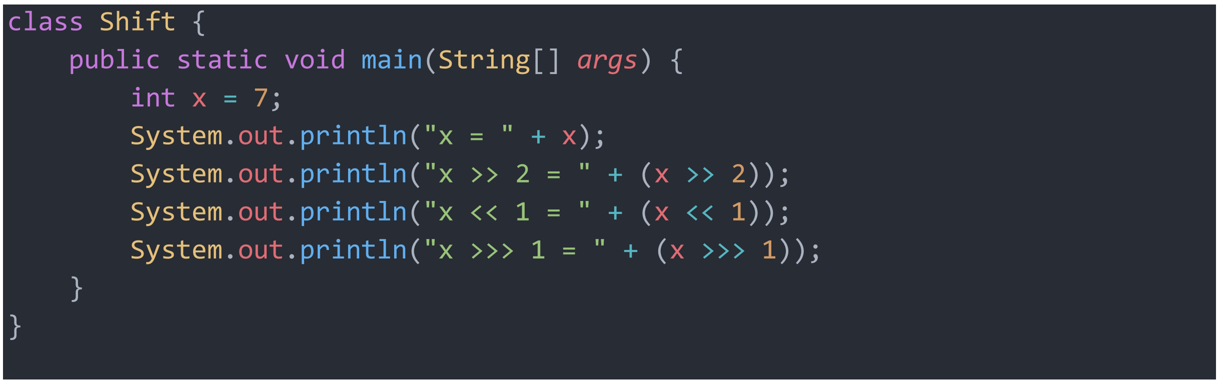


```
PS C:\Users\khaerulilman\Desktop\semester 4>  & 'C:
orage\4c5063cf68e5641502c42ad5a0bb97df\redhat.java\

Integer Division - results truncated:
------------------------------------
         10 / 3          = 3
         10 / -3         = -3
         -10 / 3         = -3

Floating-point Division by 0:
-----------------------------
         10.34 / 0       = Infinity
         -10.34 / 0      = -Infinity
         10.34 / -0      = Infinity
         0.0 / 0         = NaN
         0.0 / -0        = NaN

Modulo operations:
-------------------
         5 % 3    = 2
         -5 % 3          = -2
         5.0 % 3         = 2.0
         5.0 % -3        = 2.0
         -5.0 % 3        = -2.0
         5.0 % 0         = NaN
PS C:\Users\khaerulilman\Desktop\semester 4>
```

Kesimpulan :

program Java yang menunjukkan operasi aritmatika dasar seperti pembagian dan modulo. Program ini juga menunjukkan hasil dari beberapa operasi yang melibatkan pembagian dengan nol, baik untuk tipe data integer maupun floating-point.

