

LAPORAN PRAKTIKUM PERTEMUAN-1

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

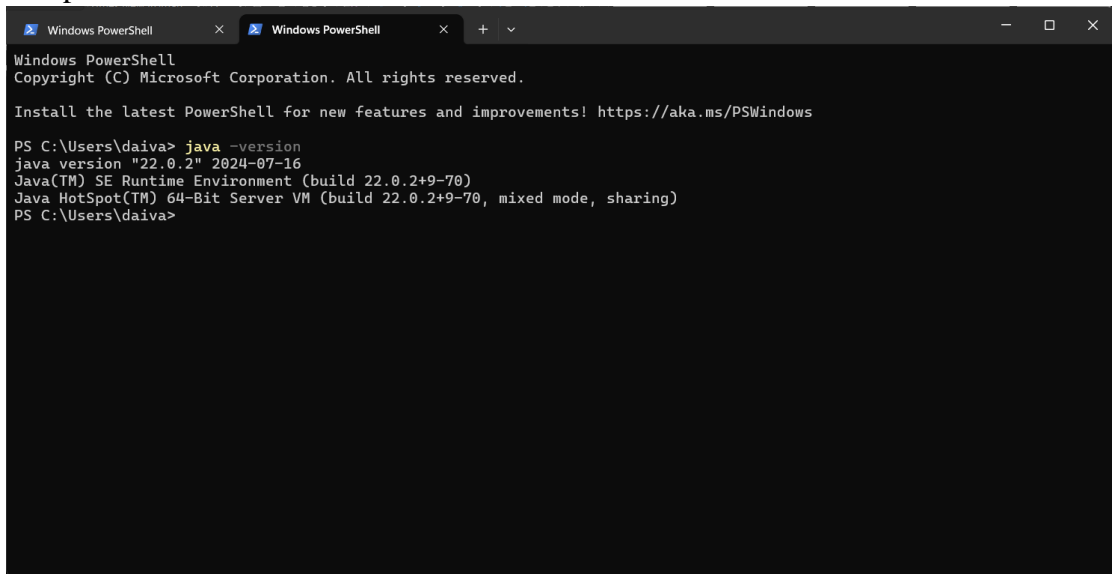


Disusun Oleh:
Daiva Raditya Pradipa (231511039)
Jurusan Teknik Komputer dan Informatika

Program Studi D-3 Teknik Informatika
Politeknik Negeri Bandung
2024

Soal Praktikum-1

1. Setup Software Environment



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\daiva> java -version
java version "22.0.2" 2024-07-16
Java(TM) SE Runtime Environment (build 22.0.2+9-70)
Java HotSpot(TM) 64-Bit Server VM (build 22.0.2+9-70, mixed mode, sharing)
PS C:\Users\daiva>
```

Penjelasan: Dalam melakukan setup environment JDK sendiri cukup mudah, kita hanya perlu masuk ke url [Java Downloads | Oracle](https://www.oracle.com/indonesia/java/technologies/javase-downloads/) dan medownload installer msi JDK version yang diinginkan. Setelah itu cukup melakukan instalasi sesuai panduan installer, dan terakhir cek apakah java telah terinstal dengan baik dengan perintah `java -version` pada CMD.

Permasalahan: Selama melakukan setup environment saya tidak menemukan permasalahan sama sekali

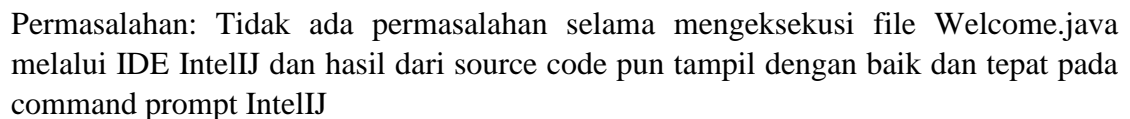
Solusi: -

Nama teman yang membantu: -

2. Using Notepad and Command Line Tool

Penamaan nama file harus selaras atau sesuai dengan nama class yang didefinisikan dalam source code

3. Using an Integrate Development Environment



Apabila kita menggunakan IDE seperti IntelliJ proses compile dan running akan dieksekusi secara langsung dengan menekan tombol button running atau debungging pada IDE yang digunakan. Berbeda dengan apabila kita mengeksekusi melalui command prompt, kita perlu melakukan proses compile terlebih dahulu menggunakan perintah `javac Welcome.java` dan proses run dengan perintah `java Welcome.java`

4. Soal 1

