

Muhammad Afrizaldi Attalah

A2

13020220057

Source code

### 1. Asgdll

```
1 public class Asgdll {
2     public static void main(String[] args){
3         float f= 20.0f;
4         double fii;
5
6         fii=10.0f;
7         System.out.println ("f : "+f);
8         System.out.println ("fii: "+fii);
9     }
10 }
```

```
f : 20.0
fii: 10.0
```

Penjelasan code: pada kode diatas menjelaskan tentang mendeklarasikan variable, dimana pada variable f dan fii di deklarasikan dengan nilai 20.0f dan 10.0f dengan tipe data float dan double, setelah itu dipanggil pada baris kode 6 dan 7.

### 2. Assign

```
1 public class Assign {
2
3     public static void main(String[] args){
4         int i;
5         System.out.println ("Hello\n");
6         i = 5;
7         System.out.println("Ini nilai i : " + i);
8     }
9 }
```

```
Hello

Ini nilai i : 5

Process finished with exit code 0
```

Penjelasan:

pada source code diatas menjelaskan tentang pendeklarasian variable dengan tipe data int(bilangan bulat), yang kemudian di masukkan pada variable i, kemudian di panggil pada baris kode ke-7.

### 3. ASIGNi

```
1 public class ASIGNi {
2     public static void main(String[] args) {
3         short ks = 1;
4         int ki = 1;
5         long kl = 10000;
6         char c = 'A';
7         char c1 = 'Z';
8         double x = 50.2f;
9         float y = 50.2f;
10
11         System.out.println("Karakter = " + c);
12         System.out.println("Karakter = " + c1);
13     }
```

```
13         System.out.println("Karakter = " + c);
14         System.out.println("Karakter = " + c1);
15
16         System.out.println("Bilangan integer (short) = " + ks);
17         System.out.println("\t(int) = " + ki);
18         System.out.println("\t(long)= " + kl);
19         System.out.println("Bilangan Real x = " + x);
20         System.out.println("Bilangan Real y = " + y);
21     }
22 }
```

```

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

```

Penjelasan:

Source code diatas menjelaskan pendklarasian variable bilangan bulat, pecahan dan karakter, dimana variable bulat dan pecahan di deklarasikan pada variable ks, ki, kl, x, y dengan tipe data bilangan bulat int dan long int lalu bilangan pecaha double dan float yang dipanggil pada baris 17-21.

#### 4. BacaData

```

1  import java.util.Scanner;
2  public class BacaData {
3      public static void main(String[] args) {
4          int a;
5          Scanner masukan;
6          System.out.print ("Contoh membaca dan menulis, ketik nilai integer: \n");
7          masukan= new Scanner (System.in);
8          a= masukan.nextInt();
9          System.out.print("Nilai yang dibaca : "+ a);
10     }
11 }

```

```

Contoh membaca dan menulis, ketik nilai integer:
440011
Nilai yang dibaca : 440011
Process finished with exit code 0

```

Penjelasan:

source code baca data, menjelaskan tentang penggunaan kelas scanner untuk membaca input dari user, dimana pada baris ke-5 akan di buat variable yang bertipe data scanner, lalu pada pada baris ke-7 akan membuat objek scanner baru dengan system.in sebagai argument untuk mengambil input dari keyboard, lalu pada baris ke-8 berguna untuk membaca input dari keyboard lalu memasukkan nya ke kdalam variable a.

#### 5. Bacakarakter

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class Bacakarakter {
    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 dataIn = new BufferedReader(newInputStreamReader(System.in));//
    }
}

```

```

        System.out.print ("hello\n");
        System.out.print("baca 1 karakter : ");

        cc =dataIn.readLine().charAt(0);
        System.out.print("baca 1 bilangan : ");

        bil =Integer.parseInt(dataIn.readLine());

