

PRAKTIKUM PERTEMUAN 1

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



Disusun oleh:

Bandyaga Adiansyah Sugandi

NIM 231511037

2B – D3

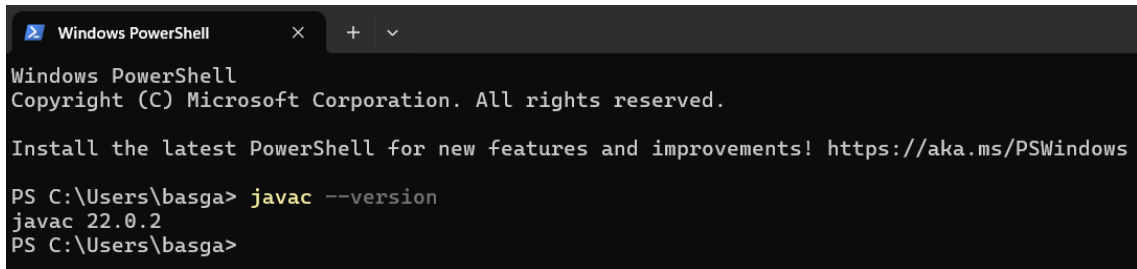
Jurusan Teknik Komputer dan Informatika

Politeknik Negeri Bandung

2024

PEMROGRAMAN BERORIENTASI OBJEK (PBO) – PRAKTIKUM 1

1. Setup Software Engine



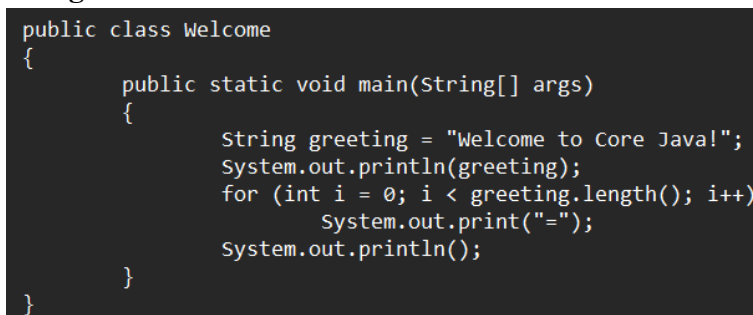
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

PS C:\Users\basga> javac --version
javac 22.0.2
PS C:\Users\basga>
```

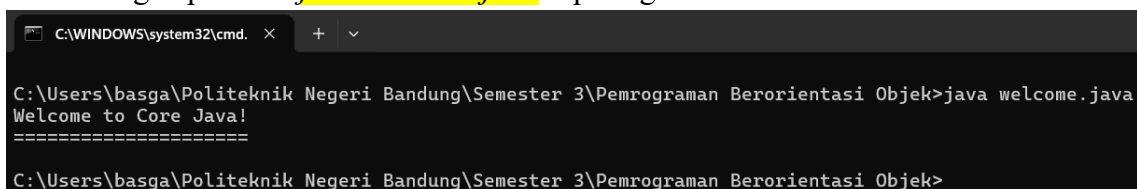
Proses instalasi Java Development Kit (JDK) untuk versi JDK 22 kemudian memeriksa apakah setup JDK tersebut telah terinstal dengan benar menggunakan CMD dengan command `javac -version` atau `java -version`

2. Using Command Line Tools



```
public class Welcome
{
    public static void main(String[] args)
    {
        String greeting = "Welcome to Core Java!";
        System.out.println(greeting);
        for (int i = 0; i < greeting.length(); i++)
            System.out.print("=");
        System.out.println();
    }
}
```

Pada persoalan ini, membuat script code welcome menggunakan bahasa pemrograman Java di Notepad kemudian dijalankan atau di-execute menggunakan CMD dengan perintah `java welcome.java` seperti gambar di bawah ini

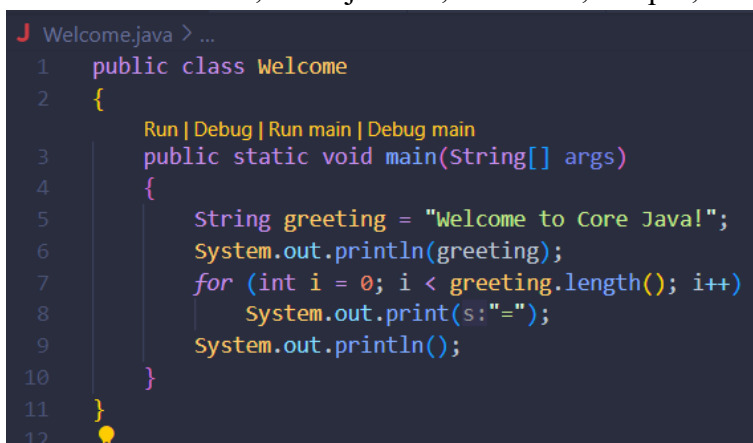


