**Bab 2**

**Dasar Pemrograman Java dan Operator**

# Bab 2 Dasar Pemrograman Java

## POKOK BAHASAN

* Identifier
* Kata kunci
* Tipe dasar
* Nilai default
* Casting dan promotion
* Bentuk Operator
* Jenis Oprator

## TUJUAN BELAJAR

Dengan praktikum ini mahasiswa diharapkan dapat:

* Mengetahui aturan penamaan identifier
* Mengenal kata-kata kunci yang ada di Java
* Mengetahui tipe-tipe dasar yang ada di Java
* Mengetahui pemberian nilai default untuk masing-masing tipe dasar
* Memahami bagaimana melakukan casting dan promotion
* Mengenal bentuk-bentuk operator
* Memahami berbagai macam jenis operator yang ada di Java

## Dasar Teori

Teori yang digunakan dalam Praktikum ini adalah sebagai berikut :

1. Java Identifier
2. Keyword dalam Java
3. Java Literals
4. Tipe Data Primitif
5. Variabel
6. Operator

Teori di atas merujuk pada modul dari JENI (Java Education Network Indonesia).

## Percobaan

### Percobaan 1 : Memberikan nilai ke suatu tipe

|  |
| --- |
| public class Assign {  public static void main(String args[]) {  boolean b=true;  System.out.println("Value b = " + b);  char c='C';  System.out.println("Value c = " + c);  byte bt=10;  System.out.println("Value bt = " + bt);  short s=20;  System.out.println("Value s = " + s);  int i=30;  System.out.println("Value i = " + i);  long l=40L;  System.out.println("Value l = " + l);  float f=3.14F;  System.out.println("Value f = " + f);  double d=3.14;  System.out.println("Value d = " + d);  }  } |

### Percobaan 2 : Mencetak nilai default dari tipe dasar

|  |
| --- |
| public class DefaultValue {  static boolean b;  static char c;  static byte bt;  static short s;  static int i;  static long l;  static float f;  static double d;  public static void main(String args[]) { System.out.println("Default value b = " + b); System.out.println("Default value c = " + c); System.out.println("Default value bt = " + bt); System.out.println("Default value s = " + s); System.out.println("Default value i = " + i); System.out.println("Default value l = " + l); System.out.println("Default value f = " + f); System.out.println("Default value d = " + d);  }  } |

### Percobaan 3 : Mengamati hasil perubahan nilai dari suatu operasi matematis

|  |
| --- |
| public class Divide {  public static void main(String[] arguments) {  float number1 = 15;  float number2 = 6;  float result = number1 / number2;  float remainder = number1 % number2; System.out.println(number1 + " divided by " + number2);  System.out.println("\nResult\tRemainder"); System.out.println(result + "\t" + remainder);  }  } |

### Percobaan 4 : Mengamati hasil perubahan nilai dari suatu operasi matematis

|  |
| --- |
| class Invest {  public static void main(String[] arguments) {  float total = 14000;  System.out.println("Original investment: $" + total);  // Inceases by 40 percent the first year  total = total + (total \* .4F);  System.out.println("After one year: $" + total);  // Loses $1,500 the second year total = total - 1500F;  System.out.println("After two years: $" + total);  // Increases by 12 percent the third year  total = total + (total \* .12F);  System.out.println("After three years: $" + total);  }  } |

### Percobaan 5 : Menampilkan bilangan oktal ke format decimal

|  |
| --- |
| public class Octal {  public static void main(String args[]) {  int six=06;  int seven=07;  int eight=010;  int nine=011;  System.out.println("Octal six = " + six);  System.out.println("Octal seven = " + seven);  System.out.println("Octal eight = " + eight);  System.out.println("Octal nine = " + nine);  }  } |

### Percobaan 6 : Menampilkan bilangan heksadesimal ke format desimal

|  |
| --- |
| public class Hexadecimal {  public static void main(String args[]) {  int x=0x0001;  int y=0x7fffffff;  int z=0xDeadCafe;  System.out.println("x = " + x);  System.out.println("y = " + y);  System.out.println("z = " + z);  }  } |

### Percobaan 7 : Mengamati perubahan nilai pada suatu tipe

|  |
| --- |
| public class Plus {  public static void main(String args[]) {  int x;  int y;  x=2147483647; //(2^31)-1  System.out.println("x = " + x);  y=x+1;  System.out.println("y = " + y);  }  } |

### Percobaan 8 : Memahami pemakaian 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("a: " + b);  System.out.println("a: " + c);  System.out.println("kata: " + kata);  }  } |

**Percobaan 9 : Melakukan increment dan decrement nilai**

|  |
| --- |
| class IncDec {  public static void main (String args[]) {  int x = 8, y = 13;  System.out.println("x = " + x);  System.out.println("y = " + y);  System.out.println("x = " + ++x);  System.out.println("y = " + y++);  System.out.println("x = " + x--);  System.out.println("y = " + --y);  }  } |

