

Tugas

Makassar, 16 Maret 2021

PEMROGRAMAN BERORIENTASI OBJEK



NUGRAHA WANASPATI

13020190261

A5

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MUSLIM INDONESIA

MAKASSAR

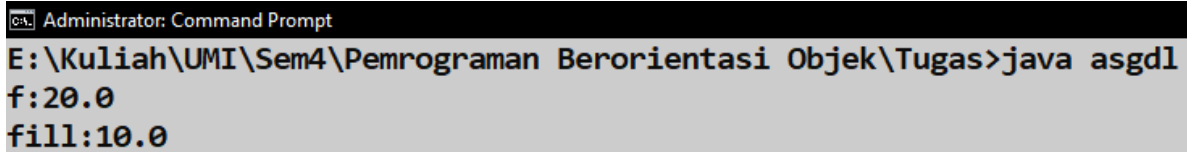
2021

```

1.class asgdl{
    public static void main(String[]args){
        float f=20.0f;
        double fill;
        fill=10.0f;
        System.out.println("f:"+f+"\nfill:"+fill);
    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java asgdl
f:20.0
fill:10.0

```

Penjelasan:

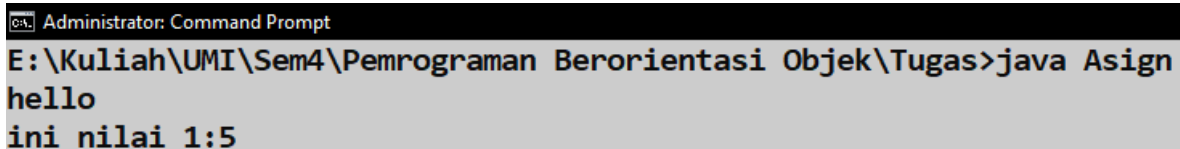
Program diatas tentang bagaimana cara merepresentasikan tipe data float dan double kemudian menampilkannya di layar.

```

2.class Asign{
    public static void main(String[]args){
        int i;
        i=5;
        System.out.print("hello\n");
        System.out.print("ini nilai 1:"+i);
    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Asign
hello
ini nilai 1:5

```

Penjelasan:

Program diatas tentang cara menampilkan teks dan angka dilayar dengan menggunakan tipe data int dan perintah System.out.print.

```

3.class game{
    public static void main(String[]args){
        /*deklarasi variabel*/
        short a=1;
        int b=100;
    }
}

```

```

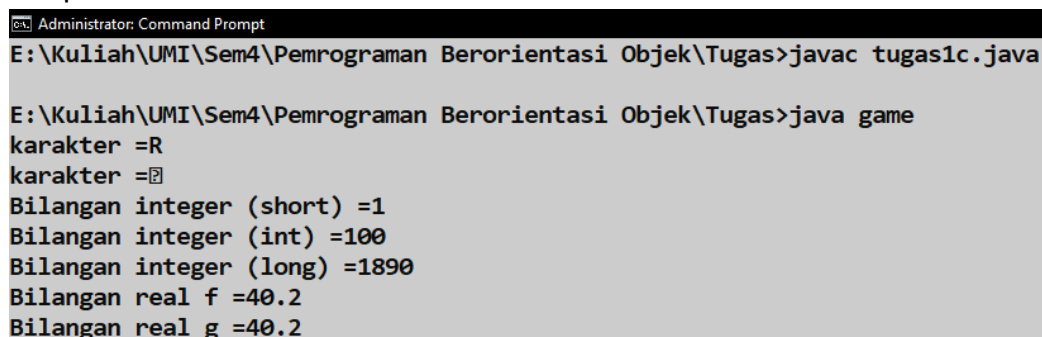
        long c=1890;
        char d=30;
        char e='R';
        double f=40.2d;
        float g=40.2f;

        System.out.println("karakter =" +e);//karakter sebagai
karakter
        System.out.println("karakter =" +d);//karakter sebagai
integer
        System.out.println("Bilangan integer (short)
="+a);//bilangan bulat dengan tipe data short
        System.out.println("Bilangan integer (int) =" +b);//bilangan
bulat dengan tipe data int
        System.out.println("Bilangan integer (long) =" +c);//bilangan
bulat dengan tipe data long
        System.out.println("Bilangan real f =" +f); //bilangan real
dengan tipe data double
        System.out.println("Bilangan real g =" +g);//bilangan real
dengan tipe data float

    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1c.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java game
karakter =R
karakter =30
Bilangan integer (short) =1
Bilangan integer (int) =100
Bilangan integer (long) =1890
Bilangan real f =40.2
Bilangan real g =40.2