Percobaan 14 : Shift operator

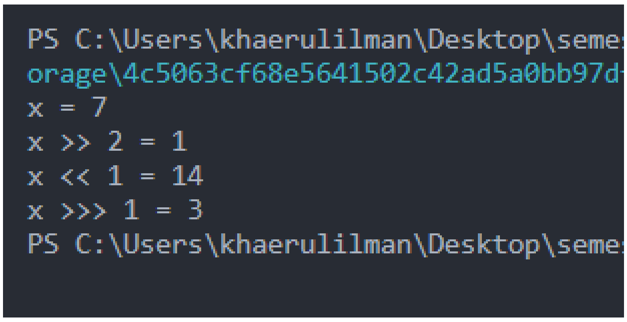


```
class Shift {
    public static void main(String[] args) {
        int x = 7;
        System.out.println("x = " + x);
        System.out.println("x >> 2 = " + (x >> 2));
        System.out.println("x << 1 = " + (x << 1));
        System.out.println("x >>> 1 = " + (x >>> 1));
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

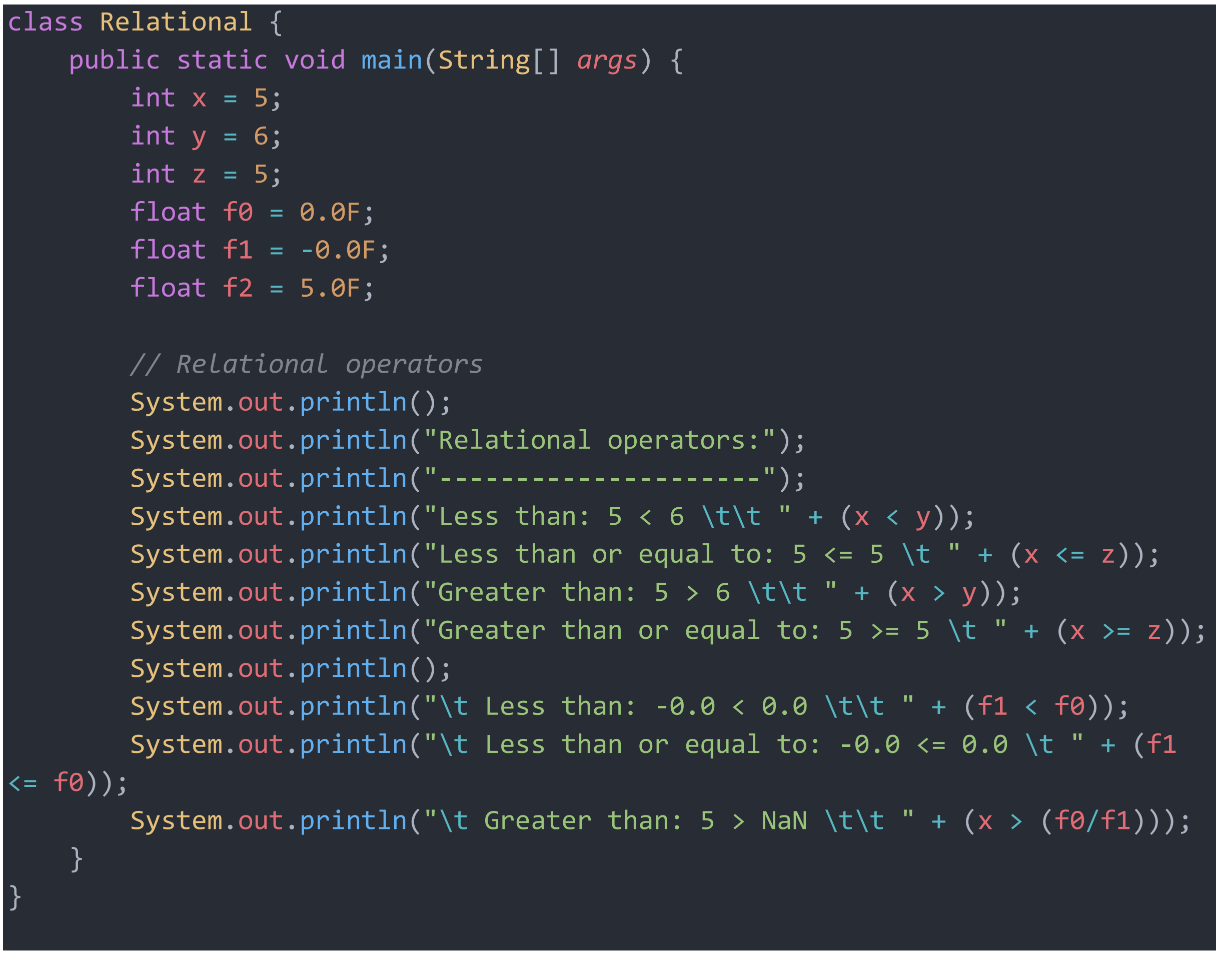


```
PS C:\Users\khaerulilman\Desktop\seme
orage\4c5063cf68e5641502c42ad5a0bb97d
x = 7
x >> 2 = 1
x << 1 = 14
x >>> 1 = 3
PS C:\Users\khaerulilman\Desktop\seme
```

Kesimpulan :

contoh program Java yang menunjukkan penggunaan operator shift (>>, <<, >>>) dalam operasi bitwise pada tipe data integer.

Percobaan 15 : Comparison operator 1 – Relational

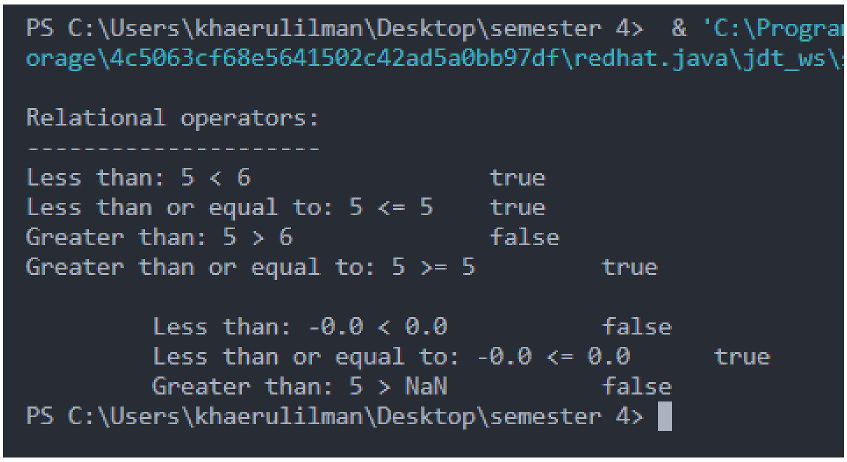


```java
class Relational {
    public static void main(String[] args) {
        int x = 5;
        int y = 6;
        int z = 5;
        float f0 = 0.0F;
        float f1 = -0.0F;
        float f2 = 5.0F;

