

LAPORAN
PEMROGRAMAN BERBASIS OBJEK



Nama : Achmad Zulfikar

Nim : 13020220007

Kelas : A1

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA
MAKASSAR
2023

1. Kode program 1

```
//Nama = Achmad Zulfikar
//Nim = 13020220007
//Hari/Tanggal = Minggu-25-2024
//Waktu Pengerjaan = 15.00-23.00

public class Kodeprogram1 {

    public static void main(String[] args) {
        /* Kamus */
        float f= 20.0f;
        double f11;

        /* Algoritma */
        f11=10.0f;
        System.out.println ("f : "+f+ "\nf11: "+f11);
    }
}
```

Output :

```
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram1.java
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram1
f : 20.0
f11: 10.0
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram1`: Mendefinisikan kelas Java dengan nama `Kodeprogram1`. Kata kunci `"public"` menunjukkan bahwa kelas ini dapat diakses dari kelas lain. Nama kelas harus sama dengan nama file Java yang menyimpannya.
2. `public static void main(String[] args)`: Metode utama (main method) dari program. Ini adalah titik masuk untuk eksekusi program Java. Keyword `"public"` menunjukkan bahwa metode ini dapat diakses dari luar kelas. `"static"` berarti metode ini bisa dipanggil tanpa harus membuat objek dari kelasnya. `"void"` menunjukkan bahwa metode ini tidak mengembalikan nilai. `"String[] args"` adalah parameter yang dapat diterima oleh metode `main`, biasanya digunakan untuk menerima argumen baris perintah.

3. `float f= 20.0f;` Mendeklarasikan variabel `f` dengan tipe data float dan memberinya nilai awal 20.0. Huruf "f" diakhir nilai (20.0f) menandakan bahwa ini adalah konstanta float.
4. `double fl;` Mendeklarasikan variabel `fl` dengan tipe data double tanpa memberikan nilai awal. Variabel ini akan diisi nilainya kemudian dalam program.
5. `fl=10.0f;` Mengisi nilai variabel `fl` dengan 10.0. Karena `fl` memiliki tipe data double, maka nilai float 10.0 akan di-convert ke double.
6. `System.out.println ("f : "+f+ "\nfl: "+fl);` Mencetak nilai dari variabel `f` dan `fl` ke konsol. `System.out.println` adalah perintah untuk mencetak output ke konsol. Tanda "+" digunakan untuk menggabungkan (concatenate) string dengan nilai variabel. `"\n"` digunakan untuk membuat baris baru dalam output.

2. Kode program 2

```
//Nama = Achmad Zulfikar
//Nim = 13020220007
//Hari/Tanggal = Minggu-25-2024
//Waktu Pengerjaan = 15.00-23.00

public class Kodeprogram2 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* Kamus */
        int i;
        /* Program */
        System.out.print ("hello\n"); i = 5;
        System.out.println ("Ini nilai i : " + i);
    }
}
```

Output :

```
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram2.java
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram2
hello
Ini nilai i : 5
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram2`: Mendefinisikan kelas Java dengan nama `Kodeprogram2`. Ini mirip dengan program sebelumnya, hanya beda nama kelasnya.
2. `public static void main(String[] args)`: Metode utama (main method) dari program. Sama seperti yang dijelaskan sebelumnya, ini adalah titik masuk untuk eksekusi program Java.
3. `int i`; Mendeklarasikan variabel `i` dengan tipe data `int` tanpa memberikan nilai awal. Variabel ini akan diisi nilainya kemudian dalam program.
4. `System.out.print("hello\n"); i = 5`; Baris ini mencetak "hello" ke konsol tanpa membuat baris baru (karena menggunakan `print` bukan `println`), kemudian memberikan nilai 5 pada variabel `i`. Tanda `\n` digunakan untuk membuat baris baru setelah mencetak "hello".
5. `System.out.println("Ini nilai i : " + i)`; Baris ini mencetak nilai dari variabel `i` ke konsol dengan pesan "Ini nilai i : " diikuti oleh nilai variabel `i`. Seperti sebelumnya, tanda `+` digunakan untuk menggabungkan string dengan nilai variabel.