```

Penjelasan:

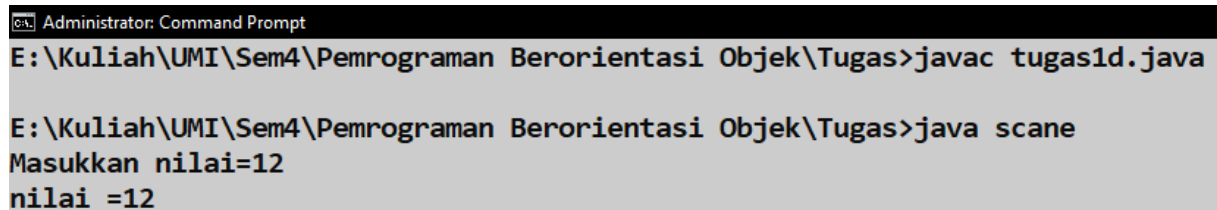
Program diatas tentang cara menginisialisasi tipe data diantaranya: karakter dengan variable yang bertipe data char, karakter dengan variable yang bertipe data int, bilangan integer dengan variable yang bertipe data short, bilangan integer dengan variable yang bertipe data int, bilangan integer dengan variable yang bertipe data long, bilangan real dengan variable yang bertipe data double dan bilangan real yang bertipe data float. Perlu di garis bawahi bahwa apabila kita menginisialisasi karakter dengan tipe data int tanpa menggunakan tanda koma atas maka yang akan tampil di layar itu tanda tanya.

```

4. import java.util.Scanner;
   class scane{
       public static void main(String[]args){
           Scanner scan=new Scanner(System.in);
           int a;
           System.out.print("Masukkan nilai=");
           a=scan.nextInt();
           System.out.print("nilai =" +a);
       }
   }

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1d.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java scane
Masukkan nilai=12
nilai =12

```

Penjelasan:

Program di atas tentang bagaimana cara menginputkan data melalui user.

```

5. import java.io.BufferedReader;
   import java.io.IOException;
   import java.io.InputStreamReader;
   //import javax.swing.*;
   class Baca {
       public static void main(String[] args) throws IOException {
           char cc;
           int bil;
           InputStreamReader isr = new InputStreamReader(System.in);
           BufferedReader dataIn = new BufferedReader(isr);    // atau
           BufferedReader dataAIn = new BufferedReader(new
InputStreamReader(System.in));
           /* Algoritma */
           System.out.print ("hello\n");
           System.out.print("baca 1 karakter : ");    //perintah baca
karakter cc
           cc =dataIn.readLine().charAt(0);
           System.out.print("baca 1 bilangan : ");    //perintah baca
bil

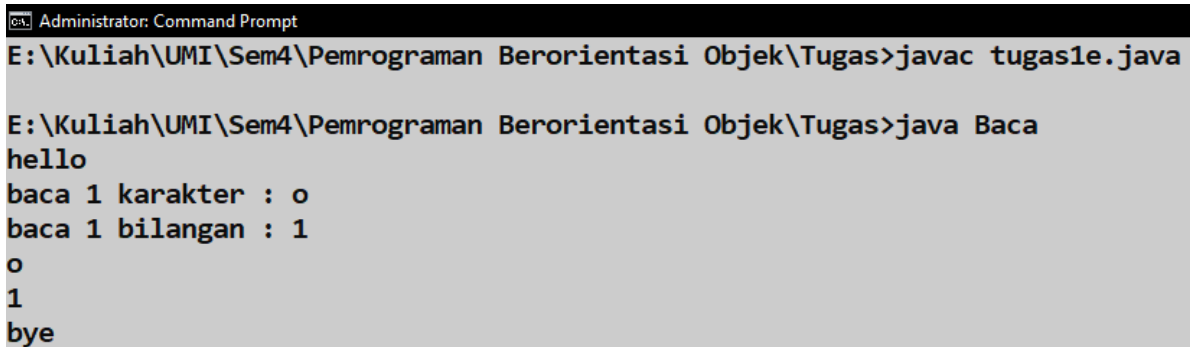
```

```

        bil = Integer.parseInt(dataIn.readLine());
        //String kar = JOptionPane.showInputDialog("Karakter 1 :
");
        //System.out.println(kar);
        //JOptionPane.showMessageDialog(null, "hello");
        System.out.print (cc +"\n" +bil+"\n");
        System.out.print ("bye \n");
    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1e.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Baca
hello
baca 1 karakter : o
baca 1 bilangan : 1
o
1
bye