        // Relational operators
        System.out.println();
        System.out.println("Relational operators:");
        System.out.println("---------------------");
        System.out.println("Less than: 5 < 6 \t\t " + (x < y));
        System.out.println("Less than or equal to: 5 <= 5 \t " + (x <= z));
        System.out.println("Greater than: 5 > 6 \t\t " + (x > y));
        System.out.println("Greater than or equal to: 5 >= 5 \t " + (x >= z));
        System.out.println();
        System.out.println("\t Less than: -0.0 < 0.0 \t\t " + (f1 < f0));
        System.out.println("\t Less than or equal to: -0.0 <= 0.0 \t " + (f1
<= f0));
        System.out.println("\t Greater than: 5 > NaN \t\t " + (x > (f0/f1)));
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

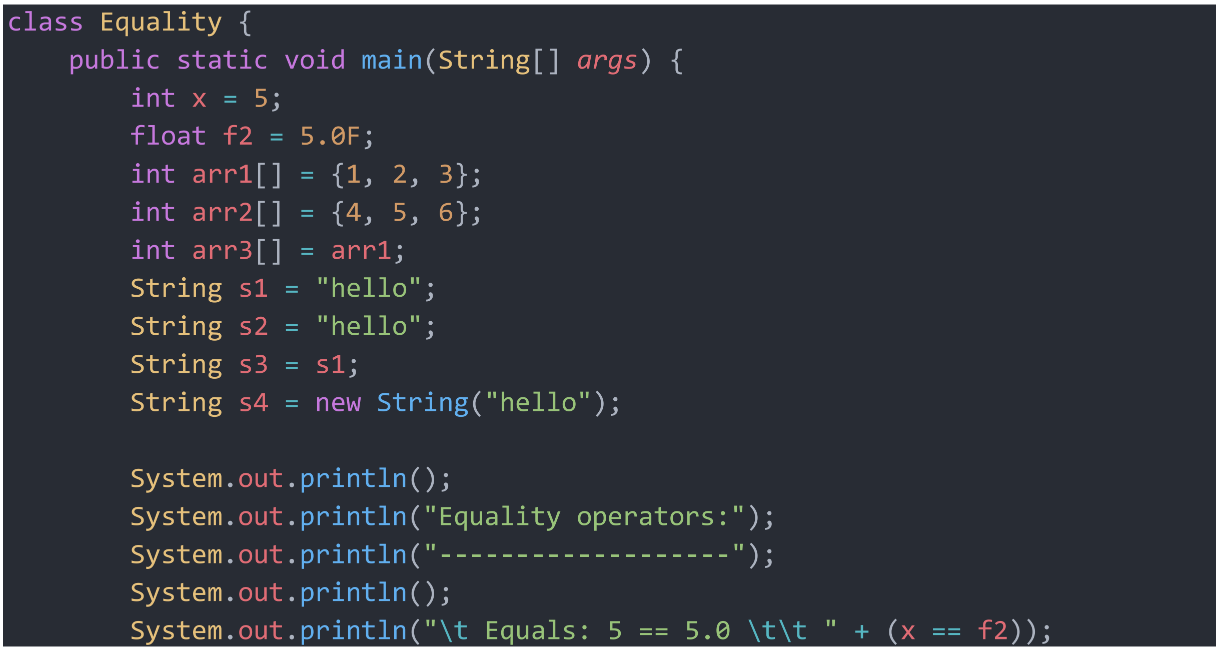


```
PS C:\Users\khaerulilman\Desktop\semester 4>  & 'C:\Progra
orage\4c5063cf68e5641502c42ad5a0bb97df\redhat.java\jdt_ws\

Relational operators:
---------------------
Less than: 5 < 6                 true
Less than or equal to: 5 <= 5    true
Greater than: 5 > 6              false
Greater than or equal to: 5 >= 5         true

         Less than: -0.0 < 0.0           false
         Less than or equal to: -0.0 <= 0.0      true
         Greater than: 5 > NaN           false
PS C:\Users\khaerulilman\Desktop\semester 4>
```