3. Kode program 3

```
//Nama = Achmad Zulfikar
//Nim = 13020220007
//Hari/Tanggal = Minggu-25-2024
//Waktu Pengerjaan = 15.00-23.00

public class Kodeprogram3 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* KAMUS */
        short ks = 1;
        int ki = 1;
        long kl = 10000;
        char c = 65; /* inisialisasi karakter dengan integer */
        char c1 = 'Z'; /* inisialisasi karakter dengan karakter */
        double x = 50.2f;
        float y = 50.2f;
        /* Algoritma */

        /* penulisan karakter sebagai karakter */
        System.out.println("Karakter = " + c);
        System.out.println("Karakter = " + c1);

        /* penulisan karakter sebagai integer */
        System.out.println("Karakter = " + c);
        System.out.println("Karakter = " + c1);

        System.out.println("Bilangan integer (short) = " + ks);
        System.out.println("\t(int) = " + ki);
        System.out.println("\t(long)= " + kl);
        System.out.println("Bilangan Real x = " + x);
        System.out.println("Bilangan Real y = " + y);
    }
}
```

Output :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram3.java
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram3
Karakter = A
Karakter = Z
Karakter = A
Karakter = Z
Bilangan integer (short) = 1
      (int) = 1
      (long)= 10000
Bilangan Real x = 50.200000076293945
Bilangan Real y = 50.2
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram3`: Mendefinisikan kelas Java dengan nama `Kodeprogram3`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `short ks = 1; int ki = 1; long kl = 10000`: Mendeklarasikan variabel dengan tipe data `short`, `int`, dan `long`, masing-masing diberi nilai awal.
4. `char c = 65; char c1 = 'Z'`: Mendeklarasikan variabel dengan tipe data `char`, `c` diinisialisasi dengan nilai integer 65, yang merupakan representasi dari karakter 'A' dalam ASCII. `c1` diinisialisasi langsung dengan karakter 'Z'.
5. `double x = 50.2f; float y = 50.2f`: Mendeklarasikan variabel dengan tipe data `double` dan `float`, di mana kedua variabel ini diinisialisasi dengan nilai 50.2 dalam bentuk `float`.
6. `System.out.println("Karakter = " + c)`: Mencetak nilai variabel `c` ke konsol, menampilkan karakter yang direpresentasikan oleh nilai ASCII yang diberikan (dalam hal ini 'A').
7. `System.out.println("Bilangan integer (short) = " + ks)`: Mencetak nilai variabel `ks` ke konsol dengan pesan yang sesuai.

8. `System.out.println("Bilangan Real x = "+ x);`; Mencetak nilai variabel `x` ke konsol dengan pesan yang sesuai.

4. Kode Program 4

```
//Nama = Achmad Zulfikar
//Nim = 13020220007
//Hari/Tanggal = Minggu-25-2024
//Waktu Pengerjaan = 15.00-23.00

import java.util.Scanner;

/* contoh membaca integer menggunakan Class Scanner*/
public class Kodeprogram4 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* Kamus */
        int a;
        Scanner masukan;
        /* Program */
        System.out.print ("Contoh membaca dan menulis, ketik nilai integer: \n");
        masukan = new Scanner(System.in);
        a = masukan.nextInt();
        /* coba ketik : masukan.nextInt();
        Apa akibatnya ?*/
        System.out.print ("Nilai yang dibaca : "+ a);
    }
}
```

Output :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praaktek>javac Kodeprogram4.java
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praaktek>java Kodeprogram4
Contoh membaca dan menulis, ketik nilai integer:
34
Nilai yang dibaca : 34
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praaktek>
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `import java.util.Scanner;`; Mengimpor kelas `Scanner` dari paket `java.util`, yang akan digunakan untuk membaca input dari pengguna.
2. `public class Kodeprogram4`; Mendefinisikan kelas Java dengan nama `Kodeprogram4`.
3. `public static void main(String[] args)`; Metode utama (main method) dari program.
4. `int a`; `Scanner masukan`; Mendeklarasikan variabel `a` dengan tipe data `int` yang akan menyimpan nilai yang dimasukkan oleh

pengguna, dan variabel masukan dengan tipe data Scanner yang akan digunakan untuk membaca input dari pengguna.

5. `System.out.print ("Contoh membaca dan menulis, ketik nilai integer: \n");` Mencetak pesan ke konsol untuk memberi petunjuk kepada pengguna agar memasukkan nilai integer.
6. `masukan = new Scanner(System.in);` Membuat objek baru dari kelas Scanner dengan parameter `System.in`, yang mengarahkan input dari keyboard.
7. `a = masukan.nextInt();` Membaca nilai integer yang dimasukkan oleh pengguna dan menyimpannya ke dalam variabel `a` menggunakan metode `nextInt()` dari objek `masukan`.
8. `System.out.print ("Nilai yang dibaca : "+ a);` Mencetak nilai yang telah dibaca dari input pengguna ke konsol.