```

Penjelasan:

Fungsi `BufferedReader` adalah membaca file dari input stream, lebih spesifik lagi bahwa class ini digunakan sebagai buffer dari karakter-karakter dengan tujuan membuat penanganan yang lebih efisien untuk String, character dan array. Fungsi `IOException` menjelaskan bahwa kita menggunakan kelas `BufferedReader`. Fungsi `InputStreamReader` adalah untuk memasukkan inputan ke dalam sebuah program.

```

6.class casting{
    public static void main(String[]args){
        int a=4,b=7;
        float c=5.f,d=6.3f;
        char g='h';
        double k=3.14;

        System.out.println((float)a); //int ke float//
        System.out.println((double)b); //int ke double//
        System.out.println((int)c); /*float ke int*/
        System.out.println((double)d); /*float ke double*/
        System.out.println((int)g); /*char ke int*/
        System.out.println((float)g); /*char ke float*/
        System.out.println((double)g); /* char ke double*/
    }
}

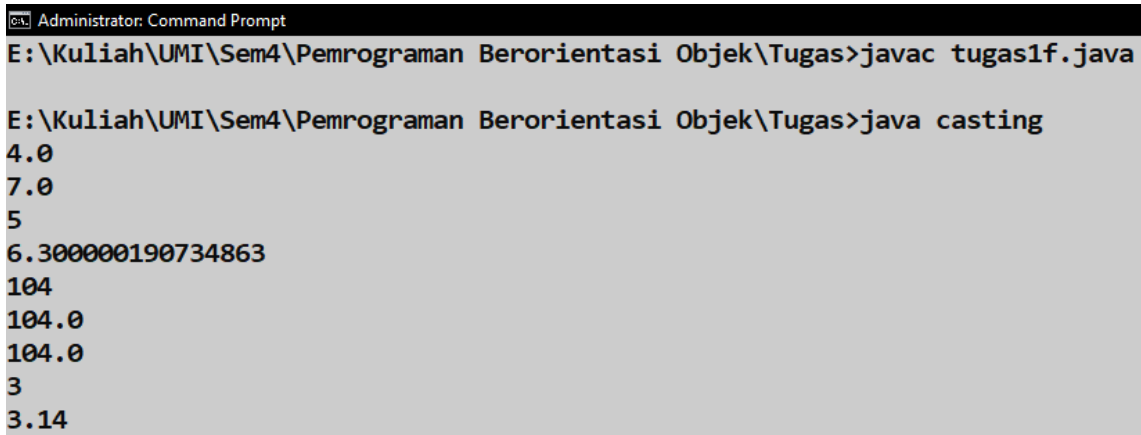
```

```

        System.out.println((int)k); /*double ke int*/
        System.out.println((float)k); /*double ke float*/
    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1f.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java casting
4.0
7.0
5
6.300000190734863
104
104.0
104.0
3
3.14

