**Praktikum Pertemuan 1**

*Disusun untuk memenuhi tugas mata kuliah Pemrograman Berorientasi Objek (Praktek)*



Disusun Oleh:

Dwika Ali Ramdhan (231511042) 2B – D3

Jurusan Teknik Komputer dan Informatika Politeknik Negeri Bandung

2024

# DAFTAR ISI

[DAFTAR ISI 1](#_Toc175251249)

[DAFTAR GAMBAR 2](#_Toc175251250)

[PEMBAHASAN 1](#_Toc175251251)

[1.1. Setup Software Engine 1](#_Toc175251252)

[1.2. Using Notepad & Command Line Tools 1](#_Toc175251253)

[1.3. Using an Integrate Development Environment 1](#_Toc175251254)

[1.4. Soal Analisis 1 2](#_Toc175251255)

[1.5. Soal Analisis 2 3](#_Toc175251256)

[REFERENSI 5](#_Toc175251257)

[LAMPIRAN 6](#_Toc175251258)

# DAFTAR GAMBAR

[Gambar 1 Cek Versi Java 1](#_bookmark5)

[Gambar 2 Program Welcome.Java 1](#_bookmark6)

[Gambar 3 Output Welcome.java 1](#_bookmark7)

[Gambar 4 Program Welcome.java Menggunakan Intellij 2](#_bookmark9)

[Gambar 5 Output Welcome.java di Intellij 2](#_bookmark10)

[Gambar 6 Script Soal No 1 3](#_bookmark12)

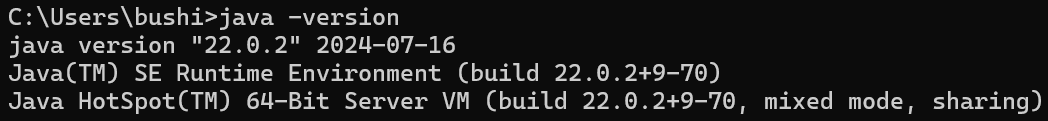
[Gambar 7 Output Soal No 1 3](#_bookmark13)

[Gambar 8 Script Soal No 2 4](#_bookmark15)

[Gambar 9 Output Soal No 2 4](#_bookmark16)

