

Praktikum 2

Dasar Pemrograman Java dan Tipe Data

Tujuan

Mengetahui dasar pemrograman Java dan syntax bahasa pemrograman Java .

Mengetahui bagaimana membuat objek, memberikan nilai dan mengkonversi tipe data pada variabel referensi Java.

Dasar Teori

OOP (*Object Oriented Programming*) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkirim pesan kepada objek yang lain.

Setiap objek memiliki dua karakteristik yaitu *attribute (state)* dan *behavior (method)*. *attribute* merupakan identitas atau informasi objek itu sendiri atau disebut juga sebagai *variable*, sedangkan *behavior* adalah tingkah laku atau apa yang dapat dilakukan oleh objek itu atau disebut juga sebagai *method*. Sebagai contoh *attribute* dari pesawat adalah nama pesawat, tipe, warna, tahun produksi, roda, dll. Sedangkan *Behavior* dari pesawat adalah cara menghidupkan mesin, cara menerbangkan pesawat (*take off*), cara menurunkan pesawat (*landing*), dll.



Attribute (State)
- Nama pesawat, Warna, Tipe, Tahun produksi

Behavior
- Cara menghidupkan mesin
- Cara menerbangkan pesawat (take off)
- Cara menurunkan pesawat (landing)

Kelas (*class*) merupakan cetakan atau *blueprint* atau *prototype*, hasil dari cetakan itu disebut objek. Contoh “pesawat” dibuat dengan rancangan yang sama, rancangan

“pesawat” ini disebut kelas. Hasil rancangan itu menjadi “pesawat Boeing 737” yang disebut objek. Pesawat-pesawat hasil rancangan tadi bisa memiliki warna, nama yang berbeda-beda tergantung perusahaan penerbangannya. Disini jelas bahwa objek memiliki sifat independen. Objek adalah instance dari kelas, satu kelas bisa memiliki beberapa objek.

Identifier adalah nama yang diberikan untuk variable (atribut), class atau method. Penamaan identifier harus diawali dengan karakter unicode, tanda \$ (dollar) atau tanda _ (underscore). Penamaan identifier ini bersifat case-sensitive dan tidak dibatasi panjang maksimum.

Java mempunyai 48 kata kunci, seperti *if*, *int*, *void*, dan lain-lain. Kata-kata kunci tersebut tidak bisa dipakai sebagai identifier. Selain kata kunci, Java juga mempunyai 3 kata literal, yaitu *true*, *false* dan *null*, yang juga tidak bisa dipakai untuk penamaan identifier.

Java mempunyai 8 tipe dasar, yaitu boolean, char, byte, short, int, long, float, dan double. Spesifikasi panjang bit dan range untuk masing-masing tipe adalah sebagai berikut:

Tipe	Panjang bit	Range
boolean	1	-
char	16	0 - $2^{16}-1$
byte	8	-2^7 - 2^7-1
short	16	-2^{15} - $2^{15}-1$
int	32	-2^{31} - $2^{31}-1$
long	64	-2^{63} - $2^{63}-1$
float	32	-
double	64	-

Nilai default untuk masing-masing tipe adalah sebagai berikut:

Tipe	Nilai Default
boolean	false
char	'\u0000'
byte	0
short	0
int	0
long	0L
float	0.0F
double	0.0

Casting diperlukan untuk mengkonversi dari suatu tipe ke tipe data yang lebih kecil panjang bitnya. Sedangkan promotion terjadi pada saat mengkonversi dari suatu tipe data ke tipe data yang lebih panjang bitnya.

Contoh:

```
int p = (int) 10L;
long i = 10;
```

A. Membetulkan syntax program yang salah

Jalankan program dibawah ini, bila terjadi kesalahan silahkan dibetulkan, amati hasilnya dan berikan analisa Anda.

Percobaan TestGreeting:

```
public class TestGreeting {
    public static void main(String[] args) {
        Greeting hello = new Greeting();
        hello.greet();
    }
}

class Greeting{
    public void greet(){
        System.out.println("Halo...");
    }
}
```

Percobaan 1:

```
public Test1 {
    public static void main(String[] args){
        System.out.println("Whats wrong with this program?");
    }
}
```

Percobaan 2:

```
public class Test2 {
    public static void main(String[] args){
        System.out.println("Whats wrong with this program?");
    }
}

public class TestAnother2{
    public static void main(String[] args){
        System.out.println("Whats wrong with this program?");
    }
}
```

Percobaan 3:

```
public class Test3 {
    public static void utama(String[] args){
        System.out.println("Whats wrong with this program");
    }
}
```