5. Kode program 5

```
////NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
//import javax.swing.*;

public class Kodeprogram5 {

    public static void main(String[] args) throws IOException {
        // TODO Auto-generated method stub
        /* Kamus */
        char cc;
        int bil;
        InputStreamReader isr = new
        InputStreamReader(System.in);
        BufferedReader dataIn = new BufferedReader(isr);
        // atau

        BufferedReader dataIn = 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");
        /* Tambahkan program membaca/input data menggunakan Class Scanner, Class Console dan Class
        JOptionPane */
    }
}
```

Output :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram5.java
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram5
hello
baca 1 karakter : hello
baca 1 bilangan : 123
h
123
bye
```

Tujuan Program: Tujuan dari program ini adalah untuk menunjukkan beberapa cara membaca input dari pengguna, baik berupa karakter maupun bilangan, menggunakan kelas `BufferedReader` dan `InputStreamReader`.

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `import java.io.BufferedReader; import java.io.IOException; import java.io.InputStreamReader;`: Mengimpor kelas-kelas yang diperlukan untuk membaca input dari pengguna, seperti `BufferedReader`, `InputStreamReader`, dan `IOException`.
2. `public class Kodeprogram5`: Mendefinisikan kelas Java dengan nama `Kodeprogram5`.
3. `public static void main(String[] args) throws IOException`: Metode utama (main method) dari program, dengan menyertakan deklarasi `throws IOException` untuk menangani exception yang dapat terjadi saat membaca input.
4. `char cc; int bil;`: Mendeklarasikan variabel `cc` dengan tipe data `char` dan `bil` dengan tipe data `int`.
5. `InputStreamReader isr = new InputStreamReader(System.in);`
`BufferedReader dataIn = new BufferedReader(isr);`: Membuat objek `InputStreamReader` yang akan membaca input dari `System.in`, dan kemudian menggunakan objek tersebut untuk membuat objek `BufferedReader` yang akan digunakan untuk membaca input karakter.
6. `cc = dataIn.readLine().charAt(0);`: Membaca input karakter dari pengguna dan menyimpannya ke dalam variabel `cc` setelah

membaca satu baris teks. `.charAt(0)` digunakan untuk mengambil karakter pertama dari baris teks yang telah dibaca.

7. `bil = Integer.parseInt(dataIn.readLine());` Membaca input bilangan dari pengguna dalam bentuk string, kemudian mengonversinya menjadi tipe data integer menggunakan metode `Integer.parseInt()`.
8. `System.out.print (cc + "\n" + bil + "\n");` Mencetak nilai variabel `cc` dan `bil` ke konsol.

6. Kode program 6

```
//Nama = Achmad Zulfikar
//Nim = 13020220007
//Hari/Tanggal = Minggu-25-2024
//Waktu Pengerjaan = 15.00-23.00

/*Casting menggunakan tipe data primitif*/
public class Kodeprogram6 {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int a=5, b=6;
        float d=2.f,e=3.2f;
        char g='5';
        double k=3.14;
        System.out.println((float)a); // int <-- float
        System.out.println((double)b); // int <-- double
        System.out.println((int)d); // float <-- int
        System.out.println((double)e); // float <-- double
        System.out.println((int)g); // char <-- int (ASCII)
        System.out.println((float)g); // char <-- float (ASCII)
        System.out.println((double)g); // char <-- double (ASCII)
        System.out.println((int)k); // double <-- int
        System.out.println((float)k); // double <-- float
    }
}
```

Output :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram6.java

D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram6
5.0
6.0
2
3.2000000047683716
53
53.0
53.0
3
3.14
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram6`: Mendefinisikan kelas Java dengan nama `Kodeprogram6`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.

3. `int a=5, b=6; float d=2.5f, e=3.2f; char g='5'; double k=3.14;`
Mendeklarasikan beberapa variabel dengan tipe data primitif, seperti `int`, `float`, `char`, dan `double`, serta memberikan nilai awal kepada masing-masing variabel.
4. `System.out.println((float)a);`; Menggunakan operasi casting untuk mengubah nilai variabel `a` yang bertipe `int` menjadi `float` dan kemudian mencetak hasilnya ke konsol.
5. `System.out.println((double)b);`; Menggunakan operasi casting untuk mengubah nilai variabel `b` yang bertipe `int` menjadi `double` dan kemudian mencetak hasilnya ke konsol.
6. `System.out.println((int)d);`; Menggunakan operasi casting untuk mengubah nilai variabel `d` yang bertipe `float` menjadi `int` dan kemudian mencetak hasilnya ke konsol.
7. `System.out.println((double)e);`; Menggunakan operasi casting untuk mengubah nilai variabel `e` yang bertipe `float` menjadi `double` dan kemudian mencetak hasilnya ke konsol.
8. `System.out.println((int)g);`; Menggunakan operasi casting untuk mengubah nilai variabel `g` yang bertipe `char` (bernilai '5' dalam ASCII) menjadi `int` dan kemudian mencetak hasilnya ke konsol.
9. `System.out.println((float)g);`; Menggunakan operasi casting untuk mengubah nilai variabel `g` yang bertipe `char` (bernilai '5' dalam ASCII) menjadi `float` dan kemudian mencetak hasilnya ke konsol.
10. `System.out.println((double)g);`; Menggunakan operasi casting untuk mengubah nilai variabel `g` yang bertipe `char` (bernilai '5' dalam ASCII) menjadi `double` dan kemudian mencetak hasilnya ke konsol.
11. `System.out.println((int)k);`; Menggunakan operasi casting untuk mengubah nilai variabel `k` yang bertipe `double` menjadi `int` dan kemudian mencetak hasilnya ke konsol.

12. `System.out.println((float)k);` Menggunakan operasi casting untuk mengubah nilai variabel `k` yang bertipe `double` menjadi `float` dan kemudian mencetak hasilnya ke konsol.

7. Kode program 7

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/*Casting menggunakan tipe data Class*/
public class Kodeprogram7 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int a=8,b=9;
        float d=2.f,e=3.2f;
        char g='5';
        double k=3.14;
        String n="67",m="45", l="100";

        a = Integer.parseInt(n); /*Konversi String ke Integer*/
        k = Double.parseDouble(m); /*Konversi String ke Double*/
        d = Float.parseFloat(l); /*Konversi String ke Float*/
        System.out.println("a : "+a+"\nk : "+k+"\nd : "+d);
        n = String.valueOf(b); /*Konversi Integer ke String*/
        m = String.valueOf(g); /*Konversi Karakter ke String*/
        l = String.valueOf(e); /*Konversi Float ke String*/
        System.out.println("n : "+n+"\nm : "+m+"\nl : "+l);
        k = Double.valueOf(a).intValue();
        /*Konversi Integer ke Double*/
        double c = Integer.valueOf(b).doubleValue();
        System.out.println("k : "+k+"\nc : "+c+"\nl : "+l);
    }
}
```