# PEMBAHASAN

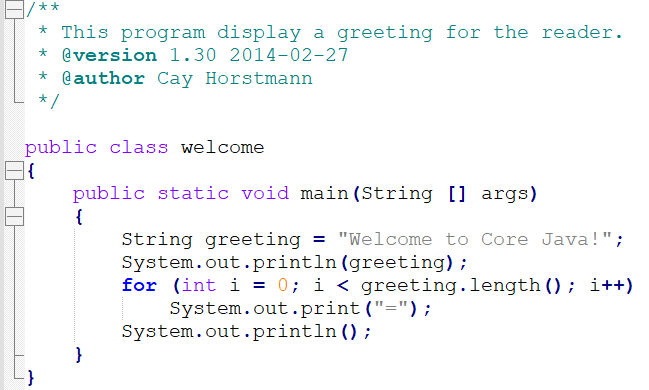
## Setup Software Engine



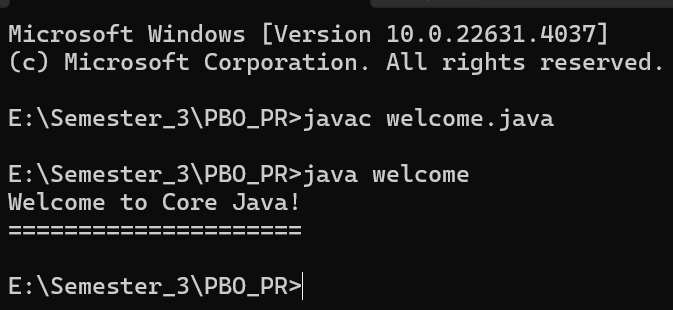
Gambar 1 Cek Versi Java

* 1. Using Notepad & Command Line Tools

1. Welcome.java Program



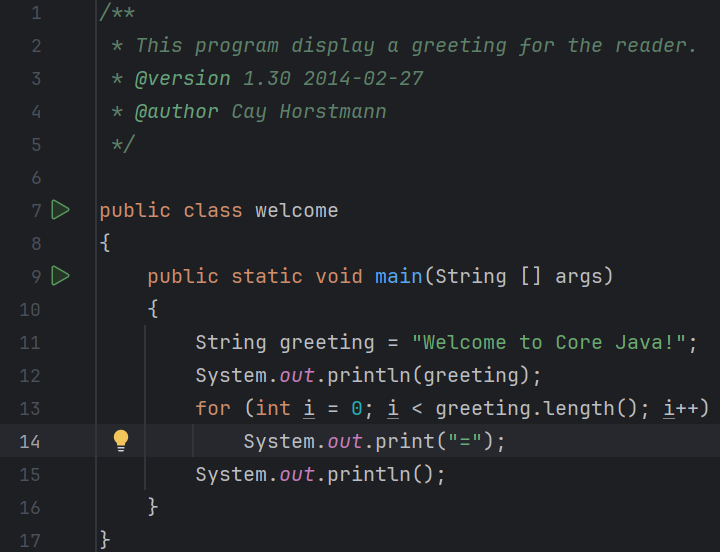
*Gambar 2 Program Welcome.Java*

1. Welcome.java Output

Gambar 3 Output Welcome.java

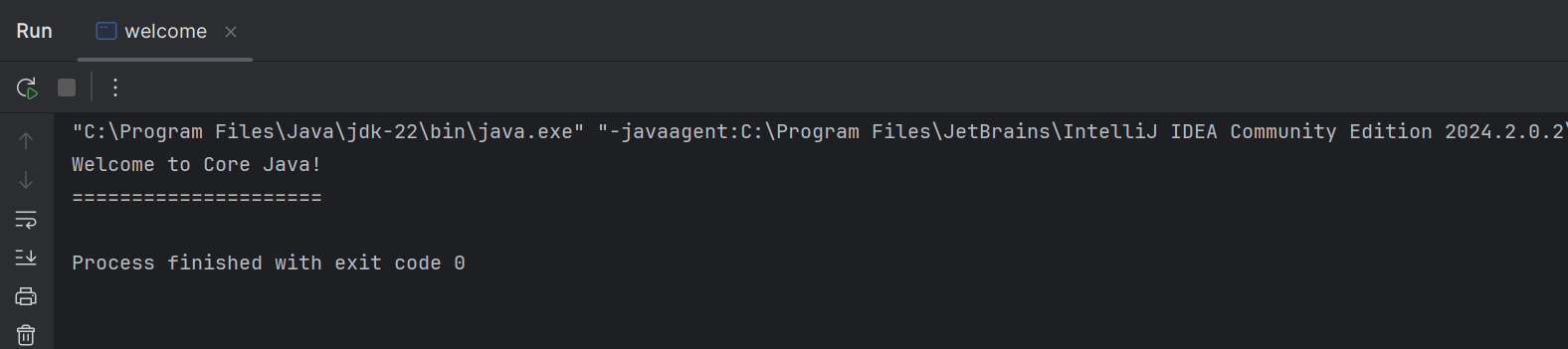
## Using an Integrate Development Environment

1. Program Welcome.java di Intellij



Gambar 4 Program Welcome.java Menggunakan Intellij

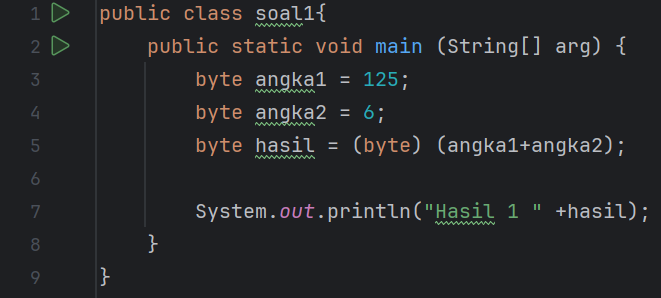
1. Output



Gambar 5 Output Welcome.java di Intellij

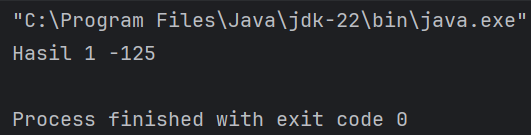
## Soal Analisis 1

1. Program Soal



Gambar 6 Program Soal1

1. Output Soal2



Gambar 7 Output Soal1

1. Penjelasan

Terdapat nilai untuk angka1 dengan tipe data *byte* yaitu 125 dan angka2 dengan tipe data *byte* yaitu 6. Hasil merupakan operasi aritamtika dari angka1 + angka2 yang menghasilkan nilai outputnya yaitu -125. Mengapa hasil outputnya -125 bukankah 125 + 6 seharunya menghasilkan nilai 131? Ya, benar tetapi dalam hal ini angka yang dimasukkan tersebut bertipe *byte* .