**Percobaan 10 : Melakukan operasi bit**

|  |
| --- |
| class Bitwise {  public static void main (String args[]) {  int x = 5, y = 6;  System.out.println("x = " + x);  System.out.println("y = " + y);  System.out.println("x & y = " + (x & y)); System.out.println("x | y = " + (x | y)); System.out.println("x ^ y = " + (x ^ y));  }  } |

**Percobaan 11 : Melakukan operasi komplemen**

|  |
| --- |
| class BitwiseComplement {  public static void main (String args[]) {  int x = 8;  System.out.println("x = " + x);  int y = ~x;  System.out.println("y = " + y);  }  } |

**Percobaan 12 : Melakukan operasi shift**

|  |
| --- |
| class Shift {  public static void main (String args[]) {  int x = 7;  System.out.println("x = " + x);  System.out.println("x >> 2 = " + (x >> 2)); System.out.println("x << 1 = " + (x << 1)); System.out.println("x >>> 1 = " + (x >>> 1));  }  } |

**Percobaan 13 : Menggunakan logical operator**

|  |
| --- |
| class LogicalOperator {  public static void main (String args[]) {  int x = 7, y = 11, z = 11;  System.out.println("x = " + x);  System.out.println("y = " + y);  System.out.println("z = " + z);  System.out.println("x < y = " + (x < y)); System.out.println("x > z = " + (x > z)); System.out.println("y <= z = " + (y <= z)); System.out.println("x >= y = " + (x >= y)); System.out.println("y == z = " + (y == z)); System.out.println("x != y = " + (x != z));  }  } |

**Percobaan 14 : Menggunakan operator boolean and**

|  |
| --- |
| public class BooleanAnd {  public static void main(String args[]) {  int a=5, b=7;  if ((a<2) & (b++<10)) b+=2;  System.out.println(b);  }  } |

**Percobaan 15 : Menggunakan operator boolean and short-circuit**

|  |
| --- |
| public class ShortCircuitBooleanAnd {  public static void main(String args[]) {  int a=5, b=7;  if ((a<2) && (b++<10)) b+=2; System.out.println(b);  }  } |

**Percobaan 16 : Menggunakan boolean or**

|  |
| --- |
| public class BooleanOr {  public static void main(String args[]) {  int a=5, b=7;  if ((a>2) | (b++<10)) b+=2;  System.out.println(b);  }  } |

**Percobaan 17 : Menggunakan boolean or short-circuit**

|  |
| --- |
| public class ShortCircuitBooleanOr {  public static void main(String args[]) {  int a=5, b=7;  if ((a>2) || (b++<10)) b+=2;  System.out.println(b);  }  } |

**Percobaan 18 : Menggunakan operator kondisi**

|  |
| --- |
| class Conditional {  public static void main (String args[]) {  int x = 0;  boolean isEven = false;  System.out.println("x = " + x);  x = isEven ? 4 : 7;  System.out.println("x = " + x);  }  } |

## Latihan

### Latihan 1 : Membuat formulasi proses casting dari tipe-tipe primitif

Lakukan percobaan casting antar tipe-tipe primitif. Lalu amati dan catatlah konversi dari suatu tipe ke tipe lainnya yang memerlukan suatu casting. Dari hasil analisa anda, buatlah suatu skema casting diantara tipe-tipe primitif.

### Latihan 2 : Membuat formulasi proses promotion dari tipe-tipe primitif

Lakukan percobaan promotion antar tipe-tipe primitif. Lalu amati dan catatlah konversi dari suatu tipe ke tipe lainnya yang menyebabkan terjadinya suatu promotion. Dari hasil analisa anda, buatlah suatu skema promotion diantara tipe-tipe primitif.

**Latihan 3 : Menampilkan representasi biner dari bilangan desimal bertipe byte**

Buatlah suatu program seperti diatas untuk menampilkan susunan bit dari suatu bilangan desimal, akan tetapi nilai bilangan input yang dimasukkan adalah bertipe byte

## Tugas

### Tugas 1 : Menganalisa batasan maksimum dari suatu tipe

Amatilah dan tulislah program berikut ini:

|  |
| --- |
| 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.

**Latihan2 : Menampilkan representasi biner dari bilangan desimal bertipe int**

Buatlah suatu program untuk menampilkan susunan bit dari suatu bilangan desimal. Nilai bilangan input yang dimasukkan adalah bertipe int.

Contoh tampilan:

Masukkan nilai desimal: 13

Susunan bit dari 13 adalah 00000000000000000000000000001101

Masukkan nilai desimal: 612

Susunan bit dari 13 adalah 00000000000000000000001001100100

Masukkan nilai desimal: -1

Susunan bit dari 13 adalah 11111111111111111111111111111111

Masukkan nilai desimal: -13

Susunan bit dari 13 adalah 11111111111111111111111111110011

### Tugas 3 : Mencari panjang menit dari durasi waktu

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

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

**Tugas 4 : Mencari representasi biner dari suatu bilangan**

Tuliskan representasi bit dari nilai –19? Jelaskan.