```
1 public class Soal1 {
2     public static void main(String[] args) {
3
4         byte angka2 = 6;
5         byte hasil = (byte) (angka1 + angka2);
6
7         System.out.println("hasil : "+hasil);
8     }
9 }
10
11 //Penjelasan
12 //Hasil dari source code : -125
13 //Alasan: Karena kemampuan kapasitas nilai yang dapat ditampung tipe data byte adalah -128 sampai 127. Sedangkan source code diatas mencoba untuk
14 //menjumlahkan variabel angka1 dengan tipe data byte dan nilai 125 ditambahkan dengan variabel angka2 dengan nilai 6 dan hasilnya disimpan variabel
15 //dengan tipe data byte yang tadi telah dijelaskan hanya dapat menampung nilai mulai dari -128 sampai dengan 127 sedangkan hasil penjumlahan var
16 //otomatis disini terjadi kondisi overflow.
17 //Mengepa hasilnya -125?
18 //Secara matematis byte terdiri dari 8 bit. Variabel angka1 dengan nilai 125 memiliki nilai bit 01111101 dan variabel angka2 memiliki nilai bit
19 //10000011. Nilai 10000011 ini perlu diinterpretasikan dalam konteks dua's complement sebagai bilangan negatif.
```

Jawaban soal yang dipertanyakan:

Hasil dari source code: -125.

Alasan: Kapasitas nilai yang dapat ditampung oleh tipe data byte adalah dari -128 hingga 127. Source code di atas mencoba menjumlahkan variabel angka1 (dengan nilai 125 dan tipe data byte) dengan variabel angka2 (dengan nilai 6). Hasil penjumlahan ini disimpan dalam variabel hasil yang juga bertipe byte. Karena hasil penjumlahan angka1 dan angka2 adalah 131, yang berada di luar rentang tipe byte, maka terjadi overflow.

Mengapa hasilnya -125?

Secara matematis, byte terdiri dari 8 bit. Variabel angka1 dengan nilai 125 memiliki representasi biner 01111101, dan variabel angka2 dengan nilai 6 memiliki representasi biner 00000110. Ketika dijumlahkan, hasilnya adalah 10000011.

Nilai 10000011 ini diinterpretasikan dalam konteks two's complement sebagai bilangan negatif karena bit paling kiri adalah 1.

Untuk menemukan angka negatifnya:

- Balik semua bit: 01111100
- Tambahkan 1: 01111101 (yang dalam desimal adalah 125).

Jadi, 10000011 mewakili -125.

Permasalahan: Mengapa hasilnya -125? dan bukan 131

Solusi: Kapasitas nilai yang dapat ditampung oleh tipe data byte adalah dari -128 hingga 127.

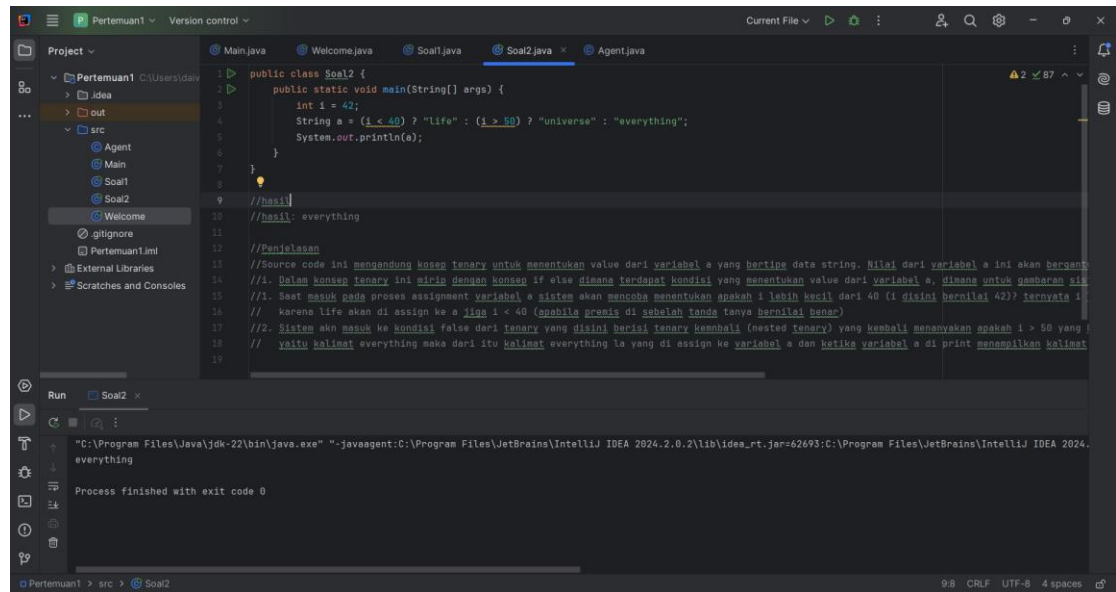
Leason learn:

Dalam menentukan tipe data yang akan digunakan suatu variable, kita perlu tahu tujuan dari variable itu dibuat dan kegunaan dari variable tersebut. Hal ini untuk memastikan kita memilih tipe data sesuai dengan kebutuhan dan untuk memastikan tipe data yang

kita pilih menggunakan memori seminim mungkin untuk meningkatkan performa aplikasi dan memenuhi kebutuhan

Teman yang membantu:
Tresnardi Fathu Rhamdan
M. Ilham

5. Soal 2