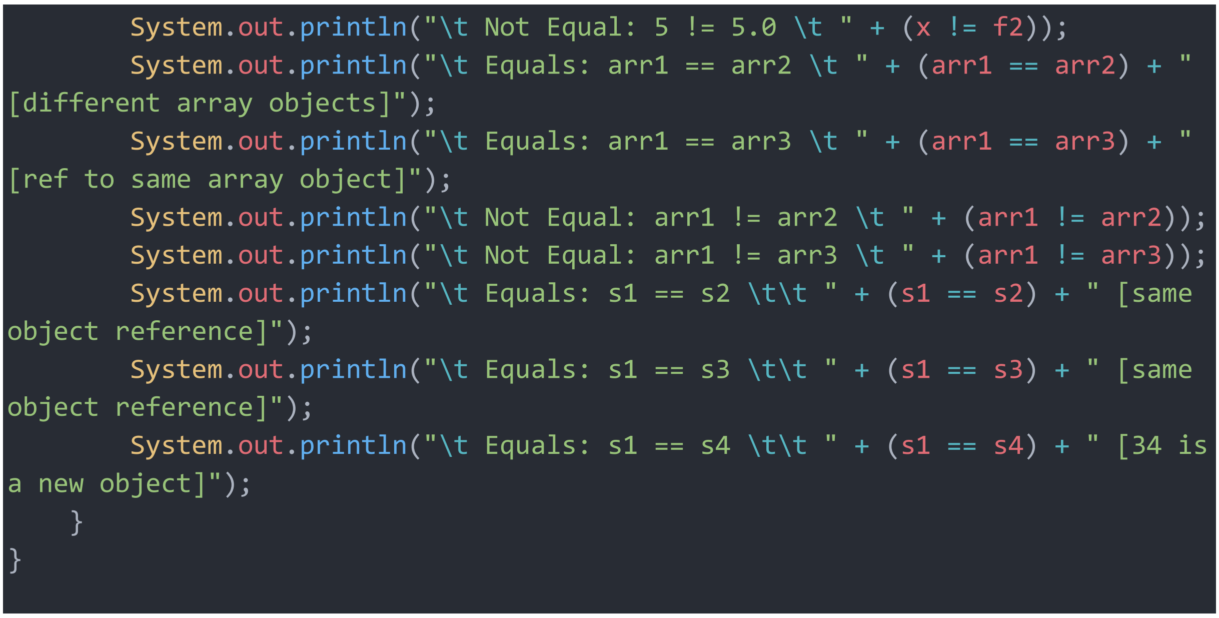
Kesimpulan :

menunjukkan penggunaan operator relasional (<, <=, >, >=) pada tipe data integer dan floating-point. Dalam program ini, beberapa variabel diinisialisasi dengan nilai integer dan floating-point, dan kemudian dilakukan operasi perbandingan menggunakan operator relasional.

Percobaan 16 : Comparison operator 2 – Equality



```java
class Equality {
    public static void main(String[] args) {
        int x = 5;
        float f2 = 5.0F;
        int arr1[] = {1, 2, 3};
        int arr2[] = {4, 5, 6};
        int arr3[] = arr1;
        String s1 = "hello";
        String s2 = "hello";
        String s3 = s1;
        String s4 = new String("hello");

        System.out.println();
        System.out.println("Equality operators:");
        System.out.println("-------------------");
        System.out.println();
        System.out.println("\t Equals: 5 == 5.0 \t\t " + (x == f2));
```