```

Penjelasan:

Program diatas tentang casting. Casting adalah suatu proses melakukan operasi antara 2 atau lebih variable yang bertipe beda.

```

7.class castinglagi{
    public static void main(String[]args){
        int a=4, b=6;
        float c=2.f, d=1.f;
        char g='5';
        double k=3.14;
        String n="65",m="45", l="100";
        /*untuk konversi tipe data karakter ke bilangan itu
menggunakan parse*/
        a=Integer.parseInt(n); /*String ke int*/
        k=Double.parseDouble(m);/*String ke double*/
        c=Float.parseFloat(l); /*String ke float*/
        System.out.println("a="+a+"\nk="+k+"\nc="+c);
        /*untuk konversi tipe data integer ke String itu menggunakan
valueOf*/
        n=String.valueOf(a); /*int ke String*/
        m=String.valueOf(g); /*karakter ke String*/
        l=String.valueOf(d); /*float ke String*/
        System.out.println("n="+a+"\nm="+m+"\nl="+l);
        /*untuk konversi bulat ke tipe data float dan double*/
    }
}

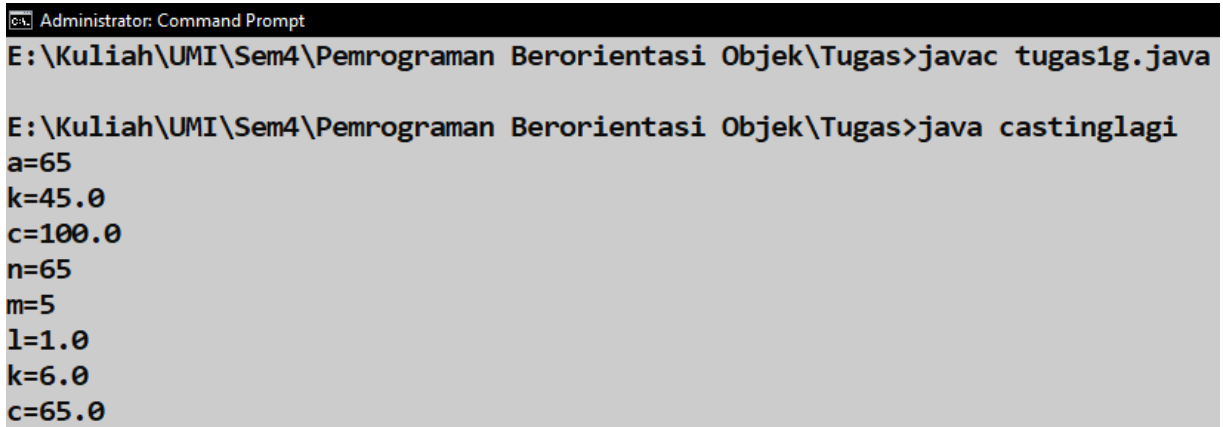
```

```

        k=Double.valueOf(b).intValue(); /*int ke double*/
        c=Float.valueOf(a).intValue(); /*int ke float*/
        System.out.println("k="+k+"\nc="+c);
    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1g.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java castinglagi
a=65
k=45.0
c=100.0
n=65
m=5
l=1.0
k=6.0
c=65.0

```

Penjelasan:

