

## LAPORAN TUGAS



### PEMROGRAMAN BERORIENTASI OBJEK

NAMA : MUTHI'AH ARIQAH MUTMAINNAH SYAM

KELAS : B2

NIM : 13020200203

## Tugas 2 PBO

### 1. Tugas Praktek : Praktek Program Java : Variabel dan tipe Data

#### 1). Program 1

##### Output :

```
run:
f : 20.0
f11: 10.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

##### Penjelasan :

Program diatas bertujuan untuk menampilkan nilai dari variable f dan f11.

- **public class Asdgll** → dibuatkan sebuah class dengan nama Asdgll yang memiliki access public.
- **public static void main(String[] args)** → merupakan method utama.
- **float f = 20.f;** → dideklarasikan sebuah variable f dgn tipe data float yang diberi nilai 20.0, setelah nilai perlu ditambahkan notasi 'f' atau 'F'.
- **double f11;** → dibuatkan variable dgn nama f11 bertipe data double.
- **f11=10.0f;** → sebuah statement assignment yang dimana variable f11 diinisialisasi nilainya menjadi 10.0
- **System.out.println ("f : "+f + "\nf11: "+f11);** → merupakan sebuah perintah dalam java untuk menampilkan/mencetak isi dari variable yang telah di deklarasikan.

## 2). Program 2

### Output :

```
run:
hello
Ini nilai i :5
BUILD SUCCESSFUL (total time: 0 seconds)
```

### Penjelasan :

Program diatas menampilkan kata “ halo” dan nilai dari variable i yaitu 5.

- **public class Asign** → dibuatkan sebuah class dengan nama Asign yang memiliki access public.
- **public static void main(String[] args)** → merupakan method utama.
- **int i;** → diberikan sebuah variable bertipe data int dengan nama i.
- **System.out.print ("hello\n"); i = 5;** → merupakan sebuah perintah untuk menampilkan isi string dan nilai variable i diisikan nilainya yaitu 5.
- **System.out.println ("Ini nilai i : " + i);** → akan mencetak nilai dari variable i.

### 3). Program 3

#### Output :

```
run:
Karakter = A
Karakter = Z
Karakter = A
Karakter = Z
Bilangan integer (short) = 1
      (int) = 1
      (long)= 10000
Bilangan Real x = 50.20000076293945
Bilangan Real y = 50.2
BUILD SUCCESSFUL (total time: 1 second)
```

Activate  
Go to Settings

#### Penjelasan :

Program ini berisi contoh sederhana untuk mendefinisikan variabel-variabel bilangan bulat (short int, int, long int), karakter, bilangan riil.

- **public class ASIGNi** → dibuatkan sebuah class dengan nama ASIGNi yang memiliki access public.
- **public static void main(String[] args)** → merupakan method utama.
- Inisialisasi masing-masing tipe data (short,int,long,char) dgn nama variable dengan integer dan masing-masing tipe data memiliki variable (ks,ki,kl,c) :
  - short ks = 1;**
  - int ki = 1;**
  - long kl = 10000;**
  - char c = 65;**
- **char c1 = 'Z';** → inisialisasi karakter dengan karakter.
- **double x = 50.2f;** → dideklarasikan sebuah variable x dgn tipe data double yang diberi nilai 50.2
- **float y = 50.2f;** → dideklarasikan sebuah variable y dgn tipe data float yang diberi nilai 50,2, setelah nilai perlu ditambahkan notasi 'f' atau 'F'.
- **System.out.println ("Karakter = "+ c);** → mencetak karakter pada nilai variable c.
- **System.out.println ("Karakter = "+ c1);** → mencetak karakter pada nilai variable c1.
- **System.out.println ("Bilangan integer (short) = "+ ks);** → menampilkan nilai pada variable ks.

- **System.out.println ("\t(int) = "+ ki);** → menampilkan nilai pada variable ki.
- **System.out.println ("\t(long)= "+ kl);** → menampilkan data pada variable kl.
- **System.out.println ("Bilangan Real x = "+ x);** → menampilkan nilai pada variable x
- **System.out.println ("Bilangan Real y = "+ y);** → menampilkan nilai pada variable y

#### 4). Program 4

##### Output :

```
run:
Contoh membaca dan menulis, ketik nilai integer:
9
Nilai yang dibaca : 9BUILD SUCCESSFUL (total time: 11 seconds)
|
```

##### Penjelasan :

Membaca integer dengan menggunakan class scanner

- **import java.util.Scanner;** → class scanner.
- **public class BacaData** → dibuatkan sebuah class dengan nama BacaData yang memiliki access public.
- **public static void main(String[] args)** → merupakan method utama.
- **int a** → dibuatkan sebuah variable bertipe data int dengan nama 'a'.
- **Scanner masukan** → deklarasi var radius & masukan dari kboard.
- **System.out.print ("Contoh membaca dan menulis, ketik nilai integer: \n");** → menampilkan kalimat "Contoh membaca dan menulis, ketik nilai integer:" lalu kita menginputkan nilai integer yang ingin ditampilkan.
- **masukan = new Scanner(System.in);** → Deklarasi instans/objek representasi dari kelas Scanner.
- **a = masukan.nextInt();** → variable a akan membaca angka bertipe data int yang telah diinputkan sbelumnya.
- **System.out.print ("Nilai yang dibaca : "+ a);** → menampilkan / mencetak nilai a.

## 5). Program 5

### Output :