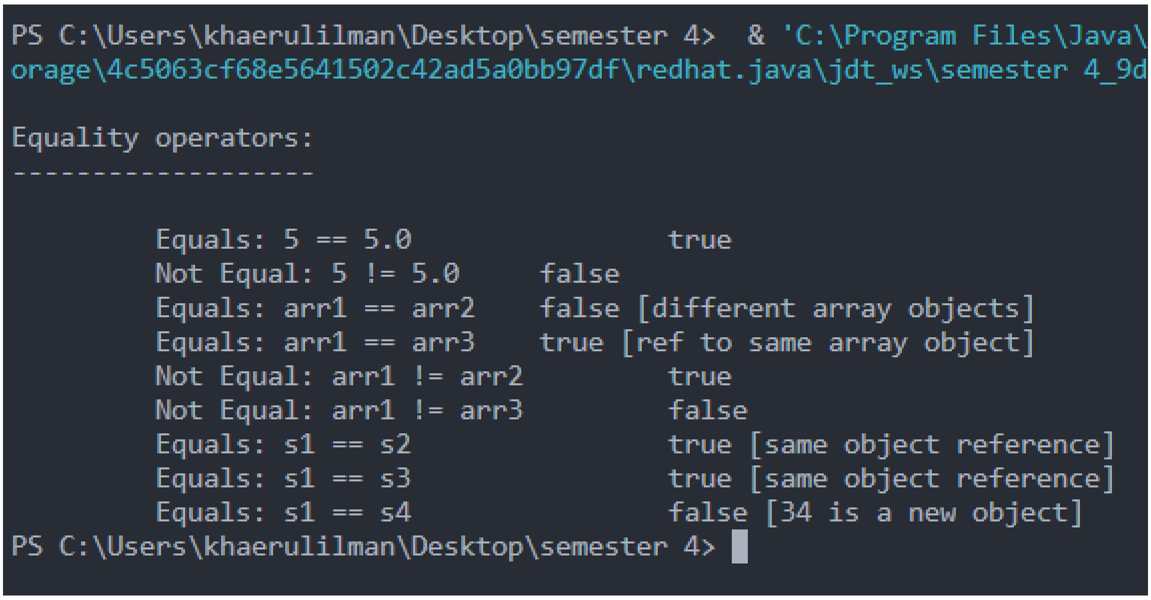


```
        System.out.println("\t Not Equal: 5 != 5.0 \t " + (x != f2));
        System.out.println("\t Equals: arr1 == arr2 \t " + (arr1 == arr2) + "
[different array objects]");
        System.out.println("\t Equals: arr1 == arr3 \t " + (arr1 == arr3) + "
[ref to same array object]");
        System.out.println("\t Not Equal: arr1 != arr2 \t " + (arr1 != arr2));
        System.out.println("\t Not Equal: arr1 != arr3 \t " + (arr1 != arr3));
        System.out.println("\t Equals: s1 == s2 \t\t " + (s1 == s2) + " [same
object reference]");
        System.out.println("\t Equals: s1 == s3 \t\t " + (s1 == s3) + " [same
object reference]");
        System.out.println("\t Equals: s1 == s4 \t\t " + (s1 == s4) + " [34 is
a new object]");
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :



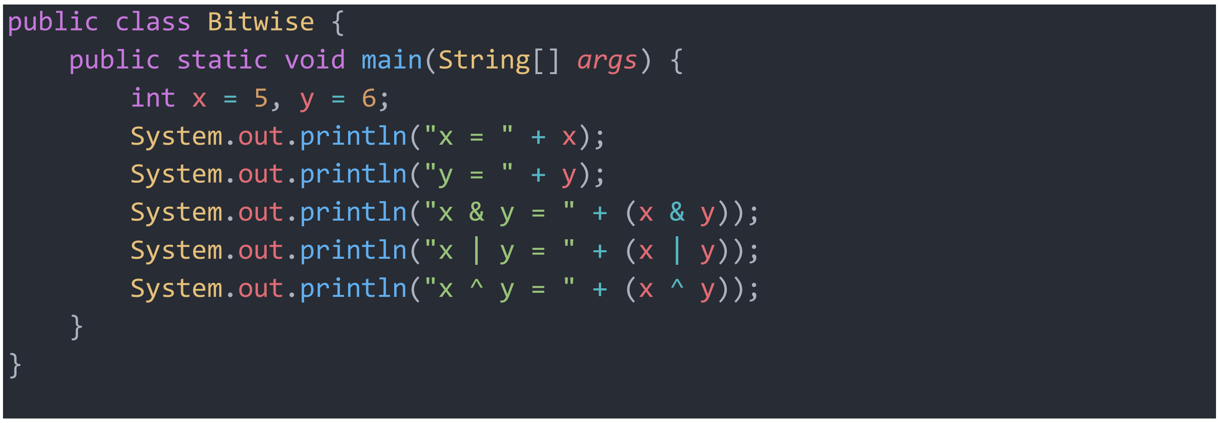
```
PS C:\Users\khaerulilman\Desktop\semester 4>  & 'C:\Program Files\Java\
orage\4c5063cf68e5641502c42ad5a0bb97df\redhat.java\jdt_ws\semester 4_9d

Equality operators:
-------------------

         Equals: 5 == 5.0                true
         Not Equal: 5 != 5.0     false
         Equals: arr1 == arr2    false [different array objects]
         Equals: arr1 == arr3    true [ref to same array object]
         Not Equal: arr1 != arr2         true
         Not Equal: arr1 != arr3         false
         Equals: s1 == s2                true [same object reference]
         Equals: s1 == s3                true [same object reference]
         Equals: s1 == s4                false [34 is a new object]
PS C:\Users\khaerulilman\Desktop\semester 4> 
```

Kesimpulan :

menunjukkan penggunaan operator kesetaraan (==) dan ketidaksamaan (!=) untuk membandingkan kesetaraan antara nilai-nilai atau objek-objek dalam Java. Dalam program ini, beberapa variabel diinisialisasi dengan nilai-nilai dan objek-objek, dan kemudian dilakukan operasi perbandingan menggunakan operator kesetaraan dan ketidaksamaan.

Latihan Praktikum 2c

Percobaan 17 : Bitwise operator

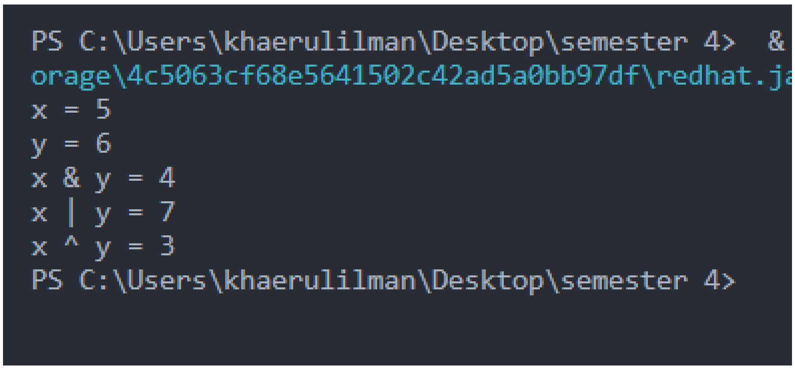