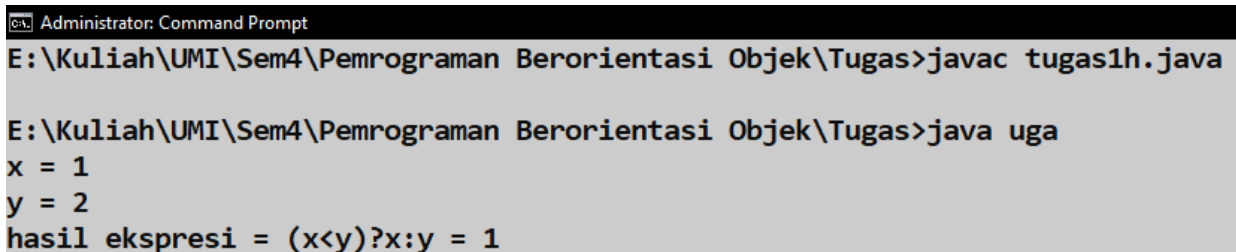
Program di atas tentang bagaimana melakukan casting dengan menggunakan kelas.

```

8. class uga{
    public static void main(String[] args) {// TODO Auto-generated
        method stub
        int x = 1;
        int y = 2;
        System.out.print("x = " + x + "\n");
        System.out.print("y = " + y + "\n");
        System.out.print("hasil ekspresi = (x<y)?x:y = " + ((x < y) ? x :
        y));
    }
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1h.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java uga
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1

```

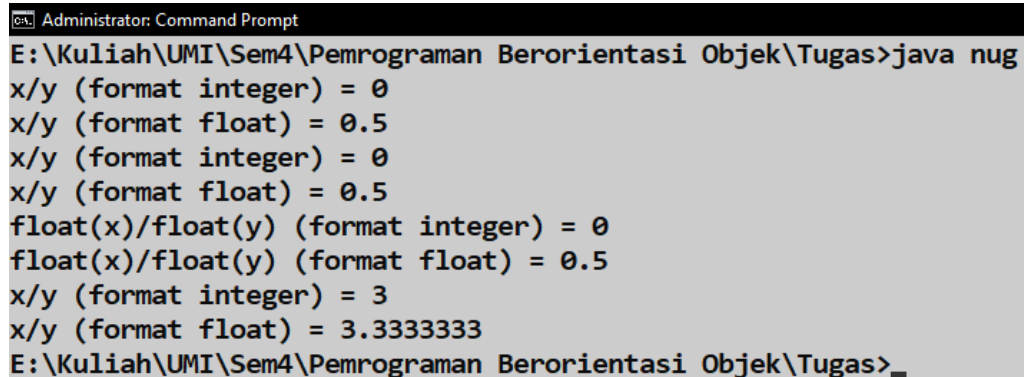
Penjelasan:

Program di atas tentang pemakaian operator kondisional dengan kode (a<b)?a:b.

9. /* pembagian integer, casting */

```
Class nug {
public static void main(String[] args) {
// TODO Auto-generated method stub
/* KAMUS */
int x = 1; int y = 2; float fx; float fy;
/* ALGORITMA */
System.out.print ("x/y (format integer) = "+ x/y);
System.out.print ("\nx/y (format float) = "+ (float)x/(float)y);
/* supaya hasilnya tidak nol */
fx=x;
fy=y;
System.out.print ("\nx/y (format integer) = "+ x/y);
System.out.print ("\nx/y (format float) = "+ fx/fy);
/* casting */
System.out.print ("\nfloat(x)/float(y) (format integer) = "+ x/y);
System.out.print ("\nfloat(x)/float(y) (format float) = "+
(float)x/(float)y);
x = 10; y = 3; float a; float b;
a=x;
b=y;
System.out.print ("\nx/y (format integer) = "+ x/y);
System.out.print ("\nx/y (format float) = "+ a/b);
}
}
```

Output:



```
C:\>Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java nug
x/y (format integer) = 0
x/y (format float) = 0.5
x/y (format integer) = 0
x/y (format float) = 0.5
float(x)/float(y) (format integer) = 0
float(x)/float(y) (format float) = 0.5
x/y (format integer) = 3
x/y (format float) = 3.333333
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>
```