```
1 public class Soal2 {
2     public static void main(String[] args) {
3         int i = 42;
4         String a = (i < 40) ? "life" : (i > 50) ? "universe" : "everything";
5         System.out.println(a);
6     }
7 }
8
9 //hasil
10 //hasil: everything
11
12 //Penjelasan
13 //Source code ini mengandung konsep ternary untuk menentukan value dari variabel a yang bertipe data string. Nilai dari variabel a ini akan bergantung
14 //1. Dalam konsep ternary ini mirip dengan konsep if else dimana terdapat kondisi yang menentukan value dari variabel a, dimana untuk gambaran s.d
15 //1. Saat masuk pada proses assignment variabel a sistem akan mencoba menentukan apakah i lebih kecil dari 40 (i disini bernilai 42)? ternyata i
16 // karena life akan di assign ke a jika i < 40 (apabila premis di sebelah tanda tanya bernilai benar)
17 //2. Sistem akan masuk ke kondisi false dari ternary yang disini berisi ternary kembali (nested ternary) yang kembali menanyakan apakah i > 50 yang
18 // yaitu kalimat everything maka dari itu kalimat everything la yang di assign ke variabel a dan ketika variabel a di print menampilkan kalimat
```

Jawaban soal yang dipertanyakan:

Hasil: everything

Penjelasan:

Source code ini menggunakan konsep **ternary operator** untuk menentukan nilai dari variabel a yang bertipe data String. Nilai dari variabel a bergantung pada nilai i. Konsep ternary ini mirip dengan konsep if-else, di mana terdapat kondisi yang menentukan nilai variabel a. Berikut adalah penjelasan langkah-langkah eksekusinya:

1. Proses Assignment:

Sistem akan memeriksa apakah i lebih kecil dari 40 (dalam hal ini, i bernilai 42). Karena i lebih besar dari 40, maka nilai life tidak akan diassign ke variabel a. Life akan diassign ke a jika i < 40 (apabila premis sebelum tanda tanya bernilai benar).

2. Kondisi False:

Karena i tidak lebih kecil dari 40, sistem melanjutkan ke kondisi false dari ternary pertama, yang berisi ternary operator kedua (nested ternary). Pada ternary kedua, sistem memeriksa apakah i lebih besar dari 50. Karena i adalah 42, yang tidak lebih besar dari 50, maka kondisi false dari ternary kedua akan dipilih. Nilai everything diassign ke variabel a dalam kasus ini.

Akhirnya, ketika variabel a diprint, hasil yang ditampilkan adalah everything.

Permasalahan: -

Solusi: -

Teman yang membantu: -

Link github source code week-1: [Tugas-PBO/Wekk-1 at main · RaditZX/Tugas-PBO \(github.com\)](https://github.com/RaditZX/Tugas-PBO)