```
run:
hello
baca 1 karakter : b
baca 1 bilangan : 2
b
2
bye
BUILD SUCCESSFUL (total time: 7 seconds)
```

### Penjelasan :

Program bertujuan untuk membaca masing-masing 1 karakter dan 1 bilangan yang diinputkan menggunakan metode `readLine()`; atau `InputStream`.

Pertama dibuatkan variable bertipe data `char` dgn nama '**cc**' dan variable bertipe data `int` dgn nama '**bil**'. Objek `BufferedReader` dibuat dengan konstruktor yang menggunakan `Reader` sebagai argumen. Lalu akan tercetak pesan "hello".

Setelah itu user akan diminta memasukkan karakter pada konsol melalui keyboard dengan **`System.out.print ("Baca 1 karakter : ")`**.

**`cc =dataIn.readLine().charAt(0);`** akan membaca karakter `cc` yang telah diinputkan.

User akan diminta lagi memasukkan karakter pada konsol melalui keyboard dengan **`System.out.print ("Baca 1 bilangan : ")`**.

**`bil =Integer.parseInt(datAIn.readLine());`** perintah ini akan membaca nilai integer `bil` yang telah diinputkan.

Perintah **`System.out.print (cc +"\n" +bil+"\n");`** yang akan menampilkan nilai dan karakter dari data yang sudah diinput. Terakhir, perintah **`System.out.print ("bye \n");`** akan menampilkan kalimat "bye".

## 6). Program 6

### Output :

```
run:
5.0
6.0
2
3.200000047683716
53
53.0
53.0
3
3.14
BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan :

Program ini bertujuan untuk menampilkan/mencetak nilai dari setiap variable dengan tipe data yang berbeda-beda dengan perintah ***System.out.println*** disertai dengan pemanggilan nilai/variable masing-masing.

- Variable '**a bernilai 5**' dan '**b bernilai 6**' dengan tipe data **int**. Tipe data integer ini mencakup semua bilangan bulat atau bilangan yang tidak memiliki komponen pecahan.
- Variable '**d bernilai 2.f**', '**e bernilai 3.2f**' dgn tipe data **float**. Tipe data ini mencakup semua bilangan dengan komponen pecahan.
- Variable '**g bernilai 5**' dengan tipe data **char**. Tipe data (char) merupakan tipe data yang digunakan untuk menyimpan satu huruf, angka, tanda baca, simbol, atau spasi kosong. Tipe data ini harus menggunakan tanda petik.
- Variable '**k bernilai 3,14**' dengan tipe data **double**. Tipe data ini umumnya digunakan untuk tipe data decimal.

## 7). Program 7

### Output :

```
run:
a : 67
k : 45.0
d : 100.0
n : 9
m : 5
l : 3.2
k : 67.0
c : 9.0
l : 3.2
BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan :

Program diatas bertujuan untuk menjelaskan fungsi InputDialog kelas JOptionPane mengembalikan nilai yang diinput pengguna aplikasi dalam bentuk tipe String dan mengembalikan nilai yang diinput pengguna ke bentuk tipe data string.

- Dibuatkan variable dengan tipe data yang berbeda dan nama yang berbeda, masing-masing variable diisikan nilainya.

***n = String.valueOf(b);***

***m = String.valueOf(g);***

***l = String.valueOf(e);***

- Source code diatas mengubah nilai variabel integer menjadi string dan mengembalikan nilainya. Karenanya kita dapat menggabungkan nilai string lain ke variabel ini.

- **a = Integer.parseInt(n);** → mengkonversi string numerik integer ke tipe numerik integer, digunakan metoda parseInt dari kelas Integer. Variabel ini digunakan untuk menerima hasil konversi dari string numerik integer ke numerik integer

- **k = Double.parseDouble(m);** → m dalah variabel berisi string numerik double seperti "243.45". Seperti tipe int dan double ini, tipe numerik lain seperti byte, short, long dan float masing-masing mempunyai kelas dan metoda bersesuaian untuk mengkonversi nilai string numerik ke nilai numerik.



- **d = Float.parseFloat(l);** → `parseFloat()` di Kelas `Float` adalah metode bawaan di Java yang mengembalikan float baru yang diinisialisasi ke nilai yang diwakili oleh String yang ditentukan, seperti yang dilakukan oleh metode `valueOf` dari kelas `Float`.

## 8). Program 8

### Output :

```
run:
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1BUILD SUCCESSFUL (total time: 2 seconds)
```

### Penjelasan :

Program diatas bertujuan untuk menjelaskan pemakaian operator kondisional pada program. Operator yang digunakan adalah operator ternary. Operator ternary adalah operator yang terdiri dari 3 operand. Di dalam bahasa Java, operator ternary ini menggunakan tanda `?` : dan merupakan penulisan singkat dari kondisi `if else`.

Pertama dibuatkan variable bertipe data integer dgn nama `x` yang nilainya adalah 1 dan `y` yang nilainya adalah 2. Setelah itu akan dicetak nilai dari masing-masing variable.

**`System.out.print("hasil ekspresi = (x<y)?x:y = "+ ((x < y) ? x : y));`** adalah perintah untuk menampilkan nilai dari operator tersebut.

## 9).Program 9

### Output :

```
run:
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) = 3BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan :

Program bertujuan untuk menampilkan hasil pembagian integer dan juga menampilkan hasil pembagian casting.