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

```

```

Bacakarakter
hello
baca 1 karakter : Z
baca 1 bilangan : 2
Z
2
bye

```

Penjelasan:

Source code diatas menjelaskan tentang penginputan dengan tipe data string dan karakter, pada 3 baris kode pertama terdapat beberapa kelas yang diperlukan dalam melakukan proses input dari keyboard.

Setelah melakukan pendeklarasian variable pada baris ke-7 dan 8, kemudian pada baris kode ke 9 dan 10 terdapat pembuatan objek baru 'InputStreamReader' untuk mengonversi byte mejadi karakter lalu 'BufferedReader' untuk membaca karakter dengan lebih efisien.

lalu pada baris ke -17 berfungsi untuk membaca baris input dan mengambil karakter pertama dari hasil input.

sedangkan, Integer.parseInt() digunakan untuk mengonversi string input menjadi bilangan bulat.

## 6. Casting1

```

1 public class Casting1 {
2     public static void main(String[] args){
3         int a=5, b=6;
4         float d=2.f, e=3.2f;
5         char g='5';
6         double k=3.14;
7
8         System.out.println((float)a);
9         System.out.println((double)b);
10        System.out.println((int)d);
11        System.out.println((double)e);
12        System.out.println((int)g);
13        System.out.println((float)g);
14        System.out.println((double)g);
15        System.out.println((int)k);
16        System.out.println((float)k);

```

```

5.0
6.0
2
3.200000047683716
53
53.0
53.0
3
3.14

```

Penjelasan:

pada source code diatas, melakukan proses convesi tipe data dimana pada variable a menghasilkan

hasil 5.0 saat di panggil sedangkan awalnya variable a mempunyai nilai 5. Begitu juga dengan variable lainnya yang mengalami perubahan/konversi tipe data.

## 7. Casting2

```
1 public class Casting2 {
2     public static void main(String[] args){
3         int a=8,b=9;
4         float d=2.5f,e=3.2f;
5         char g='5';
6         double k=3.14;
7         String n="67",m="45", l="100";
8
9         a = Integer.parseInt(n);
10        k = Double.parseDouble(m);
11        d = Float.parseFloat(l);
12        System.out.println("a : "+a+"\nk : "+k+"\nd : "+d);
13        n = String.valueOf(b);
14        m = String.valueOf(g);
15        l = String.valueOf(e);
16        System.out.println("n : "+n+"\nm : "+m+"\nl : "+l);
17        k = Double.valueOf(a).intValue();
18        /*Konversi Integer ke Double*/
19        double c = Integer.valueOf(b).doubleValue();
20        System.out.println("k : "+k+"\nc : "+c+"\nl : "+l);
21    }
22 }
```

```
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:

pada source code diatas menjelaskan fungsi yang sama yaitu tentang konversi tipe data, namun dengan menggunakan proses yang berbeda, sebelumnya menggunakan tipe data primitive namun pada casting di atas menggunakan tipe data class saat melakukan proses konversi tipe data. contoh pada baris ke-9 dilakukan konversi nilai n ke dalam tipe data a yaitu tipe data int

## 8. Ekspresi

```
1 public class Ekspresi {
2     public static void main(String[] args) {
3         int x = 1;
4         int y = 2;
5         System.out.print("x = " + x + "\n");
6         System.out.print("y = " + y + "\n");
7         System.out.print("hasil ekspresi = (x<y) ? x:y = " + ((x < y) ? x : y));
8     }
9 }
10
```

```
x = 1
y = 2
hasil ekspresi = (x<y) ? x:y = 1
Process finished with exit code 0
```

Penjelasan:

pada source code diatas menjelaskan tentang ekspresi, dimana pada baris ke-7 dilakukan perbandingan yang menghasilkan nilai true berarti menampilkan nilai x, maka dari itu hasil output setelah di panggil menghasilkan nilai dari X