```java
public class Bitwise {
    public static void main(String[] args) {
        int x = 5, y = 6;
        System.out.println("x = " + x);
        System.out.println("y = " + y);
        System.out.println("x & y = " + (x & y));
        System.out.println("x | y = " + (x | y));
        System.out.println("x ^ y = " + (x ^ y));
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

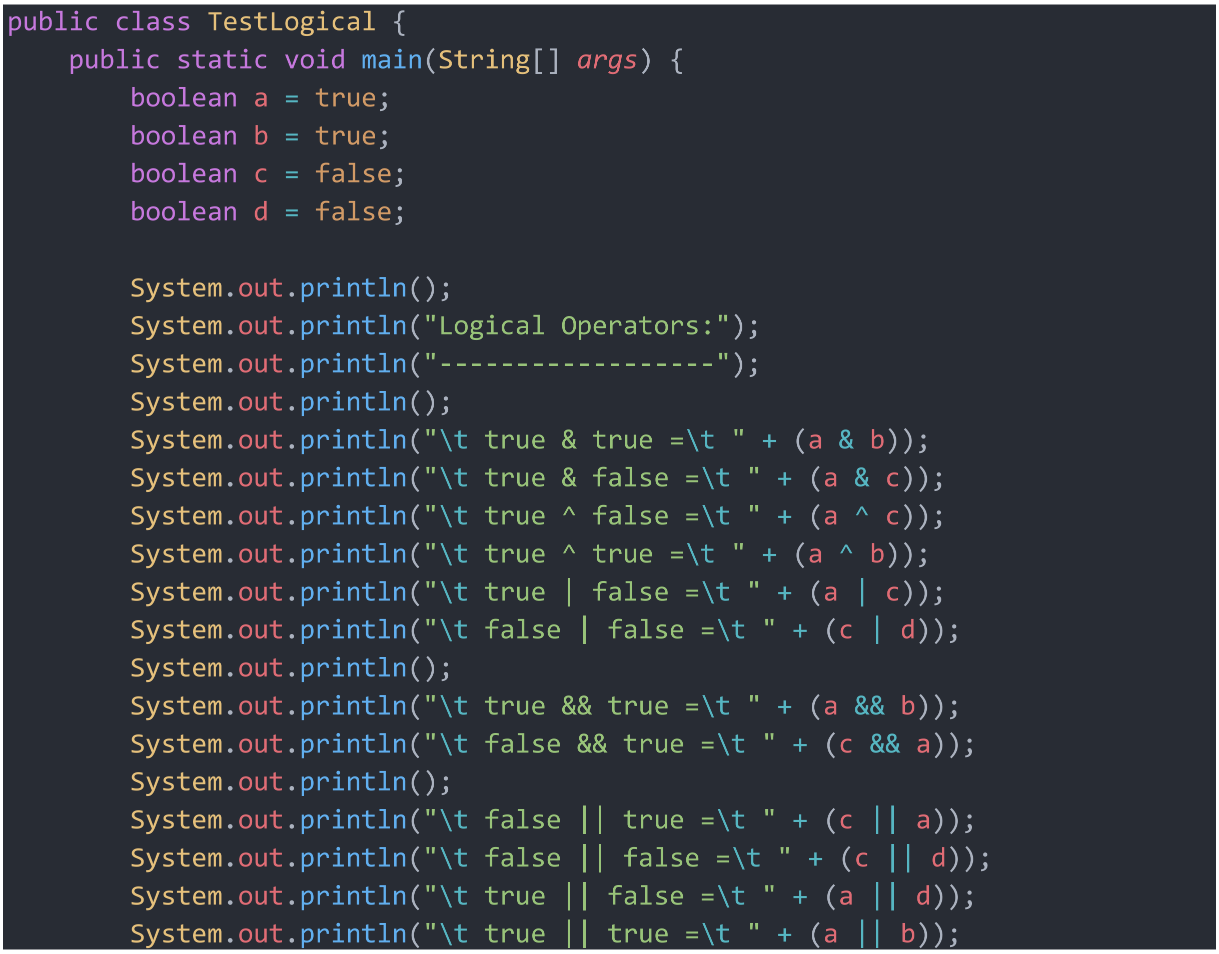


```
PS C:\Users\khaerulilman\Desktop\semester 4>  &
orage\4c5063cf68e5641502c42ad5a0bb97df\redhat.ja
x = 5
y = 6
x & y = 4
x | y = 7
x ^ y = 3
PS C:\Users\khaerulilman\Desktop\semester 4>
```