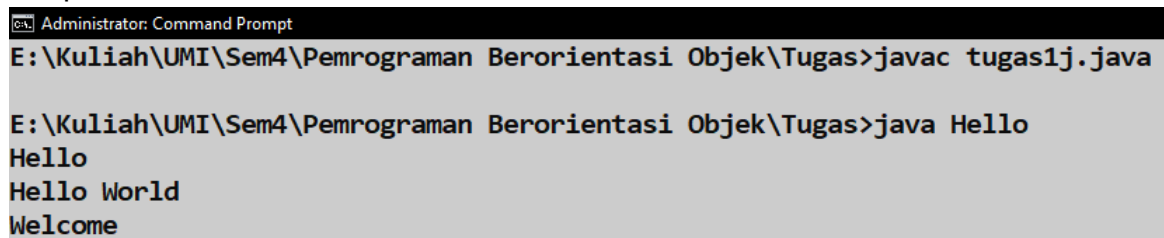
Penjelasan:

Program di atas tentang pembagian yang dimana jika ekspresi pembagian tipe data int menghasilkan bilangan bulat maka, menggunakan cara sesuai gambar diatas dengan tipe data int tetapi jika hasilnya bukan bilangan bulat maka di tulis dengan menggunakan tipe data float.

10.

```
class Hello {
    public static void main(String[] args) {
        System.out.print("Hello");
        /* menuliskan hello dan ganti baris*/
        System.out.print("\nHello ");
        /* menuliskan hello dan ganti baris*/
        System.out.println("World");
        System.out.println("Welcome");
    }
}
```

Output:



```
Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1j.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Hello
Hello
Hello World
Welcome
```

Penjelasan:

Program di atas tentang cara menampilkan teks kemudian membuat baris baru dengan perintah “\n”.

11.

```
/* Efek dari operator ++ */
class wanas {
    /**
     * @param args
     */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* Kamus */
        int i, j;
        /* Program */
        i = 3;
        j = i++;
        System.out.println ("Nilai i : " + (++i) +
            "\nNilai j : " + j);
    }
}
```

Output:

```
Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1k.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java wanas
Nilai i : 5
Nilai j : 3
```

Penjelasan:

Program di atas tentang cara menggunakan increment (++), fungsi dari increment yaitu menambah nilai variable sebanyak satu angka.

12. `/* pemakaian beberapa operator terhadap bit */`
`class Oper1 {`
`/**`
 `* @param args`
 `*/`
`public static void main(String[] args) {`
 `// TODO Auto-generated method stub`
 `/* KAMUS */`
`int n = 10; /* 1010 */`
`int x = 1; /* 1 */`
`int y = 2; /* 10 */ /* ALGORITMA */`
`System.out.println ("n = "+ n);`
`System.out.println ("x = "+ x);`
`System.out.println ("y = "+ y);`
`System.out.println ("n & 8 = "+ (n & 8)); /* 1010 AND 1000 */`
`System.out.println ("x & ~ 8 = "+ (x & ~8)); /* 1 AND`
`0111 */`
`System.out.println ("y << 2 = "+ (y << 2)); /* 10 ==>`
`1000 = 8 */`
`System.out.println ("y >> 3 = "+ (y >>3)); /* 10 ==>`
`0000 = 0 */`
`}`
`}`

Output:

```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas11.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Oper1
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0

```

Penjelasan:

Program di atas tentang cara pemakaian beberapa operator terhadap bit atau dikenal dengan operator bitwise, program di atas menggunakan beberapa operator bitwise di antaranya AND(&), Negasi/Kebalikan(~), Left Shift(<>) operator ini berlaku untuk tipe data int, long, short, char dan byte.