## 9. Ekspresi1

```
1 public class Ekspresi1 {
2     public static void main(String[] args){
3         int x = 1; int y = 2; float fx; float fy;
4
5         System.out.print ("x/y (format integer) = " + x/y);
6         System.out.print ("\nx/y (format float) = " + x/y);
7
8         fx=x;
9         fy=y;
10
11        System.out.print ("\nx/y (format integer) = " + fx/fy);
12        System.out.print ("\nx/y (format float) = " + fx/fy);
13
14        System.out.print("\nfloat(x)/float(y) (format integer)= " + (float)x/(float)y);
15        System.out.print("\nfloat(x)/float(y) (format float) = " + (float)x/(float)y);
16
17        x = 10; y = 3;
18
19        System.out.print ("\nx/y (format integer) = " + x/y);
20        System.out.print ("\nx/y (format float) = " + x/y);
21    }
22 }
```

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:

pada baris ke-5 dan ke-6 dapat dijelaskan bahwa dalam java pembagian dari integer akan di bulatkan dengan angka terdekat dalam kasus ini 0,5 dibulatkan menjadi 0. Lalu pada bari ke-11 dan ke-12 dapat menghasilkan output 0.5 dikarenakan nilai dari x dan y di deklarasikan ke dalam fx dan fy yang bertipe data float.

sama juga dengan baris ke-14 dan ke-15, namun dengan cara mengubah nya dalam pemanggilan dengan cara mendklarasikan tipe data nya menjadi float.

## 10. Incr

```
1 public class Incr {
2     public static void main(String[] args) {
3         int i, j;
4         i = 3;
5         j = i++;
6         System.out.println ("Nilai i : " + (++i) + "\nNilai j : " + j);
7     }
8 }
```

Nilai i : 5  
Nilai j : 3

Penjelasan:

pada source kode diatas menjelaskan cara penggunaan increment di java, pada baris ke-5 bukan hanya melakukan inisialisasi namun nilai dari i akan ditambah sedangkan n ilai yang diambil oleh

variable j hanya nilai 3 pada baris ke-4. Lalu pada pemanggilannya di increment kan sekali lagi, maka dari itu hasil dari variable l adalah 5.

### 11. Oper1

```
1 public class Oper1 {  
2     public static void main(String[] args) {  
3  
4         int n = 10;  
5         int x = 1;  
6         int y = 2;  
7  
8         System.out.println("n = " + n);  
9         System.out.println("x = " + x);  
10        System.out.println("y = " + y);  
11        System.out.println("n & 8 = " + (n & 8));  
12        System.out.println("x & ~ 8 = " + (x & ~8));  
13        System.out.println("y << 2 = " + (y << 2));  
14        System.out.println("y >> 3 = " + (y >> 3));  
15    }  
16 }
```

```
n = 10  
x = 1  
y = 2  
n & 8 = 8  
x & ~ 8 = 1  
y << 2 = 8  
y >> 3 = 0
```

Penjelasan:

pada source code diatas, melakukan proses operasi bitwise, dimana nilai dari variable-variable di ubah menjadi bit dan kemudian akan diubah sesuai dengan operasi, contohnya nilai y pada baris ke-14 menghasilkan nilai 0 dikarenakan nilai y yang mempunyai nilai 2 akan di geser 3 bit kekanan.

### 12. Oper2

```
1 public class Oper2 {  
2     public static void main(String[] args) {  
3         char i, j;  
4         i = 3;  
5         j = 4;  
6         System.out.println("i = " + (int) i);  
7         System.out.println("j = " + j);  
8         System.out.println("i & j = " + (i & j));  
9         System.out.println("i | j = " + (i | j));  
10        System.out.println("i ^ j = " + (i ^ j));  
11        System.out.println(Math.pow(i, j));  
12        System.out.println(~i);  
13    }  
14 }
```

```
i = 3  
j = 4  
i & j = 0  
i | j = 7  
i ^ j = 7  
81.0  
~i = -4
```

Penjelasan:

karena variable i dan j diinialisasikan dengan nilai numerik maka nilai nya akan di ubah menjadikode karakter.

pada baris ke-6 i di konversi menjadi tipe data int maka dapat di tampilkan namun untuk j tidak bisa. lalu pada baris 8, 9, 10, dan 12 terjadi operasi bit wise yaitu |=not, ^=XOR, &=and.