Output :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram7.java
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram7
a : 67
k : 45.0
d : 100.0
n : 9
m : 5
l : 3.2
k : 67.0
c : 9.0
l : 3.2
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram7`: Mendefinisikan kelas Java dengan nama `Kodeprogram7`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.

3. `int a=8,b=9; float d=2.f,e=3.2f; char g='5'; double k=3.14; String n="67",m="45", l="100";`; Mendeklarasikan beberapa variabel dengan berbagai tipe data, termasuk `int`, `float`, `char`, `double`, dan `String`, serta memberikan nilai awal kepada masing-masing variabel.
4. `a = Integer.parseInt(n);`; Menggunakan metode `parseInt()` dari kelas `Integer` untuk mengonversi nilai dari variabel `n` yang bertipe `String` menjadi `int`, dan kemudian menyimpannya ke dalam variabel `a`.
5. `k = Double.parseDouble(m);`; Menggunakan metode `parseDouble()` dari kelas `Double` untuk mengonversi nilai dari variabel `m` yang bertipe `String` menjadi `double`, dan kemudian menyimpannya ke dalam variabel `k`.
6. `d = Float.parseFloat(l);`; Menggunakan metode `parseFloat()` dari kelas `Float` untuk mengonversi nilai dari variabel `l` yang bertipe `String` menjadi `float`, dan kemudian menyimpannya ke dalam variabel `d`.
7. `n = String.valueOf(b);`; Menggunakan metode `valueOf()` dari kelas `String` untuk mengonversi nilai dari variabel `b` yang bertipe `int` menjadi `String`, dan kemudian menyimpannya ke dalam variabel `n`.
8. `m = String.valueOf(g);`; Menggunakan metode `valueOf()` dari kelas `String` untuk mengonversi nilai dari variabel `g` yang bertipe `char` menjadi `String`, dan kemudian menyimpannya ke dalam variabel `m`.
9. `l = String.valueOf(e);`; Menggunakan metode `valueOf()` dari kelas `String` untuk mengonversi nilai dari variabel `e` yang bertipe `float` menjadi `String`, dan kemudian menyimpannya ke dalam variabel `l`.
10. `k = Double.valueOf(a).intValue();`; Menggunakan metode `valueOf()` dari kelas `Double` untuk mengonversi nilai dari variabel `a` yang bertipe `int` menjadi `Double`, kemudian menggunakan metode `intValue()` dari kelas `Double` untuk mengambil nilai integer dari objek `Double` tersebut, dan kemudian menyimpannya ke dalam variabel `k`.

11. `double c = Integer.valueOf(b).doubleValue();` Menggunakan metode `valueOf()` dari kelas `Integer` untuk mengonversi nilai dari variabel `b` yang bertipe `int` menjadi `Integer`, kemudian menggunakan metode `doubleValue()` dari kelas `Integer` untuk mengambil nilai `double` dari objek `Integer` tersebut, dan kemudian menyimpannya ke dalam variabel `c`.

