

LAPORAN PRAKTIKUM

Week 1

Diajukan untuk memenuhi salah satu tugas praktikum Mata kuliah Pemrograman Berorientasi Objek



Disusun Oleh:
Fawwaz Naufal Anwar (241511011)
Jurusan Teknik Komputer dan Informatika

Program Studi D-3 Teknik Informatika
Politeknik Negeri Bandung
2025

Pembahasan

Part 1

1. Menginstall Java Development Kit versi 17
2. Welcome Core Java!
3. Source Code:

```
public class Soal3{  
    public static void main (String[] args){  
        byte angka1 = 125;  
        byte angka2 = 6;  
        byte hasil = (byte) (angka1 + angka2);  
  
        System.out.println("angka 1 =" + hasil);  
    }  
}
```

Output:

```
PS D:\POLBAN\Semester 3\PB0\P\W1> java Soal3  
angka 1 =-125
```

Persoalan: Kenapa Outputnya -125? Karena variabel yang bernama hasil bertipe data numerik byte yang mana tipe data tersebut hanya menampung 1 byte dari -128 s.d 127. Yang mana, jika hasil telah melebihi 127, maka angka akan melanjutkan ke -128 dan seterusnya.

Referensi:

https://www.w3schools.com/java/java_data_types.asp

https://www.w3schools.com/java/java_type_casting.asp

Part 2

- Data Types
 - a. Program:
 1. Source Code

[illegible]

b. Permasalahan yang dihadapi

Permasalahan yang saya hadapi dan sempat bingung untuk ditangani adalah bagaimana saya menangani input yang melebihi dari tipedata 'long'.

c. Solusi dari permasalahan yang dihadapi

Saya melakukan beberapa research terhadap buku yang saya punya dan juga website w3schools untuk mencari solusi. Saya menemukan beberapa solusi yang dapat ditangani. Pertama, menggunakan tipe data `BigInteger`. Namun, menggunakan tipe data tersebut terlalu besar dialokasi dan bukan bagian dari tipe data primitive, melainkan sebuah class. Kedua, ini merupakan solusi yang saya ambil. Metode yang saya gunakan adalah menggunakan *Exception Handling* (try dan catch) dan input awal menggunakan tipe `String` yang kemudian di parsing menjadi `Long`. Hal ini membuat input yang melebihi dari batas maksimal long bisa diterima karena input bukan diterima sebagai bilangan bulat melainkan string. Kemudian, hasil input yang disimpan di variable bertipe string diparsing menjadi long. Ketika hasil yang diparsing long melebihi batas long, maka kode yang tereksekusi adalah yang terdapat pada bagian catch dan menghasilkan pesan “Input can’t be fitted...”

Referensi:

https://www.w3schools.com/java/java_try_catch.asp

https://www.w3schools.com/java/java_data_types.asp

- d. Tidak ada teman yang membantu persoalan ini, hanya beberapa referensi dari internet dan buku milik pribadi.
- Variables
 - a. Program
 - 1. Source Code

```

1 public class Constants {
2     Run main | Debug main
3     public static void main(String[] args) {
4         final double CM_PER_INCH = 2.54;
5         double paperWidth = 8.5;
6         double paperHeight = 11;
7         System.out.println("Paper size in centimeters: " + paperWidth * CM_PER_INCH + " by " + paperHeight * CM_PER_INCH);
8     }
9 }

```

```

1 public class Constants2 {
2     Run main | Debug main
3     public static final double CM_PER_INCH = 2.54;
4     public static void main(String[] args) {
5         double paperWidth = 8.5;
6         double paperHeight = 11;
7         System.out.println("Paper size in centimeters: " + paperWidth * CM_PER_INCH + " by " + paperHeight * CM_PER_INCH);
8     }
9 }

```

2. Output

```

PS D:\POLBAN\Semester 3\PBO\P\W1\part2-soal2> java Constants
Paper size in centimeters: 21.59 by 27.94
PS D:\POLBAN\Semester 3\PBO\P\W1\part2-soal2> java Constants2
Paper size in centimeters: 21.59 by 27.94
PS D:\POLBAN\Semester 3\PBO\P\W1\part2-soal2>

```

- b. Screenshot setiap jawaban dari pertanyaan

Apa perbedaan final double dengan static final double?

final double dideklarasikan secara lokal di dalam sebuah fungsi atau method sehingga method lain yang satu class tidak dapat menggunakan konstanta tersebut. Berbeda dengan static final double, konstanta tersebut dideklarasikan diluar method dan berlaku untuk seluruh method/fungsi di satu kelas. Konstanta ini biasa disebut konstanta kelas (*class constant*).
- c. Permasalahan

Tidak ada karena saya sudah membaca bagian ini sebelumnya.
- d. Solusi dari permasalahan

Tidak ada.
- e. Tidak ada teman yang membantu persoalan ini hanya referensi dari buku milik pribadi.