Pada baris ke 11 terdapat proses pemangkatan dengan method math.pow yaitu 3 di pangkat 4= 81

### 13. Oper3

```
1 public class Oper3 {  
2     public static void main(String[] args) {  
3         if (true && true) { System.out.println(true && true);}  
4  
5         if (true & true) { System.out.println(true & false); }  
6  
7         if (true) { System.out.println(true); }  
8         if (true || true) { System.out.println(true);}  
9  
10        if (true|false) { System.out.println(true|false); }  
11    }  
12 }  
13 }
```

```
i = 3  
j = 0  
i & j = 0  
i | j = 7  
i ^ j = 7  
81.0  
~i = -4
```

Penjelasan:

pada source code diatas penggunaan && dan & mempunyai penggunaan yang seikit berbeda dimana penggunaan & (logika dan) akan mengecek ke dua variable yang dibandingkan sedangkan && akan langsung mengedintifikasi jika ada variable satu false maka variable selanjutnya sudah tidak akan dibaca. Begitu juga dengan || dan |

### 14. Oper4

```
1 public class Oper4 {  
2     public static void main(String[] args) {  
3         int i = 0;  
4         int j = 0;  
5         char c = 8; char d = 10;  
6         int e = (((int)c > (int)d) ? c: d);  
7         int k = ((i>j) ? i: j);  
8  
9         System.out.print ("Nilai e = " + e);  
10        System.out.print ("\nNilai k = " + k);  
11        i = 2;  
12        j = 3;  
13        k = ((i++>j++) ? i: j) ;  
14        System.out.print ("\nNilai k = " + k);  
15    }  
16 }
```

```
Nilai e = 10  
Nilai k = 0  
Nilai k = 4
```

Penjelasan:

source code diatas kebanyakan melakukan proses ternanry dimana dilakukan perbandingan antara 2 variabel, pada baris ke-9 menghasilkan nilai 10 dikarenakan nilai dari c=8 dan d=10 di bandingkan, kemudian menghasilkan nilai 10, pada baris ke-14 menghasilkan nilai 4 karena perbandingan antara i=2 dan j=3 lalu nilai yang akan dimasukkan kedalam k diincrement kan(3 diincrement menghasilkan 4)

### 15. Oprator

```

1 ▶ public class Oprator {
2 ▶     public static void main(String[] args) {
3
4         boolean Bool1, Bool2, IF ;
5         int i,j, hsl ;
6         float x,y,res;
7
8         System.out.println("Silahkan baca teksnya dan tambahkan perintah program di bawah ini untuk menampilkan output program");
9         Bool1 = true; Bool2 = false;
10        IF = Bool1 && Bool2 ;
11        IF = Bool1 || Bool2 ;
12        IF = ! Bool1 ;
13        IF = Bool1 ^Bool2;
14
15        i = 5; j = 2 ;
16        hsl = i+j;
17        hsl = i - j;
18        hsl = i / j;
19        hsl = i * j;
20        hsl = i /j ;
21        hsl = i%j ;

```

```

23        x = 5 ; y = 5 ;
24        res = x + y;
25        res = x - y;
26        res = x / y;
27        res = x * y;
28
29        IF = (i==j);
30        IF = (i!=j);
31        IF = (i < j);
32        IF = (i > j);
33        IF = (i <= j);
34        IF = (i >= j);
35
36        IF = (x != y);
37        IF = (x < y);
38        IF = (x > y);
39        IF = (x <= y);
40        IF = (x >= y);
41    }
42 }
43

```



## 16. PrintHello

```
1 ▶ public class PrintHello {  
2 ▶     public static void main(String[] args) {  
3         System.out.print("Hello");  
4         System.out.print("\nHello ");  
5         System.out.println("World");  
6         System.out.println("Welcome");  
7     }  
8 }  
9
```

```
PrintHello  
Hello  
Hello World  
Welcome
```

Penjelasan:

Source code diatas hanya menampilkan cara menampilkan teks dengan tipe data string.