

## **MODUL 2**

### **PENGENALAN JAVA**



#### **2.1 Sejarah Java**

- Java adalah bahasa pemrograman yang dibuat oleh James Gosling saat bekerja di Sun Microsystems
- Java dirilis ke public tahun 1995
- Java adalah bahasa pemrograman berorientasi objek dan mendukung pengelolaan memori secara otomatis
- Java terkenal dengan write once, run anywhere, karena binary program Java di-generate secara independen dan bisa dijalankan di Java Virtual Machine yang terinstall di berbagai sistem operasi

#### **2.2 Sejarah Penamaan Java**

- “GreenTalk” (1991): Dikembangkan oleh James Gosling dan tim Green Team di Sun Microsystems.
- “Oak”: Nama “GreenTalk” diganti menjadi Oak, terinspirasi dari pohon ek di depan kantor Gosling.
- “Java” (1995): Java diusulkan oleh Gosling, terinspirasi dari kopi Java yang (berasal dari Pulau Jawa, Indonesia) sering dikonsumsi oleh tim pengembang saat itu.

#### **2.3 Dimana Java Banyak Digunakan?**

- Backend, banyak perusahaan besar saat ini menggunakan Java sebagai aplikasi backend nya seperti Twitter, Netflix, Spotify, Amazon, Alibaba, Bilibili, dan lain-lain
- Big Data, teknologi-teknologi big data yang saat ini populer, kebanyakan adalah teknologi Java, seperti Apache Hadoop, Elasticsearch, Apache Cassandra, Apache Spark, Apache Kafka, dan lain-lain
- Android, di Android kita bisa menggunakan Java dan Kotlin untuk membuat aplikasinya

#### **2.4 Keunggulan dan Kelemahan Java**

##### **A. Keunggulan Java**

- **Multiplatform:** Java dapat dijalankan di berbagai platform atau sistem operasi tanpa perlu disusun ulang, sesuai dengan slogan "Write Once, Run Anywhere".
- **Berorientasi Objek:** Java merupakan bahasa pemrograman yang berorientasi pada objek, yang memudahkan pengembangan aplikasi yang kompleks dan memungkinkan penggunaan kembali kode.
- **Memiliki Library yang Lengkap:** Java dikenal memiliki library yang lengkap, yang memudahkan pengembang dalam membangun berbagai jenis aplikasi.
- **Keamanan Tinggi:** Java dirancang dengan tingkat keamanan yang tinggi, sehingga cocok untuk aplikasi yang memerlukan proteksi data.

## **B. Kelemahan Java**

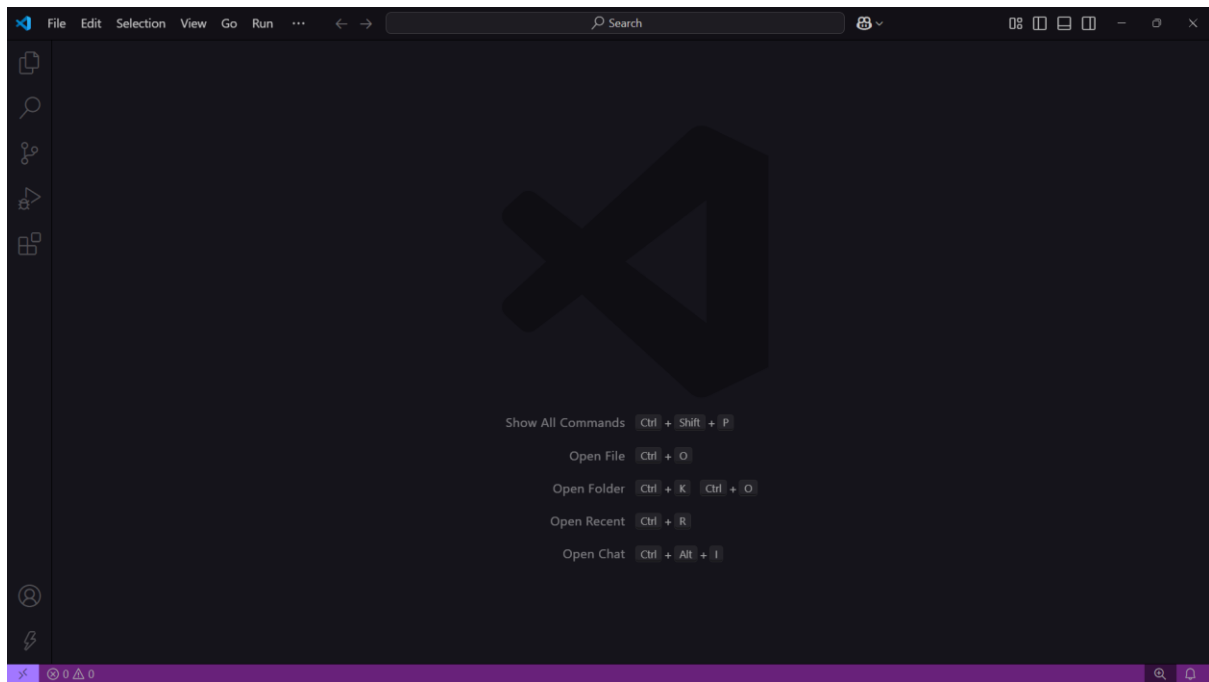
- **Penggunaan Memori yang Besar:** Java memerlukan memori yang cukup besar untuk menjalankan programnya, yang dapat mempengaruhi performa pada perangkat dengan sumber daya terbatas.
- **Mudah Didekompilasi:** Kode Java dapat dengan mudah didekompilasi kembali menjadi kode sumber, yang dapat menimbulkan risiko keamanan dan pelanggaran hak cipta.
- **Tampilan GUI Kurang Menarik:** Antarmuka pengguna grafis (GUI) yang dibuat dengan Java sering dianggap kurang menarik dibandingkan dengan bahasa pemrograman lain yang lebih fokus pada desain antarmuka.