8. Kode program 8

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/* pemakaian operator kondisional */
public class Kodeprogram8 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* KAMUS */
        int x = 1;
        int y = 2;
        /* ALGORITMA */
        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));

        /*Gunakan dalam kurung "(statemen dan kondisi)" untuk
        menyatakan satu kesatuan pernyataan*/

    }
}
```

Output :

```
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram8.java
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram8
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram8`: Mendefinisikan kelas Java dengan nama `Kodeprogram8`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `int x = 1; int y = 2;`: Mendeklarasikan dua variabel `x` dan `y` dengan tipe data `int` dan memberikan nilai awal.

4. `System.out.print("x = " + x + "\n"); System.out.print("y = " + y + "\n");`;
Mencetak nilai variabel x dan y ke konsol.
5. `System.out.print("hasil ekspresi = (x<y)?x:y = " + ((x < y) ? x : y));`;
Menggunakan operator kondisional `(x<y)?x:y` untuk memilih nilai yang akan dicetak berdasarkan kondisi `x<y`. Jika kondisi benar `(x<y)`, maka nilai x akan dipilih, jika tidak, nilai y yang akan dipilih. Hasil ekspresi tersebut kemudian dicetak ke konsol.

9. Kode program 9

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/* pembagian integer, casting */
public class Kodeprogram9 {

    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) = " + x/y);
        /* supaya hasilnya tidak nol */
        fx=x;
        fy=y;
        System.out.print ("\nx/y (format integer) = " + fx/fy);
        System.out.print ("\nx/y (format float) = " + fx/fy);
        /* casting */
        System.out.print("\nfloat(x)/float(y) (format integer)= " + (float)x/(float)y);
        System.out.print("\nfloat(x)/float(y) (format float) = " + (float)x/(float)y);
        x = 10; y = 3;
        System.out.print (" \nx/y (format integer) = " + x/y);
        System.out.print (" \nx/y (format float = " + x/y);
    }
}
```

Output :

```
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram8.java

D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram8
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram9.java

D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram9
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
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram9`: Mendefinisikan kelas Java dengan nama `Kodeprogram9`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `int x = 1; int y = 2; float fx; float fy;`: Mendeklarasikan beberapa variabel, yaitu `x` dan `y` dengan tipe data `int`, serta `fx` dan `fy` dengan tipe data `float`.
4. `System.out.print ("x/y (format integer) = "+ x/y);`: Mencetak hasil pembagian `x` dan `y` dalam format integer ke konsol. Karena keduanya bertipe `int`, maka hasilnya juga akan bertipe `int` dan pembulatannya dilakukan secara otomatis.
5. `System.out.print ("\nx/y (format float) = "+ x/y);`: Mencetak hasil pembagian `x` dan `y` dalam format float ke konsol. Meskipun hasilnya akan bertipe `int`, karena pemanggilan langsung tanpa operasi casting, maka pembulatan hasil akan tetap dilakukan secara otomatis.
6. `fx=x; fy=y;`: Menggunakan operasi assignment untuk menyalin nilai variabel `x` dan `y` ke dalam variabel `fx` dan `fy`.
7. `System.out.print ("\nx/y (format integer) = "+ fx/fy);`: Mencetak hasil pembagian `fx` dan `fy` dalam format integer ke konsol. Karena keduanya bertipe `float`, maka hasilnya akan bertipe `float` dan tidak akan ada pembulatan.
8. `System.out.print ("\nx/y (format float) = "+ fx/fy);`: Mencetak hasil pembagian `fx` dan `fy` dalam format float ke konsol.
9. `System.out.print("\nfloat(x)/float(y) (format integer)= "+ (float)x/(float)y);`: Menggunakan operasi casting untuk mengubah tipe data variabel `x` dan `y` menjadi `float`, sehingga hasil pembagiannya akan bertipe `float`.

10. `System.out.print("\nfloat(x)/float(y) (format float) = "+ (float)x/(float)y);` Mencetak hasil pembagian x dan y yang sudah di-casting menjadi float ke konsol.
12. `x = 10; y = 3;` Mengubah nilai variabel x dan y.
13. `System.out.print ("\nx/y (format integer) = "+ x/y);` Mencetak hasil pembagian x dan y setelah nilai keduanya diubah ke konsol.
14. `System.out.print ("\nx/y (format float) = "+ x/y);` Mencetak hasil pembagian x dan y dalam format float ke konsol. Meskipun nilai keduanya sudah diubah, karena pemanggilan langsung tanpa operasi casting, maka pembulatan hasil akan tetap dilakukan secara otomatis.