**-System.out.print ("x/y (format integer) = "+x/y);**

**-System.out.print("\nx/y (format float) = "+x/y);**

Saat menjalankan perintah diatas maka hasil x/y yang tercetak pada output adalah nilai 0. Agar hasil output tidak 0 maka harus membuat variable-variabel yang baru dengan tipe data float kemudian menyalin nilai dari variable x dan y kedalam variable baru tersebut.

## 10). Program 10

### Output :

```
run:
Hello
Hello World
Welcome
BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan :

Program bertujuan untuk menampilkan kalimat yang diinputkan.

**System.out.print("Hello");** → menampilkan/mencetak nilai/kata "Hello".

**System.out.print("\nHello ");** → mencetak kata "Hello" dan ganti baris.

**System.out.println("World");** → Mencetak kata "World" dan ganti baris.

**System.out.println("Welcome");** → mencetak kata "Welcome".

## 11). Program 11

### Output :

```
run:  
Nilai i : 5  
Nilai j : 3  
BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan :

Program bertujuan menampilkan nilai dari variable yang menggunakan operator aritmetika (+) dan yang nilainya telah dinaikkan 1.

- **int i, j;** → dibuatkan variable i dan j dengan tipe data int.
- **i = 3;** → variable i diinputkan nilai 3.
- **j = i++** → isi variable j adalah nilai dari i yang telah dinaikkan 1.
- **System.out.println ("Nilai i : " + (++i) + "\nNilai j : " + j);**  
→ perintah diatas akan menampilkan nilai masing-masing dari variable.

## 12).Program 12

### Output :

```
run:
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0
BUILD SUCCESSFUL (total time: 1 second)
```

Activate W

### Penjelasan :

Program diatas bertujuan menjelaskan penggunaan operator Bitwise Binary &(and), >>(bitwise shift right), dan <<(bitwise shift left).

- **int n = 10;** → variable n dgn tipe data int bernilai 10.
- **int x = 1;** → variable x dgn tipe data int bernilai 1.
- **int y = 2;** → variable y dgn tipe data int bernilai 2.

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

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

**System.out.println ("y = "+ y);**

→ Perintah diatas akan mencetak/menampilkan nilai dari masing” variable.

- **System.out.println ("n & 8 = "+ (n & 8));** → perintah ini membandingkan 2 bit dan menghasilkan nilai1 hanya jika kedua bit bernilai 1.

- **System.out.println ("x & ~ 8 = "+ (x & ~8));** → perintah ini membandingkan 2 bit dan menghasilkan nilai1 hanya jika kedua bit bernilai 1 dan membalik (invert) nilai setiap bit.

- **System.out.println ("y << 2 = "+ (y << 2));** → perintah ini memindahkan susunan bit kekiri dan membuang bit paling kiri serta memberikan nilai 0 pada bit paling kanan

- **System.out.println ("y >> 3 = "+ (y >>3));** → perintah ini memindahkan susunan bit kekanan dan membuang bit paling kanan serta memberikan nilai 0 pada bit paling kiri.

### 13).Program 13

#### Output :

```
run:
i = 3
j = 0
i & j = 0
i | j = 3
i ^ j = 3
81.0
~i = -4
BUILD SUCCESSFUL (total time: 1 second)
```

#### Penjelasan :

Program diatas bertujuan menjelaskan penggunaan operator Bitwise Binary &(and), |(bitwise inclusive OR) , ^(bitwise exclusive OR), ~(bitwise complement).

- **System.out.println("i = "+ (int) i);** → perintah akan menampilkan nilai dari variable i yg dijadikan int.

- **System.out.println("j = "+ j);** → perintah akan menampilkan nilai dari variable j.

- **System.out.println("i & j = "+ (i & j));** → perintah ini membandingkan 2 bit dan menghasilkan nilai1 hanya jika kedua bit bernilai 1.

- **System.out.println("i | j = "+ (i | j));** → perintah akan membandingkan 2 bit dan menghasilkan nilai0 hanya jika kedua bit bernilai 0.

- **System.out.println("i ^ j = "+ (i ^ j));** → perintah ini membandingkan 2 bit dan menghasilkan nilai1 hanya jika kedua bit berbeda nilainya.

- **System.out.println(" ~i = "+ ~i);** → perintah akan membalik (invert) nilai setiap bit.

## 14. Program 14

### Output :

```
run:
true
false
true
true
true
BUILD SUCCESSFUL (total time: 1 second)
```

### Penjelasan :

Program bertujuan menjelaskan penggunaan operator logical dan operator bitwise.

- **if (true && true){ System.out.println(true && true); }** → diberikan suatu kondisi jika true&&true, maka akan tercetak hasil dari perintah yang memberikan nilai true jika kedua operand bernilai true.

- **if (true & true) { System.out.println(true & false); }** → diberikan suatu kondisi jika true&true, maka akan tercetak hasil dari perintah yang membandingkan 2 bit dan menghasilkan nilai1 hanya jika kedua bit bernilai 1. (hasilnya adalah false).

- **if (true) { System.out.println(true); }** → perintah akan menampilkan hasil true

- **if (true || true){ System.out.println(true); }** → kondisi jika true || true, maka akan menampilkan hasil dari perintah yang memberikan nilai false jika kedua operand bernilai false.