## **2.5 Rekomendasi Belajar Tambahan**

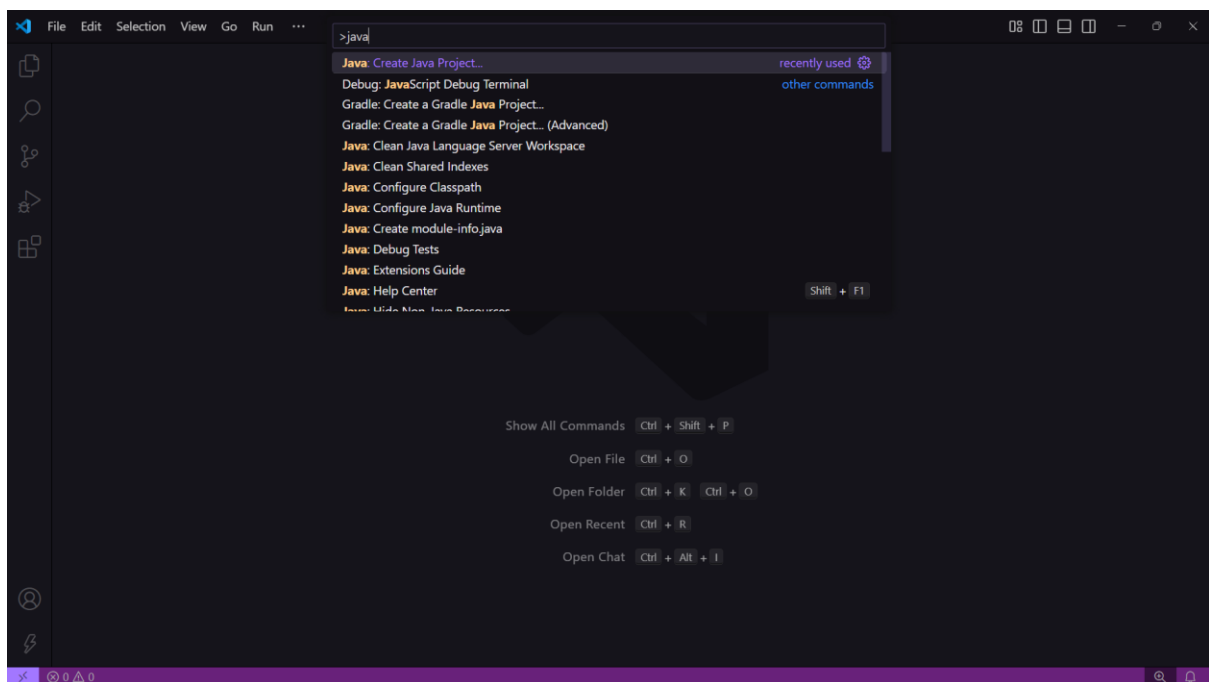
- Youtube: [programmer zaman now](https://www.youtube.com/channel/UC8butISFwT-97Ag87Pyp9kw)
- Tutorialspoint: [www.tutorialspoint.com/java/](https://www.tutorialspoint.com/java/)