10. Kode Program 10

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

public class Kodeprogram10 {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        /* menuliskan hello ke layar */
        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 :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram10.java
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram10
Hello
Hello World
Welcome
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram10`: Mendefinisikan kelas Java dengan nama `Kodeprogram10`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `System.out.print("Hello");`: Menggunakan fungsi `print` dari kelas `System.out` untuk menampilkan teks "Hello" ke layar tanpa ganti baris.
4. `System.out.print("\nHello ");`: Menggunakan fungsi `print` untuk menampilkan teks "Hello" ke layar dengan ganti baris sebelumnya.
5. `System.out.println("World");`: Menggunakan fungsi `println` dari kelas `System.out` untuk menampilkan teks "World" ke layar dengan ganti baris setelahnya.
6. `System.out.println("Welcome");`: Menggunakan fungsi `println` untuk menampilkan teks "Welcome" ke layar dengan ganti baris setelahnya.

11. Kode program 11

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/* Efek dari operator ++ */
public class Kodeprogram11 {

    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 :

```
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram11.java
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram11
Nilai i : 5
Nilai j : 3
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram11`: Mendefinisikan kelas Java dengan nama `Kodeprogram11`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `int i, j`; Mendeklarasikan dua variabel `i` dan `j` dengan tipe data `int`.
4. `i = 3`; Memberikan nilai 3 kepada variabel `i`.
5. `j = i++`; Menggunakan operator postfix increment (`i++`), yang berarti nilai `i` akan diberikan ke variabel `j`, kemudian nilai `i` akan bertambah 1 setelah ekspresi dievaluasi.
6. `System.out.println ("Nilai i : " + (++i) + "\nNilai j : " + j)`; Mencetak nilai variabel `i` yang sudah diincrement terlebih dahulu (prefix increment, `++i`), dan kemudian mencetak nilai variabel `j`.

12. Kode program 12

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/* pemakaian beberapa operator terhadap bit */
public class Kodeprogram12 {

    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 :

```
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram12.java  
D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram12  
n = 10  
x = 1  
y = 2  
n & 8 = 8  
x & ~ 8 = 1  
y << 2 = 8  
y >> 3 = 0
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram12`: Mendefinisikan kelas Java dengan nama `Kodeprogram12`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `int n = 10`;; Mendeklarasikan variabel `n` dengan tipe data `int` dan memberikan nilai awal 10, yang dalam representasi biner adalah 1010.
4. `int x = 1`;; Mendeklarasikan variabel `x` dengan tipe data `int` dan memberikan nilai awal 1, yang dalam representasi biner adalah 1.
5. `int y = 2`;; Mendeklarasikan variabel `y` dengan tipe data `int` dan memberikan nilai awal 2, yang dalam representasi biner adalah 10.
6. `System.out.println ("n & 8 = " + (n & 8))`;; Menggunakan operator bitwise AND (&) untuk melakukan operasi bit-wise AND antara nilai variabel `n` dan 8 (1000 dalam biner). Hasilnya adalah 8 karena hanya bit kedua dari kiri yang menyala pada kedua operand.
7. `System.out.println ("x & ~8 = " + (x & ~8))`;; Menggunakan operator bitwise NOT (~) untuk mengubah bit dari 8 (1000 dalam biner) menjadi 0111, kemudian melakukan operasi bit-wise AND antara nilai variabel `x` dan hasil negasi tersebut. Hasilnya adalah 0 karena bit yang menyala pada 8 telah dimatikan oleh operasi bitwise NOT.
8. `System.out.println ("y << 2 = " + (y << 2))`;; Menggunakan operator bitwise shift left (<<) untuk menggeser bit dari nilai variabel `y` ke

kiri sebanyak 2 kali. Hasilnya adalah 8 karena setiap geseran ke kiri menggandakan nilai.

9. `System.out.println("y >> 3 = " + (y >>3));` Menggunakan operator bitwise shift right (`>>`) untuk menggeser bit dari nilai variabel `y` ke kanan sebanyak 3 kali. Hasilnya adalah 0 karena setiap geseran ke kanan akan membuang digit paling kanan.

13. Kode program 13

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/* pemakaian beberapa operator terhadap RELATIONAL DAN bit */
public class Kodeprogram13 {

    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 :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram13.java
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram13
i = 3
j = 4
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram13`: Mendefinisikan kelas Java dengan nama `Kodeprogram13`.