- Operator

- a. Program Code:

```
public class FloatingPoint{
    public static void main(String[] args) {
        double x = 92.98;
        int nx = (int) Math.round(x);

        System.out.println(nx);
    }
}
```

- b. Persoalan:

1. Pada kasus berikut jelaskan nilai nx setelah digunakan Math.round(x);

```
PS D:\POLBAN\Semester 3\PB0\P\W1\part2-soal3> java FloatingPoint
93
```

93

2. Kenapa dibutuhkan cast (int) dalam penggunaan Math.round(x)?

Berdasarkan w3schools, method Math.round() memiliki 2 nilai return yaitu 'long' ketika parameternya 'double' dan 'int' ketika parameternya 'float'. Program di atas menggunakan double sebagai parameter dari method round(), sehingga perlu melakukan type casting dari long menjadi int.

- c. Permasalahan

Tidak ada kendala ketika saya mengerjakan ini.

- d. Solusi dari permasalahan

Tidak ada

- e. Tidak ada teman yang membantu, saya menggunakan referensi dari website bernama w3schools.

- Operator (1)

- a. Program

1. Source Code:

```

class ConvertDataType
{
    static short methodOne(long l)
    {
        int i = (int) l;
        return (short)i;
    }
    Run main | Debug main
    public static void main(String[] args)
    {
        double d = 10.25;
        float f = (float) d;
        byte b = (byte) methodOne((long) f);
        System.out.println(b);
    }
}

```

2. Output:

```

PS D:\POLBAN\Semester 3\PB0\P\W1\part2-soal4> java ConvertDataType
10

```

b. Persoalan

1. Jelaskan output dari variable b!

Variable b diassign oleh sebuah fungsi dengan nilai return short bernama methodOne(long l) yang dicasting menjadi byte.

2. Jelaskan apa yang berubah dari variable d menjadi variable b setelah dilakukan cast ?

Perubahan variable d menjadi b memiliki pola yang menarik, dimana pada type-casting terdapat 2 jenis yaitu *Widening Casting* (dari yang lebih kecil menuju yang besar) dan *Narrowing Casting* (besar ke kecil). Nah, dari source code tersebut jika kita tracing kemana pola berjalannya, maka akan sebagai berikut double -> float -> long -> int -> short -> byte. Hal tersebut merupakan tipe Narrowing Casting.

c. Permasalahan.

Tidak ada.

d. Jawaban dari Permasalahann

Tidak ada.

e. Tidak ada teman yang membantu dikarenakan saya dapat menjawabnya sendiri dan menggunakan buku, web sebagai referensi belajar.

- String

a. Program

1. Source Code:

```
import java.util.Scanner;

public class SoalString {
    Run main | Debug main
    public static void main (String[] args) {
        Scanner input = new Scanner(System.in);
        int iteration;

        String A = input.nextLine();
        int lenA = (int) A.length();

        String B = input.nextLine();
        int lenB = (int) B.length();

        int sumLen = lenA + lenB;
        System.out.println(sumLen);

        if (lenA < lenB) {
            iteration = lenA;
        } else if (lenA > lenB) {
            iteration = lenB;
        } else {
            iteration = lenA;
        }

        for (int i = 0; i < iteration; i++) {
            if (A.charAt(i) > B.charAt(i)) {
                System.out.println("yes");
                break;
            } else {
                System.out.println("No");
                break;
            }
        }

        String AUpper = A.substring(0, 1).toUpperCase() + A.substring(1);
        String BUpper = B.substring(0, 1).toUpperCase() + B.substring(1);
        System.out.println(AUpper+" "+BUpper);
    }
}
```

2. Data Test

A = Hello
B = Java

3. Output

```
PS D:\POLBAN\Semester 3\PBO\P\W1\part2-soal5> java SoalString
hello
java
9
No
Hello Java
PS D:\POLBAN\Semester 3\PBO\P\W1\part2-soal5> |
```

b. Permasalahan yang saya hadapi

Saya sempat kesulitan mencari method string yang cocok.

c. Solusi

Menemukan di referensi web lain seperti <https://www.includehelp.com/> untuk mengerjakan latihan ini. Seperti penggunaan str.charAt(int), dan str.substring.

d. Tidak ada teman yang membantu.

Referensi:

<https://www.includehelp.com/java/convert-a-character-array-to-the-string-in-java.aspx>

https://www.w3schools.com/java/java_ref_string.asp

Horstmann Cay S. - Core Java. Volume I - Fundamentals. 1-Pearson (2019)