## **2.6 Membuat Proyek Java di vscode**

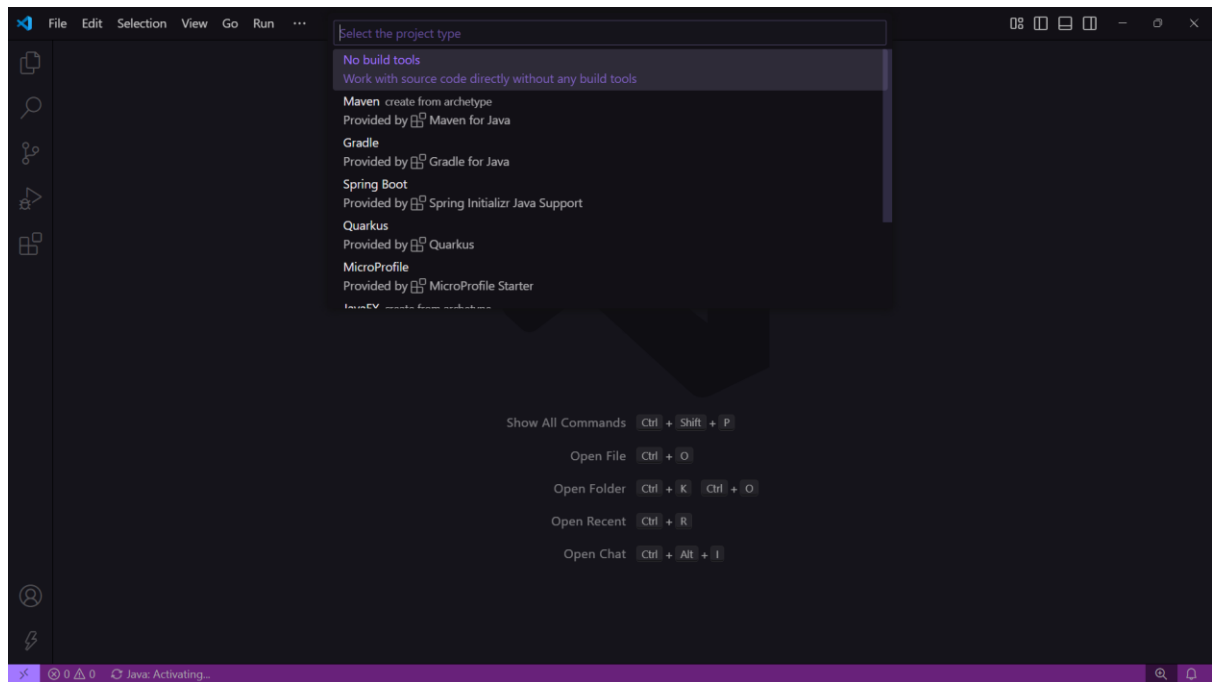
- Buka VS Code
- Buka Command Palette (Ctrl + Shift + P)
- Ketik "Java: Create Java Project" lalu pilih opsi tersebut.
- Pilih template "No Build Tools" untuk proyek sederhana (atau pilih Maven/Gradle jika diperlukan).
- Pilih folder tempat proyek akan disimpan.
- Beri nama proyek, misalnya: HelloJava.