- **if (true | false) { System.out.println(true | false); }** → kondisi jika true | false, maka akan tercetak hasil dari perintah yang membandingkan 2 bit dan menghasilkan nilai0 hanya jika kedua bit bernilai 0. (hasilnya adalah true).

## 15.) Program 15

### Output :

```
run:
Nilai e = 10
Nilai k = 0
Nilai k = 4BUILD SUCCESSFUL (total time: 3 seconds)
```

### Penjelasan :

Program diatas menampilkan nilai dari variable e dan k

**public class Oper4** dibuatkan sebuah class dengan nama Oper4

**public static void main(String[] args)** merupakan method utama

**int i = 0;** diberikan sebuah variable bertipe data integer dengan nama i

**int j = 0;**

**char c = 8; char d = 10;**

**int e = (((int)c > (int)d) ? c: d);**

**int k = ((i>j) ? i: j);**

Inisialisasi masing-masing tipe data ( int,char) dengan nama variable dengan integer dan masing-masing tipe data memiliki variable (j,c,d,k,e)

**System.out.print ("Nilai e = "+ e);** akan mencetak nilai dari variable e

**System.out.print ("\nNilai k = "+ k);** akan mencetak nilai dari variable k

**i = 2;** nilai i=2

**j = 3;** nilai j=3

**k = ((i++>j++) ? i: j) ;** perulangan

**System.out.print ("\nNilai k = "+ k);** akan mencetak nilai dari variable k



## 16.) Program 16

### Output :

run:

Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output

Operasi Logical Bool

Bool1 && Bool2 = false

Bool1 || Bool2 = true

!Bool1 = false

Bool1 ^Bool2 = true

Operasi Numerik

i + j = 7

i - j = 3

i / j = 2

i \* j = 10

i / j = 2

i % j = 1

Operasi Numerik

x + y = 10.0

x - y = 0.0

x / y = 1.0

x + y = 25.0

Operasi Relasional Numerik

i == j = false

i != j = true

i < j = false

i > j = true

i <= j = false

i >= j = true

Operasi Relasional Numerik

x != y = false

x < y = false

x > y = false

x <= y = true

x >= y = true

BUILD SUCCESSFUL (total time: 1 second)

## Penjelasan :

Program ini bertujuan untuk menjelaskan penggunaan operasi logical bool, operasi numerik, dan operasi relasional numerik.

### "Operasi Logical Bool"

**Bool1 = true; Bool2 = false;** → dibuatkan variable Bool1 yang isinya adalah true dan bool2 yang isinya adalah false.

**TF = Bool1 && Bool2;** → Operator yang digunakan adalah logical AND, memberikan nilai true jika kedua operand bernilai true.

**System.out.println ("Bool1 && Bool2 = " + TF );** → akan mencetak nilai dari hasil perbandingan operator logical AND.

**TF = Bool1 || Bool2;** → Operator logical OR, memberikan nilai false jika kedua operand bernilai false.

**System.out.println ("Bool1 || Bool2 = " + TF );** → akan mencetak nilai dari hasil perbandingan operator logical OR.

**TF = ! Bool1;** → operator logical negation(unary), memberikan nilai kebalikan (negasi) dari operand.

**System.out.println ("!Bool1 = " + TF );** → perintah ini akan mencetak hasil dari nilai operator logical negation(unary).

**TF = Bool1 ^Bool2;** → Operator bitwise exclusive OR, membandingkan dua bit dan menghasilkan nilai 1(true) hanya jika kedua bit berbeda nilainya.

**System.out.println ("Bool1 ^Bool2 = " + TF );** → akan mencetak nilai dari hasil perbandingan operator bitwise exclusive OR.

### "Operasi Numerik"

**i = 5; j = 2 ;** → dibuatkan sebuah variable i dgn nilai 5 dan j dgn nilai 2.

**hsl = i+j;** → operator penjumlahan, akan menambah nilai dari setiap variabel.

**System.out.println ("i + j = " + hsl );** → akan menampilkan hasil dri penjumlahan.

**hsl = i - j;** → operator pengurangan, akan mengurangi nilai dari setiap variable

**System.out.println ("i - j = " + hsl );** → akan menampilkan hasil pengurangan

`hsl = i / j;` → operator pembagian, proses membagi nilai dari tiap variabel  
`System.out.println ("i / j = " + hsl );` → akan menampilkan hasil pembagian  
`hsl = i * j;` → operator perkalian, proses mengalikan nilai dari tiap variabel  
`System.out.println ("i * j = " + hsl );` → akan menampilkan hasil perkalian.  
`hsl = i / j;` → operator pembagian, proses membagi nilai dari tiap variabel  
`System.out.println ("i / j = " + hsl );` → akan menampilkan hasil pembagian  
`hsl = i%j;` → operator modulus (sisa dari hasil pembagian).  
`System.out.println ("i % j = " + hsl );` → akan mencetak sisa dari hasil pembagian nilai.