13. /* pemakaian beberapa operator terhadap RELATIONAL DAN bit */

```

class Oper2 {
/**
 * @param args
 */
public static void main(String[] args) {
// TODO Auto-generated method stub
/* KAMUS */
char i, j;
/* ALGORITMA */
i = 3; /* 00000011 dalam biner */
j = 4; /* 00000100 dalam biner */

System.out.println("i = "+ (int) i);
System.out.println("j = "+ j);
System.out.println("i & j = "+ (i & j)); /* 0: 00000000 dalam
biner */
System.out.println("i | j = "+ (i | j)); /* 7:
00000111 biner */
System.out.println("i ^ j = "+ (i ^ j)); /* 7:
00000111 biner Ingat!!! operator "^" pada bahasa java bukan
sebagai pangkat*/
System.out.println(Math.pow(i, j)); /* Class Math
memiliki method pow(a,b) untuk pemangkatan*/
System.out.println(" ~i = "+ ~i); /* -4: 11111100

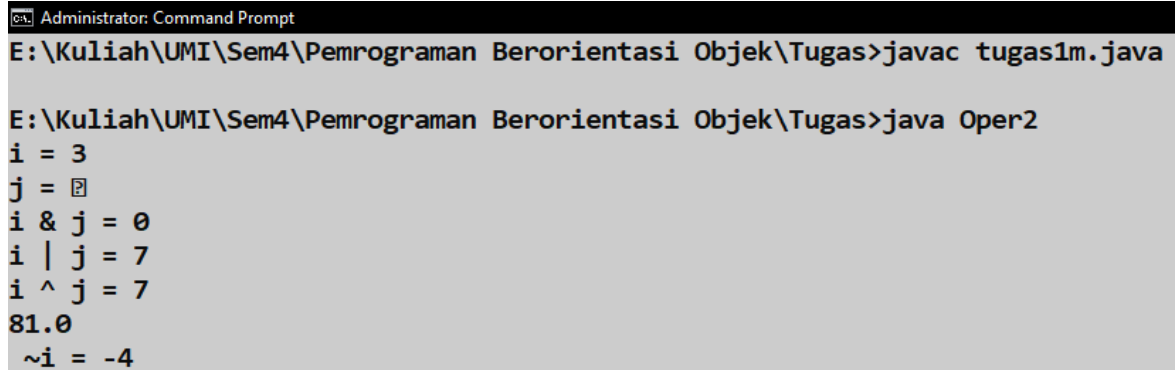
```

```

biner */
}
}

```

Output:



```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1m.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Oper2
i = 3
j = 4
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4

```

Penjelasan:

program di atas tentang pemakaian operator terhadap relational dan bit, pada int j=4 itu tidak akan tampil hasilnya di layar karena tipe data j ialah char dan value nya int jadi tidak ada relasi. Pada program di atas ada tambahan operator yaitu OR(|) dan XOR(^). Pada bahas java tanda ^ bukan berarti pemangkatan, untuk pemangkatan menggunakan math.pow(variable).

```

14. class Oper3 {
    /**
     * @param args
     */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* Algoritma */
        if (true && true){ System.out.println(true && true); }
        /* true = true and true */
        if (true & true) { System.out.println(true & false); } /*
        true & true */
        if (true) { System.out.println(true); } /* true
        */
        if (true || true){ System.out.println(true); } /* true
        = true or true */
        if (true|false) { System.out.println(true|false); } /*
        true|false */
    }
}

```

Output:

```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1n.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Oper3
true
false
true
true
true

```

Penjelasan:

Program di atas tentang pemakaian Operator logika.

15. `/* Operator terner */`
`class Oper4 {`
`/**`
 `* @param args`
 `*/`
`public static void main(String[] args) {`
 `// TODO Auto-generated method stub`
`/* KAMUS */`
`int i = 0; /* perhatikan int i,j=0 bukan seperti ini */`
`int j = 0;`
`char c = 8; char d = 10;`
`int e = (((int)c > (int)d) ? c: d);`
`int k = ((i>j) ? i: j);`
`/* ALGORITMA */`
`System.out.print ("Nilai e = "+ e);`
`System.out.print ("\nNilai k = "+ k);`
`i = 2;`
`j = 3;`
`k = ((i++>j++) ? i: j) ;`
`System.out.print ("\nNilai k = "+ k);`
`}`
`}`