2. `public static void main(String[] args):` Metode utama (main method) dari program.
3. `char i, j;` Mendeklarasikan dua variabel `i` dan `j` dengan tipe data `char`.
4. `i = 3;` Memberikan nilai 3 kepada variabel `i`, yang dalam representasi biner adalah 00000011.
5. `j = 4;` Memberikan nilai 4 kepada variabel `j`, yang dalam representasi biner adalah 00000100.
6. `System.out.println("i & j = " + (i & j));` Menggunakan operator bitwise AND (&) untuk melakukan operasi bit-wise AND antara nilai variabel `i` dan `j`. Hasilnya adalah 0 karena tidak ada bit yang sama antara `i` dan `j`.
7. `System.out.println("i | j = " + (i | j));` Menggunakan operator bitwise OR (|) untuk melakukan operasi bit-wise OR antara nilai variabel `i` dan `j`. Hasilnya adalah 7 karena ada tiga bit yang menyala pada `i` dan `j`.
8. `System.out.println("i ^ j = " + (i ^ j));` Menggunakan operator bitwise XOR (^) untuk melakukan operasi bit-wise XOR antara nilai variabel `i` dan `j`. Hasilnya adalah 7 karena ada tiga bit yang berbeda antara `i` dan `j`.
9. `System.out.println(Math.pow(i, j));` Menggunakan fungsi `pow()` dari kelas `Math` untuk melakukan pemangkatan antara nilai variabel `i` dan `j`. Namun, perlu dicatat bahwa ini bukan operasi bitwise, tetapi operasi pemangkatan.
10. `System.out.println("~i = " + ~i);` Menggunakan operator bitwise NOT (~) untuk mengubah semua bit dalam nilai variabel `i` menjadi kebalikannya. Hasilnya adalah -4 karena dalam representasi signed integer, bit paling kiri adalah bit tanda, sehingga nilai tersebut dianggap sebagai bilangan negatif.

14. Kode program 14

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

public class Kodeprogram14 {

    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 :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram14.java

D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram14
true
false
true
true
true
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram14`: Mendefinisikan kelas Java dengan nama `Kodeprogram14`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `if (true && true) { System.out.println(true && true);}`: Menggunakan operator logika AND (`&&`) untuk menguji apakah keduanya bernilai `true`. Hasilnya adalah `true` karena keduanya `true`.
4. `if (true & true) { System.out.println(true & false);}`: Menggunakan operator bitwise AND (`&`) untuk menguji apakah keduanya bernilai `true`. Hasilnya adalah `true` karena keduanya `true`.
5. `if (true) { System.out.println(true); }`: Menggunakan kondisi `if` dengan nilai `true`. Karena nilai kondisi adalah `true`, pernyataan dalam blok `if` akan dieksekusi.

6. `if (true || true) { System.out.println(true); }`: Menggunakan operator logika OR (`||`) untuk menguji apakah salah satu atau keduanya bernilai `true`. Hasilnya adalah `true` karena keduanya `true`.
7. `if (true | false) { System.out.println(true | false); }`: Menggunakan operator bitwise OR (`|`) untuk menguji apakah salah satu atau keduanya bernilai `true`. Hasilnya adalah `true` karena keduanya `true`.

15. Kode program 15

```
//NIM = 13020220007
//Nama = Achmad Zulfikar
//Hari/Tanggal = Minggu 25-02-2024
//Waktu Pengerjaan = 15.00 - 23.00

/* Operator ternary */
public class Kodeprogram15 {

    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 :

```
D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram15.java

D:\Tugas PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram15
Nilai e = 10
Nilai k = 0
Nilai k = 4
```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram15`: Mendefinisikan kelas Java dengan nama `Kodeprogram15`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.

3. `int i = 0; int j = 0;` Mendeklarasikan dua variabel `i` dan `j` dengan tipe data `int` dan memberikan nilai awal 0.
4. `char c = 8; char d = 10;` Mendeklarasikan dua variabel `c` dan `d` dengan tipe data `char` dan memberikan nilai awal 8 dan 10.
5. `int e = (((int)c > (int)d) ? c : d);` Menggunakan operator ternary untuk mengevaluasi apakah nilai variabel `c` lebih besar dari `d`. Jika benar, nilai variabel `e` akan sama dengan `c`, jika tidak, nilai variabel `e` akan sama dengan `d`.
6. `int k = ((i > j) ? i : j);` Menggunakan operator ternary untuk mengevaluasi apakah nilai variabel `i` lebih besar dari `j`. Jika benar, nilai variabel `k` akan sama dengan `i`, jika tidak, nilai variabel `k` akan sama dengan `j`.
7. `System.out.print ("Nilai e = " + e);` Mencetak nilai variabel `e`.
8. `System.out.print ("\nNilai k = " + k);` Mencetak nilai variabel `k`.
9. `k = ((i++ > j++) ? i : j);` Menggunakan operator ternary untuk mengevaluasi apakah nilai variabel `i` setelah di-increment lebih besar dari nilai variabel `j` setelah di-increment. Jika benar, nilai variabel `k` akan sama dengan nilai variabel `i` setelah di-increment, jika tidak, nilai variabel `k` akan sama dengan nilai variabel `j` setelah di-increment.
10. `System.out.print ("\nNilai k = " + k);` Mencetak nilai variabel `k` setelah perubahan nilai.

16. Kode program 16