### **“Operasi Relational Numerik”**

`TF = (x != y);` → Operator perbandingan tdk sama dengan.  
`System.out.println ("x != y = " + TF );` → menampilkan hasil perbandingan.  
`TF = (x < y);` → operator kurang dari.  
`System.out.println ("x < y = " + TF );` → menampilkan hasil perbandingan.  
`TF = (x > y);` → operator lebih dari  
`System.out.println ("x > y = " + TF );` → menampilkan hasil perbandingan  
`TF = (x <= y);` → operator kecil dari atau sama dengan.  
`System.out.println ("x <= y = " + TF );` → menampilkan hasil perbandingan  
`TF = (x >= y);` → operator besar dari atau sama dengan.  
`System.out.println ("x >= y = " + TF );` → menampilkan hasil perbandingan

**Nilai selain true akan dikembalikan apabila ekspresinya bernilai benar, dan fasle apabila ekspresinya salah**

## 2. Tugas Praktek : Praktek Program Java : Standar IO dan Struktur Kontrol

### 1). Program 1

#### Output :

run:

```
Baca string dan Integer:
masukkan sebuah string: 4
String yang dibaca : 4BUILD SUCCESSFUL (total time: 10 seconds)
|
```

#### Penjelasan :

**public class BacaString** { dibuatkan sebuah class dengan nama Bacastring

**public static void main(String[] args)** merupakan method utama

**String str;** dibuatkan sebuah variable bertipe data string dengan nama str

**System.out.print ("\nBaca string dan Integer: \n");** menampilkan kalimat "Baca string dan Integer" lalu menginput nilai integer yang ingin ditampilkan

**System.out.print("masukkan sebuah string: ");** menampilkan kalimat "masukkan sebuah string"

**str= datAIn.readLine();**

**System.out.print ("String yang dibaca : "+ str);** akan mencetak nilai dari variable str

## 2). Program 2

Output :

```
run:  
Program akan looping, akhiri dengan ^c
```

Penjelasan :

**class ForEver {** → dibuatkan sebuah class dengan nama ForEver

**public static void main(String[] args) {** → merupakan method utama

**System.out.println("Program akan looping, akhiri dengan ^c");** → menampilkan kalimat "program akan looping, akhiri dengan ^c"

**while (true);** kondisi tersebut bernilai true

**System.out.print ("Print satu baris ....\n");** → maka akan menampilkan kalimat "print satu baris" di baris baru

### 3). Program 3

#### Output :

```
run:
Contoh IF satu kasus
Ketikkan suatu nilai integer : 5

Nilai a positif 5BUILD SUCCESSFUL (total time: 14 seconds)
|
```

#### Penjelasan :

**public class If1** { dibuatkan sebuah class dengan nama If1 yang memiliki akses public

**public static void main(String[] args)** { merupakan method utama

**int a;** dibuatkan sebuah variable bertipe data int dengan nama 'a'

**Scanner masukan=new Scanner(System.in);** deklarasi objek representasi dari kelas scanner

**System.out.print ("Contoh IF tiga kasus \n");** mencetak kalimat "contoh if tiga kasus"

**System.out.print ("Ketikkan suatu nilai integer :");** lalu menginputkan nilai integer yang ingin ditampilkan

**a=masukan.nextInt();** variable a akan membaca angka bertipe data int yang telah di input sebelumnya

**if (a >= 0){** jika  $a \geq 0$

**System.out.println ("Nilai a positif "+ a);** maka akan menampilkan nilai positif pada variable a **System.out.println ("Nilai a negatif "+ a);** maka akan menampilkan nilai negative pada variable a

#### 4). Program 4

##### Output :

```
run:
Contoh IF dua kasus
Ketikkan suatu nilai integer :5
Nilai a positif 5
BUILD SUCCESSFUL (total time: 7 seconds)
|
```

##### Penjelasan :

**public class If2** { dibuatkan sebuah class dengan nama If2 yang memiliki akses public

**public static void main(String[] args)** { merupakan method utama

**int a;** diberikan sebuah variable bertipe data int dengan nama a

**Scanner masukan=new Scanner(System.in);** deklarasi objek representasi dari kelas scanner

**System.out.print ("Contoh IF dua kasus \n");** menampilkan kalimat “contoh if dua”

**System.out.print ("Ketikkan suatu nilai integer :");** menampilkan kalimat “ketikkan suatu nilai integer” lalu input nilai integer yang ingin ditampilkan

**a=masukan.nextInt();** variable a akan membaca angka bertipe data int yang telah di inputkan sebelumnya

**if (a >= 0){** jika a lebih besar sama dengan 0 maka

**System.out.println ("Nilai a positif "+ a);** maka akan mencetak karakter pada variable a yaitu nilai a positif

**else /\* a < 0 \*/** jika **a < 0** maka kondisi kedua di jalankan

**System.out.println ("Nilai a negatif "+ a);** maka akan mencetak nilai a negatif

## 5). Program 5

### Output :

```
run:
Contoh IF tiga kasus
Ketikkan suatu nilai integer :8
Nilai a positif 8
BUILD SUCCESSFUL (total time: 8 seconds)
|
```