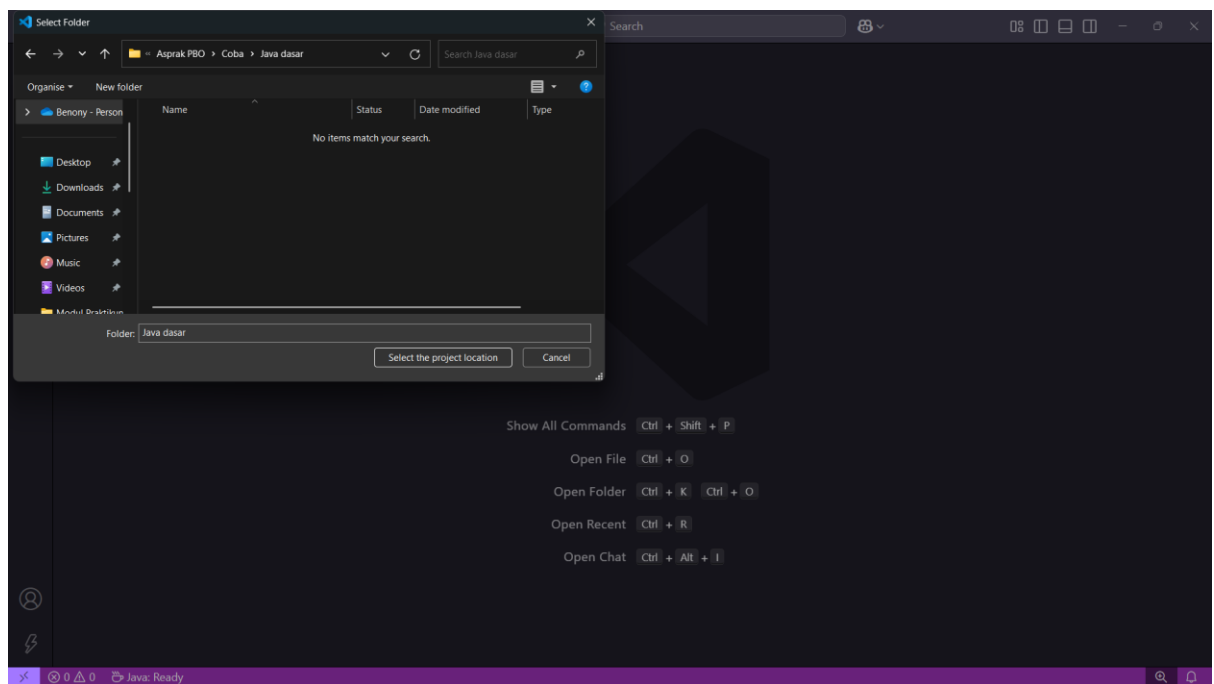
Langkah 1: Buka vscode



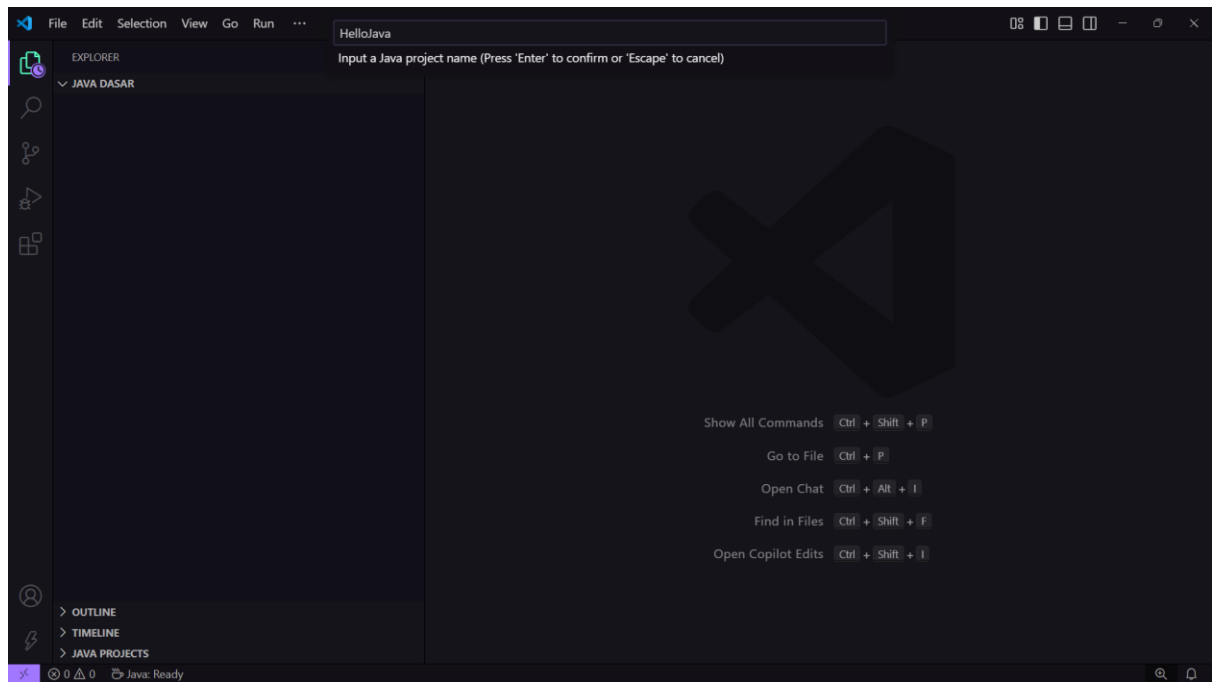
Langkah 2: Buka Command Palette (Ctrl + Shift + P) dan ketik “Java: Create Java Project”