```

/* Contoh pengoperasian variabel bertipe dasar */
public class Kodeprogram16 {

    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 program di bawah ini untuk menampilkan output program");

        Bool1 = true; Bool2 = false;
        TF = Bool1 && Bool2 ; /* Boolean AND */
        TF = Bool1 || Bool2 ; /* Boolean OR */
        TF = ! Bool1 ; /* NOT */
        TF = Bool1 ^Bool2; /* XOR */
        /* operasi numerik */
        i = 5; j = 2 ;
        hsl = i * j;
        hsl = i - j;
        hsl = i / j;
        hsl = i * j;
        hsl = i /j ; /* pembagian bulat */
        hsl = i%j ; /* sisa modulo */

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

```

Output :

```

D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>javac Kodeprogram16.java

D:\Tugas PB0\Tugas 1\Tugas1_SourceCode_Praktek>java Kodeprogram16
Silahkan baca teksnya dan tambahkan perintah program di bawah ini untuk menampilkan output program

```

Penjelasan Keyword dan Konstruksi yang Digunakan:

1. `public class Kodeprogram16`: Mendefinisikan kelas Java dengan nama `Kodeprogram16`.
2. `public static void main(String[] args)`: Metode utama (main method) dari program.
3. `boolean Bool1, Bool2, TF`;; Mendeklarasikan tiga variabel boolean `Bool1`, `Bool2`, dan `TF`.
4. `int i, j, hsl`;; Mendeklarasikan dua variabel integer `i` dan `j`, serta satu variabel `hsl` untuk menampung hasil operasi.
5. `float x, y, res`;; Mendeklarasikan dua variabel float `x` dan `y`, serta satu variabel `res` untuk menampung hasil operasi.
6. `System.out.println("Silahkan baca teksnya dan tambahkan perintah program di bawah ini untuk menampilkan output program");`;; Memberikan petunjuk kepada pengguna untuk menambahkan

perintah program di bawahnya untuk menampilkan output program.

7. Operasi logika:

- $TF = Bool1 \ \&\& \ Bool2$; Menggunakan operator logika AND untuk mengevaluasi apakah kedua boolean Bool1 dan Bool2 bernilai true. Hasilnya disimpan dalam variabel TF.
- $TF = Bool1 \ || \ Bool2$; Menggunakan operator logika OR untuk mengevaluasi apakah salah satu atau kedua boolean Bool1 dan Bool2 bernilai true. Hasilnya disimpan dalam variabel TF.
- $TF = !Bool1$; Menggunakan operator logika NOT untuk mengubah nilai boolean Bool1 menjadi kebalikannya. Hasilnya disimpan dalam variabel TF.
- $TF = Bool1 \ \wedge \ Bool2$; Menggunakan operator XOR (exclusive OR) untuk mengevaluasi apakah salah satu dari dua boolean Bool1 dan Bool2 bernilai true, tetapi tidak keduanya. Hasilnya disimpan dalam variabel TF.

8. Operasi aritmatika pada integer:

- $hsl = i + j$, $hsl = i - j$, $hsl = i * j$, $hsl = i / j$; Melakukan operasi penjumlahan, pengurangan, perkalian, dan pembagian pada variabel integer i dan j. Hasilnya disimpan dalam variabel hsl.
- $hsl = i / j$; Melakukan pembagian bulat antara variabel i dan j.
- $hsl = i \% j$; Menghitung sisa pembagian (modulus) dari i dibagi j.

9. Operasi aritmatika pada float:

- $res = x + y$, $res = x - y$, $res = x * y$, $res = x / y$; Melakukan operasi penjumlahan, pengurangan, perkalian, dan

pembagian pada variabel float x dan y. Hasilnya disimpan dalam variabel res.

10. Operasi relasional:

- $(i == j)$, $(i != j)$, $(i < j)$, $(i > j)$, $(i <= j)$, $(i >= j)$: Menggunakan operator relasional untuk mengevaluasi hubungan antara dua variabel integer i dan j, seperti kesetaraan, ketidaksetaraan, kurang dari, lebih dari, kurang dari atau sama dengan, dan lebih dari atau sama dengan.
- $(x != y)$, $(x < y)$, $(x > y)$, $(x <= y)$, $(x >= y)$: Menggunakan operator relasional untuk mengevaluasi hubungan antara dua variabel float x dan y.