### Penjelasan :

**public class If3** { dibuatkan sebuah class dengan nama If3 yang memiliki akses public

**public static void main(String[] args)** { merupakan method utama

**int a;** dibuatkan sebuah variable bertipe data int dengan nama 'a'

**Scanner masukan=new Scanner(System.in);** deklarasi objek representasi dari kelas scanner

**System.out.print ("Contoh IF tiga kasus \n");** akan menampilkan kalimat "contoh if tiga kasus"

**System.out.print ("Ketikkan suatu nilai integer :");** lalu menginputkan nilai integer yang ingin ditampilkan

**a=masukan.nextInt();** variable a akan membaca angka bertipe data int yang telah di input sebelumnya

**if (a > 0){** jika  $a > 0$

**System.out.println ("Nilai a positif "+ a);** maka akan menampilkan nilai positif pada variable a

**else if (a == 0){** kondisi lainnya jika  $a == 0$ , maka

**System.out.println ("Nilai Nol "+ a);** akan menampilkan nilai 0 pada variable a

**else /\* a < 0** kondisi lain jika  $a < 0$ , maka

**System.out.println ("Nilai a negatif "+ a);** akan menampilkan nilai negative pada variable a



## 6). Program 6

### Output :

```
run:
true
benar
BUILD SUCCESSFUL (total time: 0 seconds)
```

### Penjelasan :

**public class KasusBoolean** dibuatkan sebuah class dengan nama KasusBoolean

**public static void main(String[] args)** merupakan method utama

**bool= true;** yaitu tipe data yang memiliki kondisi benar

**if(bool)** jika kondisi

**System.out.print("true\n");** bernilai true akan menampilkan kalimat 'true'

**else** kondisi lainnya

**System.out.print("false\n");** "bernilai salah akan menampilkan kalimat 'false'

**if(!bool) {** jika kondisi

**System.out.print("salah\n");** bernilai salah akan mencetak kalimat 'salah'

**else** kondisi lainnya

**System.out.print("benar\n");** bernilai benar akan mencetak kalimat 'benar'

## 7). Program 7

### Output :

```
run:
Ketikkan sebuah huruf, akhiri dengan RETURN
a
Yang anda ketik adalah a
BUILD SUCCESSFUL (total time: 7 seconds)
|
```

### Penjelasan :

**public class KasusSwitch** { dibuatkan sebuah class dengan nama KasusSwitch yang memiliki akses public

**public static void main(String[] args)** { merupakan method utama

**char cc;** tipe data char dengan variable cc

**Scanner masukan=new Scanner(System.in);** deklarasi objek representasi dari kelas scanner **System.out.print ("Ketikkan sebuah huruf, akhiri dengan RETURN \n");** Akan menampilkan kalimat “ ketikkan sebuah huruf”

**cc=masukan.next().charAt(0);**

**switch (cc)**

**case 'a': { System.out.print (" Yang anda ketik adalah a \n"); break;}** Jika yang anda input adalah a maka akan muncul kalimat “ yang anda ketik adalah a”

**case 'u': { System.out.print (" Yang anda ketik adalah u \n"); break;}** Jika yang anda input adalah u maka akan muncul kalimat “ yang anda ketik adalah u”

**case 'e': { System.out.print (" Yang anda ketik adalah e \n"); break;}** Jika yang anda input adalah e maka akan muncul kalimat “ yang anda ketik adalah e”

**case 'i': { System.out.print (" Yang anda ketik adalah i \n"); break;}** Jika yang anda input adalah i maka akan muncul kalimat “ yang anda ketik adalah

**case 'o': { System.out.print (" Yang anda ketik adalah o \n"); break;}** jika yang anda input adalah o maka akan muncul kalimat “ yang anda ketik adalah o”

**default:**

**System.out.print (" Yang anda ketik adalah huruf mati \n");** jika yang anda ketik (a,i,u,e,o) adalah salah, maka akan menampilkan ' yang anda ketik adalah huruf mati'

8). Program 8

**Output :**

```
run:
Jari-jari lingkaran =14
Luas lingkaran = 615.734
Akhir program
BUILD SUCCESSFUL (total time: 30 seconds)
```

**Penjelasan :**

**public class Konstant** { dibuatkan sebuah class dengan nama Konstant yang memiliki akses public

**public static void main(String[] args)** { merupakan method utama

**final float PHI = 3.1415f;**

**float r;** variable r dengan tipe data float

**Scanner masukan=new Scanner(System.in);** deklarasi objek representasi dari kelas scanner

**System.out.print ("Jari-jari lingkaran =");** akan mencetak 'jari-jari lingkaran='

**r = masukan.nextFloat();** masukkan nilai yang bertipe data float

**System.out.print ("Luas lingkaran = "+ (PHI \* r \* r)+"\n");** 'menampilkan hasil dari perhitungan luas lingkaran'

**System.out.print ("Akhir program \n");** 'menampilkan hasil akhir program'

## 9).Kode Program 9

### Output :

```
run:
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
2
5
Ke dua bilangan : a = 2 b = 5
Nilai b yang maksimum: 5
BUILD SUCCESSFUL (total time: 18 seconds)
|
```

### Penjelasan:

**public class Max2** dibuatkan sebuah class dengan nama Max2

**int a, b** diberikan sebuah variable bertipe data integer dengan nama a dan b

**System.out.print ("Maksimum dua bilangan : \n");** yaitu suatu perintah untuk menampilkan hasil dari 2 bilangan yang dimasukkan.

**a=masukan.nextInt();** yaitu masukkan nilai pertama

**b=masukan.nextInt();** yaitu masukkan nilai kedua

**if (a >= b)** jika a lebihkecil sama dengan b

**System.out.println ("Nilai a yang maksimum "+ a);** mencetak hasil maksimum dari nilai a yang dimasukkan

**else /\* a > b \*/{** jika kondisi pertama salah maka kondisi kedua dijalankan

**System.out.println ("Nilai b yang maksimum: "+ b);** mencetak hasil maksimum dari nilai b yang dimasukkan

## 10).Kode Program 10

### Output :

```
run:
Baca N, print 1 s/d N N = 2
1
2
Akhir program

BUILD SUCCESSFUL (total time: 21 seconds)
```

---

### Penjelasan:

**public class PriFor** dibuatkan nama class dengan nama prifor **System.out. ("Baca N, print 1 s/d N ");** mencetak nilai n **System.out. ("N = ");** mencetak nilai n yang dimasukkan **for (i = 1; i <= N; i++){** perulangan **System.out.println (i); }** mencetak hasil dari nilai n yang dimasukkan **System.out.println ("Akhir program \n");** mencetak hasil dari akhir program yang sudah dijalankan.