Kesimpulan :

menunjukkan penggunaan operator bitwise (&, |, ^) untuk melakukan operasi logika pada level bit dari dua nilai integer

Percobaan 18 : Logical operator



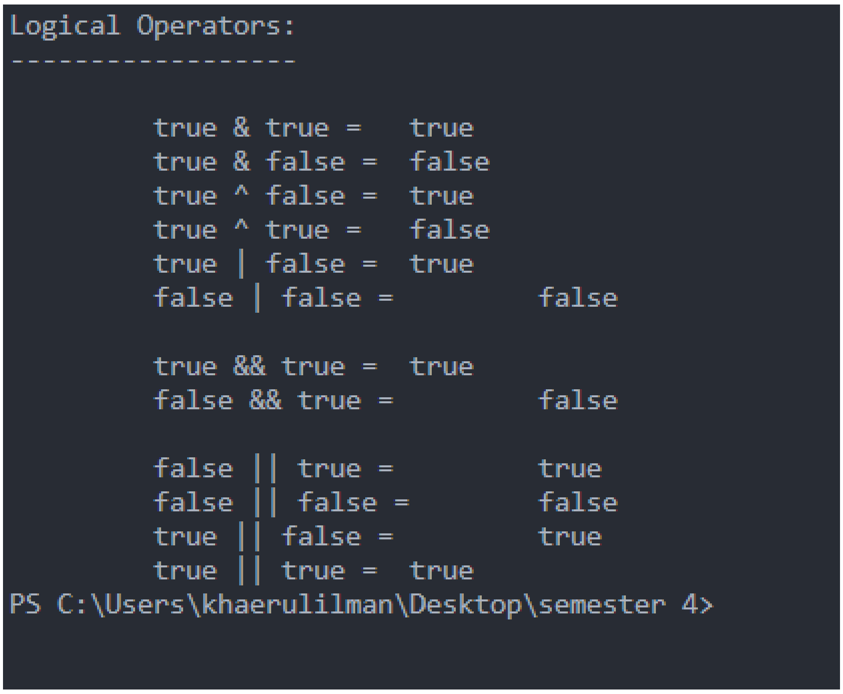
```java
public class TestLogical {
    public static void main(String[] args) {
        boolean a = true;
        boolean b = true;
        boolean c = false;
        boolean d = false;

        System.out.println();
        System.out.println("Logical Operators:");
        System.out.println("------------------");
        System.out.println();
        System.out.println("\t true & true =\t " + (a & b));
        System.out.println("\t true & false =\t " + (a & c));
        System.out.println("\t true ^ false =\t " + (a ^ c));
        System.out.println("\t true ^ true =\t " + (a ^ b));
        System.out.println("\t true | false =\t " + (a | c));
        System.out.println("\t false | false =\t " + (c | d));
        System.out.println();
        System.out.println("\t true && true =\t " + (a && b));
        System.out.println("\t false && true =\t " + (c && a));
        System.out.println();
        System.out.println("\t false || true =\t " + (c || a));
        System.out.println("\t false || false =\t " + (c || d));
        System.out.println("\t true || false =\t " + (a || d));
        System.out.println("\t true || true =\t " + (a || b));
```

```
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

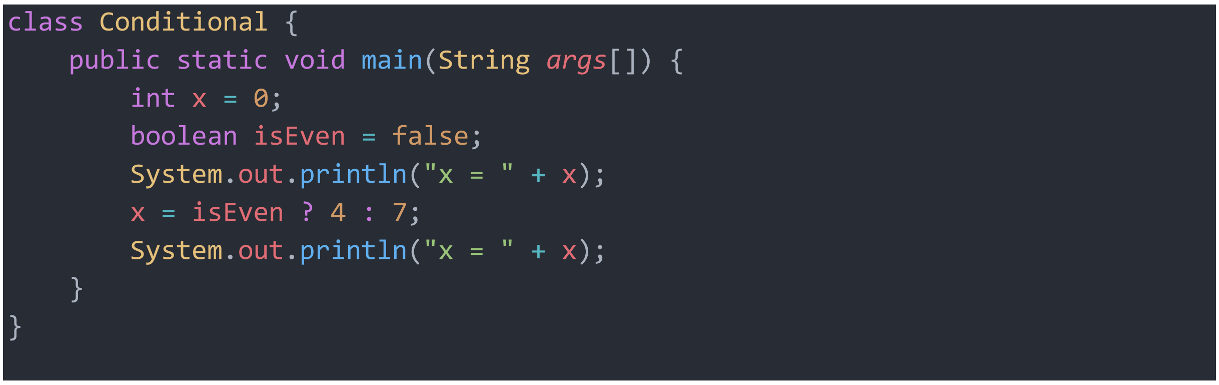