Output:

```

Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1o.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Oper4
Nilai e = 10
Nilai k = 0
Nilai k = 4

```

Penjelasan:

Program di atas tentang cara pemakaian operator ternary dan mengkombinasikan dengan increment.

```
16. class Oprator {
    /**
     * @param args
     */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* Kamus */
        boolean Bool1, Bool2, TF ;
        int i, j, hsl ;
        float x, y, res;
        /* algoritma */
        System.out.println ("Silahkan baca teksnya dan tambahkan perintah
        untuk menampilkan output");
        Bool1 = true;
        Bool2 = false;
        TF = Bool1 && Bool2 ; /* Boolean AND */
        System.out.println("Bool1 && Bool2 = " + (Bool1 && Bool2));
        TF = Bool1 || Bool2 ; /* Boolean OR */
        System.out.println("Bool1 || Bool2 = " + (Bool1 | Bool2));
        TF = ! Bool1 ; /* NOT */
        System.out.println("!Bool1 = " + !Bool1);
        TF = Bool1 ^ Bool2; /* XOR */
        System.out.println("Bool1 ^ Bool2 = " + (Bool1 ^ Bool2));
        /* operasi numerik */
        i = 5; j = 2 ;
        hsl = i + j;
        System.out.println("i + j = " + (i + j));
        hsl = i - j;
        System.out.println("i - j = " + (i - j));
        hsl = i / j;
        System.out.println("i / j = " + (i / j));
        hsl = i * j;
        System.out.println("i * j = " + (i * j));
        hsl = i / j; /* pembagian bulat */
        System.out.println("i / j = " + (i / j));
        hsl = i % j; /* sisa. modulo */
        System.out.println("i % j = " + (i % j));
```

```

/* operasi numerik */
x = 5 ;
y = 5 ;
res = x + y;
System.out.println("x + y = " + (x + y));
res = x - y;
System.out.println("x - y = " + (x - y));
res = x / y;
System.out.println("x / y = " + (x / y));
res = x * y;
System.out.println("x * y = " + (x * y));
/* operasi relasional numerik */
TF = (i == j);
System.out.println("i == j = " + (i == j));
TF = (i != j);
System.out.println("i != j = " + (i != j));TF = (i < j);
System.out.println("i < j = " + (i < j));
TF = (i > j);
System.out.println("i > j = " + (i > j));
TF = (i <= j);
System.out.println("i <= j = " + (i <= j));
TF = (i >= j);
System.out.println("i >= j = " + (i >= j));
/* operasi relasional numerik */
TF = (x != y);
System.out.println("x != y = " + (x != y));
TF = (x < y);
System.out.println("x < y = " + (x < y));
TF = (x > y);
System.out.println("x > y = " + (x > y));
TF = (x <= y);
System.out.println("x <= y = " + (x <= y));
TF = (x >= y);
System.out.println("x >= y = " + (x >= y));
}
}

```

Output:

```
Administrator: Command Prompt
E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>javac tugas1p.java

E:\Kuliah\UMI\Sem4\Pemrograman Berorientasi Objek\Tugas>java Oprator
Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output
Bool1 && Bool2 = false
Bool1 || Bool2 = true
!Bool1 = false
Bool1 ^ Bool2 = true
i + j = 7
i - j = 3
i / j = 2
i * j = 10
i / j = 2
i % j = 1
x + y = 10.0
x - y = 0.0
x / y = 1.0
x * y = 25.0
i == j = false
i != j = true
i < j = false
i > j = true
i <= j = false
i >= j = true
x != y = false
x < y = false
x > y = false
x <= y = true
x >= y = true
```

Penjelasan:

Program di atas tentang penggunaan operator dari berbagai tipe data seperti tipe data Boolean yang menggunakan operator logika and, true kebalikan, dan xor. Kemudian penggunaan operator di operasi numeric mulai dari +, -, *, / dan juga operator untuk sisa bagi(%). Kemudian operasi relasional numeric dari dua tipe data yang berbeda