## 11).Kode Program 11

### Output :

```
run:
Nilai N >0 = 2
Print i dengan ITERATE :
1
2
BUILD SUCCESSFUL (total time: 7 seconds)
```

### Penjelasan:

**public class PrintIterasi** dibuatkan class dengan nama printIterasi

**int N;** diberikan variable bertipe data integer dengan nama N

**int i;** diberikan variable bertipe data integer dengan i

**System.out.print ("Nilai N >0 = ");** mencetak nilai n yang dimasukkan

**i = 1;** nilai i =1

**System.out.print ("Print i dengan ITERATE : \n");** mencetak nilai i dengan iterate

**for {**perulangan

**System.out.println(i);** mencetak nilai i

**if (i == N)** jika nilai i sama dengan n

**break;** berhenti

**else i++;** jika nilai i dinaikkan satu

**(i == N)** jika nilai i sama dengan n

## 12).Kode Program 12

### Output :

```
run:
Nilai N >0 = 1
Print i dengan REPEAT:
1
BUILD SUCCESSFUL (total time: 5 seconds)
```

### Penjelasan:

**Public Class Repeat** dibuatkan class dengan nama repeat  
**int N;** diberikan variable bertipe data integer dengan nama N  
**int i;** diberikan variable bertipe data integer dengan nama i  
**System.out.print ("Nilai N >0 = ");** mencetak nilai n yang dimasukkan  
**N = masukan.nextInt();** nilai n yang dimasukkan  
**i = 1;** nilai i=1  
**System.out.print ("Print i dengan REPEAT: \n");** mencetak nilai i dengan repeat  
**Do**  
**i++;** jika nilai i dinaikkan satu  
**while (i <= N);** kondisi perulangan jika nilai lebih kecil daripada nilai n

### 13).Kode Program 13

#### Output :

```
run:
Nilai N >0 = 3
Print i dengan WHILE:
1
2
3
BUILD SUCCESSFUL (total time: 2 seconds)
```

#### Penjelasan:

**public class PrintWhile** dibuatkan class dengan nama printwhile

**int N;** diberikan variable bertipe data integer dengan nama N

**int i;** diberikan variable bertipe data integer dengan nama i

**System.out.print ("Nilai N >0 = ");** mencetak nilai n

**N = masukan.nextInt();** masukkan nilai integer n

**i = 1;** nilai i sama dengan satu

**System.out.print ("Print i dengan WHILE: \n");** mencetak nilai i dengan while

**while (i <= N)** kondisi perulangan jika nilai i lebih kecil sama dengan n

**System.out.println (i);** diproses

**i++;** nilai i dinaikkan satu

**(i > N)** nilai i lebih besar dengan n



#### 14).Kode Program 14

##### Output :

```
run:
Nilai N >0 = 4
Print i dengan WHILE (ringkas):
1
2
3
4
BUILD SUCCESSFUL (total time: 11 seconds)
|
```

##### Penjelasan:

**public class PrintWhile1** { dibuatkan class dengan nama printwhile1

**int N;** diberikan variable bertipe data integer dengan nama N

**int i;** diberikan variable bertipe data integer dengan nama i

**int i = 1;** nilai i samadengan satu

**System.out.print ("Nilai N >0 = ");** mencetak nilai n lebihbesar 0

**N = masukan.nextInt();** masukkan nilai n

**System.out.print ("Print i dengan WHILE (ringkas): \n");** mencetak nilai i dengan while

**while (i <= N){** kondisi perulangan jika nilai i lebih kecil sama dengan n

**System.out.println (i++);** maka akan mencetak nilai i dinaikkan satu

**(i > N)** jika nilai i lebih besar dari n

## 15).Kode Program 15

### Output :

```
run:
Masukkan nilai x (int), akhiri dg 999 : 23
Masukkan nilai x (int), akhiri dg 999 : 21
Masukkan nilai x (int), akhiri dg 999 : 11
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 55
BUILD SUCCESSFUL (total time: 17 seconds)
|
```

### Penjelasan:

**public class PrintXinterasi** dibuatkan class dengan nama printxinterasi  
**int Sum=0;** diberikan variable bertipe data integer  
**int x;** diberikan variable bertipe data integer dengan nama X  
**System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");** mencetak nilai x  
**x = masukan.nextInt();** nilai x yang dimasukkan  
**if (x == 999){** kondisi perulangan jika nilai x sama dengan 999  
**System.out.print ("Kasus kosong \n");** mencetak jika nilai kosong  
**else**  
**Sum = x;** nilai x ditambahkan  
**System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");** mencetak nilai x  
**if(x==999)** jika x sama dengan 999  
**break;** kondisi berhenti  
**else{**  
**Sum = Sum + x;** proses penjumlahan  
**System.out.println("Hasil penjumlahan = "+ Sum);** mencetak hasil setelah dijumlahkan