```
C:\WINDOWS\system32\cmd. x + v

C:\Users\basga\Politeknik Negeri Bandung\Semester 3\Pemrograman Berorientasi Objek>java welcome.java
Welcome to Core Java!
=====
C:\Users\basga\Politeknik Negeri Bandung\Semester 3\Pemrograman Berorientasi Objek>
```

3. Using an IDE

Membuat script code atau program dengan Java menggunakan IDE seperti Visual Studio Code, IntelliJ IDEA, NetBeans, Eclipse, dan sebagainya.



```
Welcome.java > ...
1  public class Welcome
2  {
   Run | Debug | Run main | Debug main
3      public static void main(String[] args)
4      {
5          String greeting = "Welcome to Core Java!";
6          System.out.println(greeting);
7          for (int i = 0; i < greeting.length(); i++)
8              System.out.print(s:"=");
9          System.out.println();
10     }
11 }
12
```

4. Soal Analisis 1

```
Soal1.java > ...
1 public class Soal1 {
    Run | Debug | Run main | Debug main
2     public static void main(String[] args) {
3         byte angka1 = 125;
4         byte angka2 = 6;
5         byte hasil = (byte) (angka1 + angka2);
6
7         System.out.println("Hasil 1 = " + hasil);
8     }
9 }
10
```

```
basga@Aga MINGW64 ~/Politeknik Negeri Bandung/Semester 3/Pemrograman Berorientasi Objek
$ java Soal1.java
Hasil 1 = -125
```

Permasalahan yang dihadapi:

⇒ Mengapa hasil atau output pada program ketika dijalankan adalah -125?

Solusi dari permasalahan yang dihadapi:

- ⇒ Tipe data **byte** adalah tipe data pada bahasa Java yang dapat menampung angka dari **-128 hingga 127** (https://www.w3schools.com/java/java_data_types.asp)
- ⇒ Hasil penjumlahan sebenarnya dari **angka1** dan **angka2** yaitu adalah **125 + 6 = 131**
- ⇒ Nilai 131 melebihi batas dari tipe data **byte** yaitu 127 maka akan terjadi *overflow* sehingga Java akan melakukan proses yang memaksa nilai 131 ini dapat ditampung pada tipe data **byte** dengan cara mengembalikannya ke nilai minimum yaitu -128 ditambah dengan sisa penambahannya yaitu 3 menjadi -125
- ⇒ Penjabaran:
- $125 + 6 = 131$ (hasil penjumlahan sebenarnya)
 - $125 + 2 = 127$ (batas overflow)
 - $127 + 1 = -128$ (kembali ke underflow agar tetap tertampung pada tipe data **byte**)
 - $-128 + 3 = -125$ (output akhir)

Nama teman yang membantu memecahkan masalah:

- Nur Akmal
- Muhammad Ilham F.

5. Soal Analisis 2

```
Soal2.java > Language Support for Java(TM) by Red Hat > Soal2 > main(String[])
1 public class Soal2 {
    Run | Debug | Run main | Debug main
2     public static void main(String[] args) {
3         int i = 42;
4         String s = (i < 40) ? "life " : (i > 50) ? "universe" : "everything";
5         System.out.println(s);
6     }
7 }
8
```

```
basga@Aga MINGW64 ~/Politeknik Negeri Bandung/Semester 3/Pemrograman Berorientasi Objek
$ java Soal2.java
everything
```

Permasalahan yang dihadapi:

⇒ Mengapa hasil atau output pada program ketika dijalankan adalah *everything*?

Solusi dari permasalahan yang dihadapi:

- ⇒ Pada program ini, menggunakan sebuah metode *nested condition (ternary) operator* di mana sebuah condition ada tiga operands yaitu *question mark (?)*, lalu output conditonal *true* yang diikuti dengan *colon / titik dua (:)*, dan output conditional *false*. Ini merupakan sebuah metode alternatif dari bentuk umum *if...else* (https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Conditional_operator)
- ⇒ Pada program diatas, variabel i bernilai 42, maka kondisi $(1 < 40)$ dan kondisi $(1 > 50)$ tidak akan tereksekusi karena tidak memenuhi kondisinya. Maka dari itu kondisi diluar itulah yang akan dieksekusi yaitu adalah *“everything”*.

Nama teman yang membantu memecahkan masalah:

- (tidak ada)