Tipe data *byte* di Java adalah tipe data primitif yang digunakan untuk menyimpan nilai numerik kecil. *Byte* adalah tipe data 8-bit, yang berarti setiap variabel *byte* di Java menggunakan 8 bit (1 *byte*) dari memori. Rentang nilai *byte* adalah dari -128 hingga 127 yang jika ditotal menjadi 256.

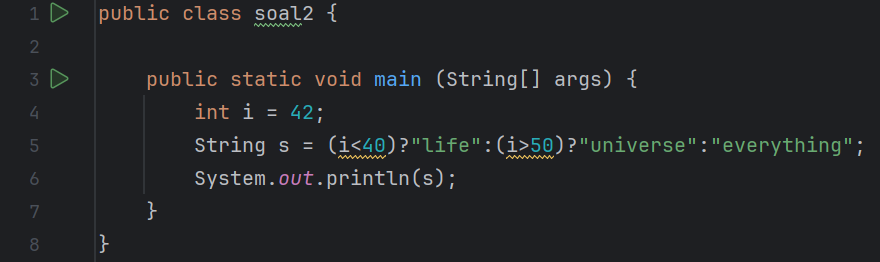
Jadi, karena nilai nya menghasilkan 131 sedangkan batas pada tipe data byte sendiri untuk positif sampai 127 maka akan terjadi overflow dimana perlu dikonversi kembali ke tipe data *byte* yaitu dengan mengurai 131 dengan 256 yang merupakan total tipe data *byte* yang menghasilkan nilai -125.

1. Kesulitan dan Solusi

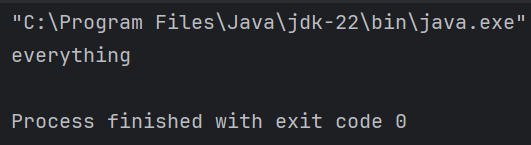
Saya menghadapi kesulitan dalam memahami bagaimana hasil output program dapat menghasilkan -125. Saya menemukan bagaimana proses overflow dilakukan dari w3schools.com dan chatGPT.com

## Soal Analisis 2

1. Program Soal2



Gambar 8 Program Soal2

1. Output Soal2

Gambar 9 Output Soal2

1. Penjelasan

Terdapat variabel i dengan tipe data integer bernilai 42 dan variabel stirng s yang akan diisi berdasarkan hasil dari operasi tenary , yang ternyata berdasarkan operasi tenary dengan kondisi i <40 dan i >50 dan i bernial 42 menghasilkan kata everything mengapa demikian? .

*variable = (condition) ? expressionTrue : expressionFalse;*

Operasi ternary di Java adalah operator yang digunakan sebagai alternatif singkat untuk pernyataan if-else yang sederhana. Operator ini membantu menentukan satu dari dua nilai berdasarkan suatu kondisi. Ini adalah satu-satunya operator kondisional di Java yang mengambil tiga operan, sehingga disebut "ternary."

Pertama, kondisi (i < 40) diperiksa. Karena i bernilai 42, kondisi (i < 40) adalah false. Karena kondisi pertama false, operator ternary beralih ke bagian else-nya, yang merupakan ekspresi ternary kedua.

*(i > 50) ? "universe" : "everything"*

Selanjutnya, kondisi (i > 50) diperiksa. Karena i adalah 42, kondisi (i > 50) juga false. Karena kondisi kedua juga false, bagian else dari ekspresi kedua yang diambil, yaitu "everything".

1. Kesulitan dan Solusi

Sama seperti soal1 saya sulit memahami bagaimana cara kerja program ini bagaimana tanda tanya “?” dan tanda seru “!” dapat digunakan sebagai *if-else*. Saya menemukan hal ini dari ChatGPT, GeeksforGeeks, dan W3Schools. Mungkin ini lebih kepada lesson learn dari pada kesulitan.

# REFERENSI

https://www.w3schools.com/java/java\_data\_types.asp

https://www.w3schools.com/java/java\_conditions\_shorthand.asp

https://www.geeksforgeeks.org/java-ternary-operator-with-examples/

https://chatgpt.com/

# LAMPIRAN

LINK GITHUB: <https://github.com/mahesyasn18/PBO_Mahesya_221511054>