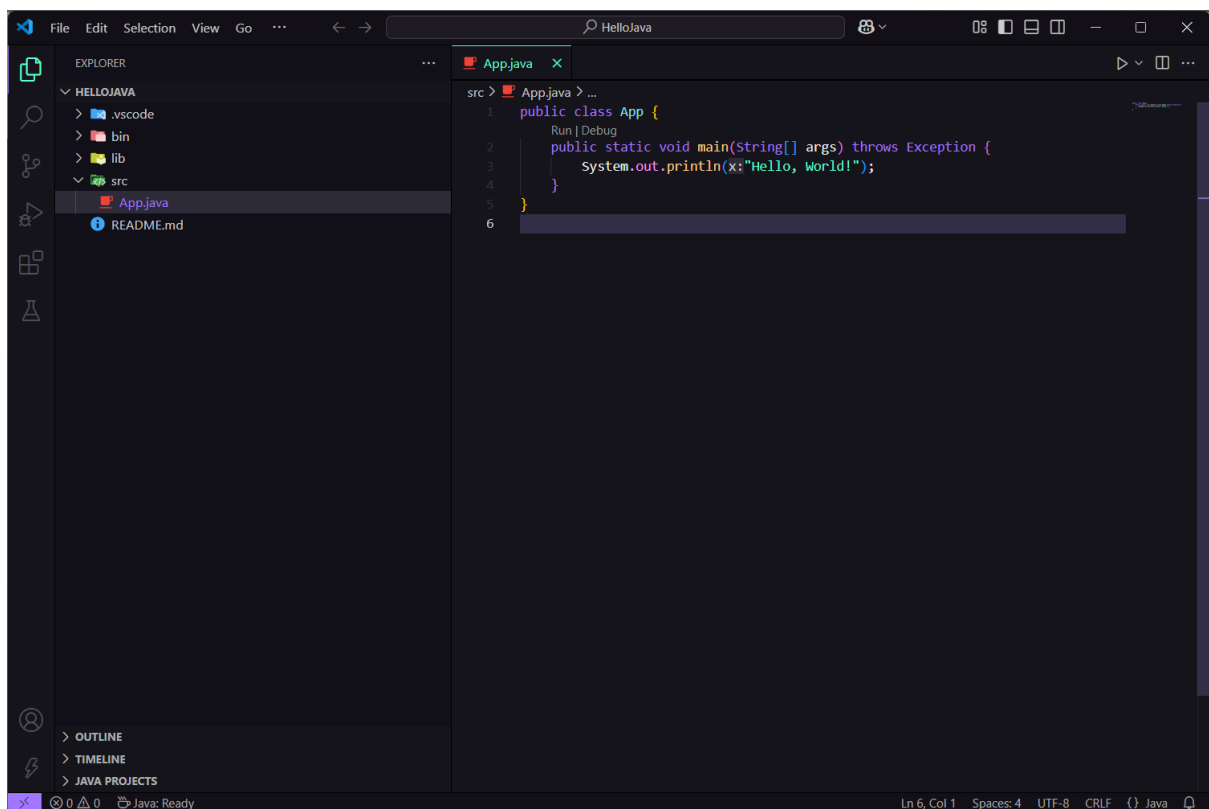
Langkah 3: Pilih template "No Build Tools" untuk membuat proyek sederhana