## 16).Kode Program 16

### Output :

```
run:
Masukkan nilai x (int), akhiri dg 999 : 10
Masukkan nilai x (int), akhiri dg 999 : 12
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 22
BUILD SUCCESSFUL (total time: 21 seconds)
```

### Penjelasan:

**public class PrintXRepeat** { diberikan class dengan nama PrintXRepeat  
**int Sum;** diberikan variable bertipe data integer untuk penjumlahan  
**int x;** diberikan variable bertipe data integer dengan nama x  
**System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");** mencetak nilai x yang dimasukkan  
**x = masukan.nextInt();** masukkan nilai x  
**if (x == 999){** jika x sama dengan 999  
**System.out.print("Kasus kosong \n");** mencetak jika bernilai kosong  
**else {** jika kondisi perulangan pertama bernilai salah maka kondisi kedua dijalankan  
**Sum = 0;** menambahkan nilai 0  
**do{**  
**Sum = Sum + x;** nilai x yang dimasukkan akan ditambahkan  
**System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");** mencetak nilai x yang dimasukkan  
**x = masukan.nextInt();** nilai x yang dimasukkan

**while (x != 999);** jika nilai x tidak sama dengan 999  
**System.out.println ("Hasil penjumlahan = "+Sum);** mencetak hasil setelah dijumlahkan

#### 17).Kode Program 17

##### Output :

```
run:
Masukkan nilai x (int), akhiri dg 999 : 12
Masukkan nilai x (int), akhiri dg 999 : 13
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 25
BUILD SUCCESSFUL (total time: 11 seconds)
|
```

##### Penjelasan:

**Public class PrintXWhile** dibuatkan class dengan nama PrintXWhile

**int x;** diberikan variable bertipe data integer dengan nama x

**Sum = 0;** penjumlahan sama dengan 0

**System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");** mencetak nilai x yang dimasukkan

**x = masukan.nextInt();** nilai x yang dimasukkan

**(x != 999)** jika x tidak sama dengan 999{

**Sum = Sum + x;** proses penjumlahan

**System.out.print("Masukkan nilai x (int), akhiri dg 999 : ");** mencetak nilai x

**x = masukan.nextInt();** nilai x yang dimasukkan

**System.out.println("Hasil penjumlahan = "+ Sum);** mencetak hasil dari penjumlahan

## 18). Kode Program 18

### Output :

```
run:
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
3
5
Ke dua bilangan : a = 3 b = 5
Maksimum = 5
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a = 5 b = 3
BUILD SUCCESSFUL (total time: 9 seconds)
```

### Penjelasan:

**public class SubProgram {** dibuatkan class dengan nama subprogram  
**return ((a >= b) ? a : b);** kembalikan nilai a jika lebih besar dengan nilai b  
**public static void tukar (int a, int b)** menukar dua bilangan a dan b  
**int temp;** diberikan variable bertipe data integer  
**temp = a;** temp sama dengan a  
**a = b;** jika a sama dengan b  
**b = temp** b sama dengan temp  
**System.out.println ("Ke dua bilangan setelah tukar: a = "+ a +" b = "+ b);**  
mencetak dua bilangan a dan b setelah dijumlahkan  
**System.out.print ("Maksimum dua bilangan \n");** mencetak dua bilangan yang dimasukkan  
**System.out.print ("Ketikkan dua bilangan,pisahkan dg RETURN : \n");** mencetak kedua bilangan yang dipisahkan

**a = masukan.nextInt();** nilai a yang dimasukkan  
**b = masukan.nextInt();** nilai b yang dimasukkan  
**System.out.println ("Ke dua bilangan : a = "+ a +" b = "+ b);** mencetak kedua bilangan setelah diproses dalam penjumlahan  
**System.out.println ("Maksimum = " + (maxab(a, b)));** mencetak dua bilangan yaitu bilangan a dan b  
**System.out.print("Tukar kedua bilangan... \n");** mencetak kedua bilangan yang sudah ditukar  
**tukar (a, b);** nilai a dan b yang sudah ditukar

19).Kode Program 19

### Output :

```

run:
Contoh IF tiga kasus
Temperatur (der. C) = 33
Wujud air cair
33BUILD SUCCESSFUL (total time: 6 seconds)
|
  
```

### Penjelasan:

**public class Tempair** dibuatkan class dengan nama Tempair

**int T;** diberikan variable bertipe data integer dengan nama T

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

**System.out.print ("Temperatur (der. C) = ");** mencetak kasus pertama temperatur dengan nilai T yang sudah dimasukkan

**T=masukan.nextInt();** nilai T yang dimasukkan

**if (T < 0) {** jika nilai t lebih kecil dari 0

**System.out.print ("Wujud air beku \n"+ T);**  
mencetak

kasus kedua yaitu wujud air beku dengan nilai T yang sudah dimasukkan

**else if ((0 <= T) && (T <= 100)){** kondisi perulangan jika keduanya bernilai benar

**System.out.print ("Wujud air cair \n"+ T);** mencetak kasus ketiga yaitu wujud air cair ditambah dengan nilai T yang dimasukkan  
**else if (T > 100){** kondisi perulangan jika nilai T lebih besar dari 100  
**System.out.print ("Wujud air uap/gas \n"+ T); }**; mencetak wujud air yaitu uap atau gas