Percobaan 4:

```
public class Test4 {
    static void main(String[] args){
        System.out.println("Whats wrong with this program");
    }
}
```

Percobaan 5:

```
public class Test5 {
    public void main(String[] args){
        System.out.println("whats wrong with this program");
    }
}
```

Percobaan 6:

```
public class Test6 {
int[] x = new int [10];

public Test6(){
    for (int i=0;i<=10;i++){
        x[i]=i;
        System.out.println("Isi x ke:" + i + "=" + x[i]);
    }
}

public static void main(String[] args){
    Test6 test = new Test6();
}
}
```

Percobaan 7:

```
Public class Test7 {
int [] x = new int[10];
int val=20;

public static void main(String[] args){
    Test7 test=new Test7();
    test.Test7();
}

public void Test7(){
    for(int i=0;i< 10; i++){
        x[i]=i;
        float res=val/x[i];
        System.out.println("result"+"="+res);
    }
}
}
```

Percobaan 8:

```
public class Test8 {
String name="Tifatul Sembiring";

public void getName(){
    return name;
}
}
```

```

public static void main(String[] args){
    Test8 test = new Test8();
    System.out.println("Nama Menkominfo:"+ test.getName());
}
}

```

Percobaan 9:

```

public class Test9 {
    string text="abaaba";

    public void addText(){
        if(text.length()>=6)
        {
            text += text;
        }
        else
        {
            text += text.substring(3);
        }
        system.out.println("new text:"+ text);
    }

    public static void main(string[] args){
        test9 test = new test9 ();
        test.addText();
    }
}

```

Percobaan 10:

```

public class Test10 {

    boolean newAcct = true;
    double deposit =17.29;

    public void check(){
        if(newAcct==true)
        {
            balance=100.0;
        }
        if(deposit > 0)
        {
            balance += deposit;
        }
        System.out.println("Balance="+balance);
    }

    public static void main(String[] args){
        Test10 test = new Test10();
        test.check();
    }
}

```

B. Membuat objek, memberikan nilai dan mengkonversi tipe data

Jalankan program dibawah ini, amati hasilnya dan berikan analisa Anda.

Percobaan 1:

Program berikut ini menunjukkan bagaimana deklarasi variabel dan pemberian nilai.

```
public class Assign {
    public static void main(String [] args){
        int x,y;
        float z=3.414f;
        double w=3.1415;
        boolean truth=true;
        char c;
        String str;
        String str1="bye";
        c='A';
        str="Hi out there!";
        x=6;
        y=1000;

        //menampilkan isi variabel

        System.out.println("z="+z+" w="+w+" str1="+str1+"
c="+c+"\nstr="+str);

        System.out.println("x="+x+" y="+y+" truth="+truth);
    }
}
```

Percobaan 2:

Program berikut ini menggambarkan nilai default variable.

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

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

Percobaan 3:

Program berikut ini menggambarkan mekanisme “*pass by value*”.

```
public class PassTest {

    public static int changeInt(int value){
        value = 55;
        return value;
    }

    static void changeObjectRef(MyDate ref){
        ref = new MyDate(1,1,2010);
    }

    static void changeObjectAttr(MyDate ref){
        ref.setDay(11);
        ref.setMonth(9);
        ref.setYear(2012);
    }

    public static void main(String args[]){
        int val;
        val = 11;
        System.out.println("Int value is : " +val);
        val = changeInt(val);
        System.out.println("Int value is : " +val);

        MyDate date;
        date = new MyDate(22, 7, 1964);
        date.print();
        changeObjectRef(date);
        date.print();
        changeObjectAttr(date);
        date.print();
    }
}

class MyDate{
    int day, month, year;

    MyDate(int iday, int imonth, int iyear){
        day = iday;
        month = imonth;
        year = iyear;
    }

    public void setDay(int value){
        day = value;
    }

    public void setMonth(int value){
        month = value;
    }

    public void setYear(int value){
        year = value;
    }

    public void print(){
        System.out.println("Day:" + day + "Month:" + month + "Year:" +
year);
    }
}
```

Percobaan 4:

Program berikut ini menggunakan bilangan basis-2 (biner), basis-8 (octal), basis-10 (decimal), basis-16 (hexa-desimal).

```
public class Octal {
    public static void main(String [] args){
        int ten_biner    = 0b1010;
        int ten_octal    = 012;
        int ten_desimal  = 10;
        int ten_hexa     = 0xA;

        System.out.println("biner    = " +ten1);
        System.out.println("octal    = " +ten2);
        System.out.println("decimal  = " +ten3);
        System.out.println("hexa     = " +ten4);
    }
}
```

Percobaan 5:

Program berikut ini menggunakan bilangan “unicode”.

```
public class CobaUnicode {

    public static void main(String args[]){
        ch\u0061r a = 'a';
        char \u0062 = 'b';
        char c = '\u0063';
        String kata = "\u0061\u0062\u0063";

        System.out.println("a: "+a);
        System.out.println("b: "+b);
        System.out.println("c: "+c);
        System.out.println("kata: "+kata);
    }
}
```

Percobaan 6:

Program berikut ini menunjukkan proses “Primitive Conversion – Assignment” atau pemberian nilai dengan konversi tipe data secara otomatis. Hal ini diperbolehkan untuk konversi variabel ke tipe data yang lebih besar atau biasa disebut dengan “promotion”.

```
Public class Conversion01 {

    public static void main(String[] args){
        int i;
        double d;
        i = 10;
        d= i;

        System.out.println("Nilai i : " +i);
        System.out.println("Nilai d : " +d);
    }
}
```

Percobaan 7:

Program berikut ini menunjukkan pemberian nilai dengan sekaligus konversi tipe data dengan “casting” (short). Hal ini harus dilakukan untuk konversi variabel ke tipe data yang lebih kecil.

```
public class Conversion02 {  
  
    public static void main(String[] args){  
        double d = 1.2345;  
        short s = (short)d;  
  
        System.out.println("Nilai d = " +d);  
        System.out.println("Nilai s = " +s);  
    }  
}
```

Percobaan 8:

Program berikut ini menunjukkan pemberian nilai dengan sekaligus konversi tipe data dengan “casting” (byte). Hal ini harus dilakukan untuk konversi variabel ke tipe data yang lebih kecil.

```
public class Conversion03 {  
  
    public static void main(String [] args){  
        int i;  
        byte b;  
        i = 259;  
        b = (byte) i;  
  
        System.out.println("Nilai i = " +i);  
        System.out.println("Nilai b = " +b);  
    }  
}
```

Percobaan 9:

Program berikut ini menunjukkan pemberian nilai dengan sekaligus konversi tipe data dengan “casting” (short). Hal ini harus dilakukan untuk konversi variabel ke tipe data yang lebih kecil.

```
public class Conversion04 {  
  
    public static void main(String [] args){  
        double f = 2.32323;  
        short s = (short) f;  
  
        System.out.println("Nilai f = " +f);  
        System.out.println("Nilai s = " +s);  
    }  
}
```

Percobaan 10:

Program berikut ini menunjukkan pemberian nilai dengan sekaligus konversi tipe data dengan “casting” dari tipe data satu ke tipe data lainnya. Hal ini harus dilakukan untuk konversi variabel ke tipe data yang lebih kecil.

```
public class Conversion05 {  
  
    public static void main(String [] args){  
  
        short s = 9;  
        int i = 10;  
        float f = 11.1f;  
        double d = 12.2;  
  
        short x = (short) (s*i);  
        float y = (float) (f/d);  
        double z = (double) (x*y);  
  
        System.out.println("Nilai x = " +x);  
        System.out.println("Nilai y = " +y);  
        System.out.println("Nilai z = " +z);  
    }  
}
```

Lakukan percobaan promotion dan casting antar tipe-tipe data primitif. Lalu amati dan catatlah hasil konversinya dari suatu tipe ke tipe lainnya yang memerlukan suatu casting. Dari hasil analisa anda, buatlah suatu skema casting diantara tipe-tipe primitif. Jelaskan karakteristik “casting” dan “promotion”, dan mengapa perlu dilakukan?

Tugas

1. Menganalisa batasan maksimum dari suatu tipe

```
public class BigInteger {  
    public static void main(String args[]) {  
        long p=2147483648;  
    }  
}
```

Lakukan kompilasi pada file tersebut dan amati pesan kesalahannya. Lakukan analisa mengapa bisa terjadi kesalahan padahal batasan nilai maksimum dari suatu bilangan bertipe long adalah 263-1 (9223372036854775807)?. Kemudian berikanlah solusi yang tepat untuk mengatasi persoalan diatas.

2. Mencari panjang menit dari durasi waktu

Buatlah program dengan konsep OOP untuk

Input : jam awal, menit awal, jam akhir, menit akhir

Output : durasi waktu (dalam menit) antara jam awal menit awal dengan jam akhir menit akhir.

Cara memasukkan input melalui JOptionPane.

```
import javax.swing.JOptionPane;

public class InputPane {

    public static void main(String args[]) {
        int nilai;
        String str=JOptionPane.showInputDialog("Masukkan nilai :");
        nilai=Integer.parseInt(str);
        System.out.println(nilai);
        System.exit(0);
    }
}
```