```
Logical Operators:
------------------

         true & true =   true
         true & false =  false
         true ^ false =  true
         true ^ true =   false
         true | false =  true
         false | false =         false

         true && true =  true
         false && true =         false

         false || true =         true
         false || false =        false
         true || false =         true
         true || true =  true
PS C:\Users\khaerulilman\Desktop\semester 4>
```

Kesimpulan :

menunjukkan penggunaan operator logika (&, |, ^, &&, ||) pada tipe data boolean.

Percobaan 19 : Conditional operator

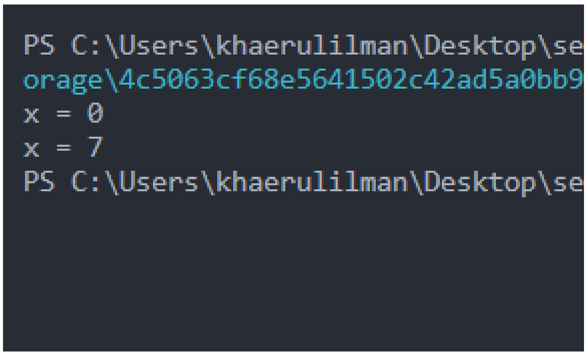


```
class Conditional {
    public static void main(String args[]) {
        int x = 0;
        boolean isEven = false;
        System.out.println("x = " + x);
        x = isEven ? 4 : 7;
        System.out.println("x = " + x);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :

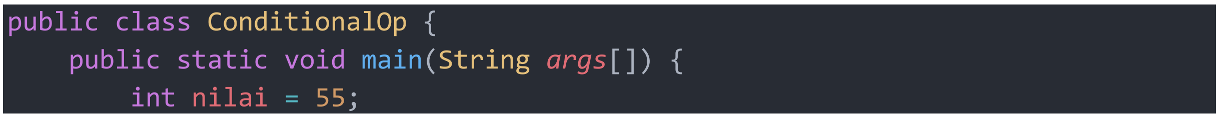


```
PS C:\Users\khaerulilman\Desktop\se
orage\4c5063cf68e5641502c42ad5a0bb9
x = 0
x = 7
PS C:\Users\khaerulilman\Desktop\se
```

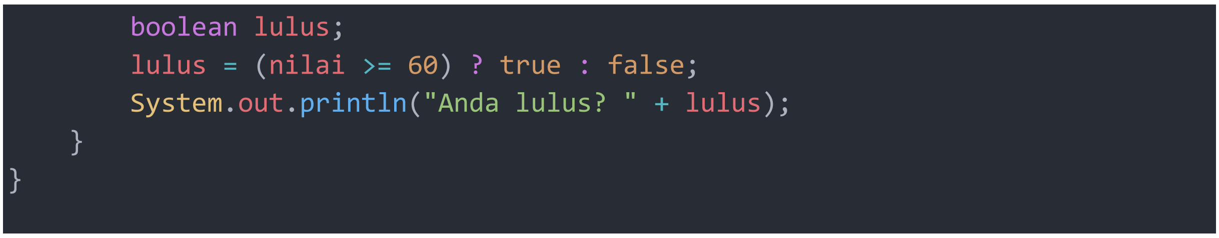
Kesimpulan :

program Java yang menunjukkan penggunaan operator ternary (? :) untuk melakukan ekspresi kondisional.

Percobaan 20 : Conditional operator



```
public class ConditionalOp {
    public static void main(String args[]) {
        int nilai = 55;
```

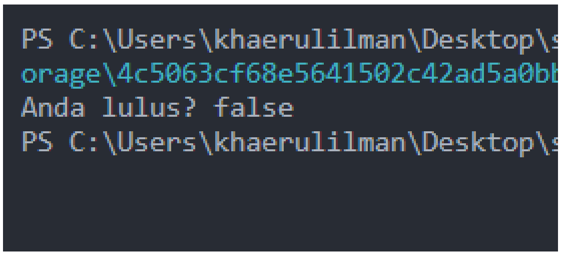


```
        boolean lulus;
        lulus = (nilai >= 60) ? true : false;
        System.out.println("Anda lulus? " + lulus);
    }
}
```

Catatan kesalahan :

Program benar tidak ada kesalahan program

Output :



```
PS C:\Users\khaerulilman\Desktop\
orage\4c5063cf68e5641502c42ad5a0b
Anda lulus? false
PS C:\Users\khaerulilman\Desktop\
```

Kesimpulan :

menunjukkan penggunaan operator ternary (? :) untuk menentukan kondisi lulus atau tidak lulus berdasarkan nilai yang diberikan.