Langkah 4: Pilih lokasi tempat proyek akan disimpan



Langkah 5: Beri nama proyek, misalnya: HelloJava



Langkah 6: Proyek java yang sudah dibuat akan muncul seperti ini

## 2.7 Pemrograman Java Dasar

### a. Program Hello World

Program Java selalu dimulai dari *class* dan method *main()*

```
public class HelloJava {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

Penjelasan:

- `public class HelloJava` → Nama class harus sama dengan nama file (HelloJava.java).
- `public static void main(String[] args)` → Method utama tempat program berjalan.
- `System.out.println("Hello, World!");` → Mencetak teks ke layar

### b. Java Output

Berbagai cara menampilkan Output pada Java:

```
public class Output {  
    public static void main(String[] args) throws Exception {  
        // Java Output  
        // Print text to the console with new line  
        System.out.println("Hello, World!");  
        System.out.print("Saya belajar Java.");  
        System.out.print("Java mudah untuk dipelajari.\n\n");  
  
        // Print number  
        System.out.println(50);  
        System.out.println(3.14);  
        System.out.println(10 * 5);  
  
        // Print boolean  
        System.out.println(true); // output: true  
        System.out.println(false);  
  
        // Java comments  
        // This is a single line comment  
        /*  
        * ini adalah komentar multi line  
        * ini adalah komentar multi line  
        */  
    }  
}
```

- `System.out.println()`: digunakan untuk mencetak teks ke layar dan berpindah ke baris baru.
- `System.out.print()`: mencetak teks tanpa berpindah ke baris baru, kecuali ada `\n` (new line)
- Bisa mencetak angka langsung tanpa tanda kutip.
- Bisa mencetak nilai `true` dan `false`.
- Komentar satu baris menggunakan `//`
- Komentar multi-baris menggunakan `/* ... */`
- Komentar ini tidak akan dieksekusi oleh program.

### c. Variable dan Tipe Data

```
public class Variabel {
    public static void main(String[] args) {
        int umur = 20;           // Integer
        double gaji = 3.5;       // Double
        char inisial = 'A';      // Karakter
        boolean aktif = true;    // Boolean
        String nama = "James";   // String

        System.out.println("Nama: " + nama + ", Umur: " + umur);
    }
}
```

Tipe data utama pada Java:

Tipe Data	Contoh
int (bilangan bulat)	int x = 10;
double (bilangan desimal)	double y = 5.5;
char (karakter)	char z = 'A';
boolean (true/false)	boolean isOn = true;
String (teks)	String nama = "Java";

#### - Tipe Data Number

Tipe data number adalah tipe data yang digunakan untuk menyimpan data numerik.

Tipe data number terdiri dari:

1. Integer
2. Floating-point

```
// byte
// Tipe data byte digunakan untuk menyimpan bilangan bulat dengan
// rentang -128 sampai 127.
byte numberByte = 100;
System.out.println(numberByte);
```

```

// short
// Tipe data short digunakan untuk menyimpan bilangan bulat dengan
rentang -32,768 sampai 32,767.
short numberShort = 1000;
System.out.println(numberShort);

// int
// Tipe data int digunakan untuk menyimpan bilangan bulat dengan
rentang -2,147,483,648 sampai 2,147,483,647.
int numberInt = 1000000;
System.out.println(numberInt);

// long
// Tipe data long digunakan untuk menyimpan bilangan bulat dengan
rentang -9,223,372,036,854,775,808 sampai 9,223,372,036,854,775,807.
long numberLong = 1000000000;
System.out.println(numberLong);

// Floating-point
// float
// Tipe data float digunakan untuk menyimpan bilangan desimal dengan
rentang 3.4e-038 sampai 3.4e+038.
float numberFloat = 3.14f;
System.out.println(numberFloat);

// double
// Tipe data double digunakan untuk menyimpan bilangan desimal
dengan rentang 1.7e-308 sampai 1.7e+308.
double numberDouble = 3.14;
System.out.println(numberDouble);

// Cara lain penulisan tipe data number
// Tipe data number juga bisa ditulis dengan cara
// menggunakan underscore (_) sebagai pemisah angka.
int numberInt2 = 1_000_000;
System.out.println(numberInt2);
int numberInt3 = 10_000_000;
System.out.println(numberInt3);

```

#### - Tipe data Boolean

Tipe data boolean adalah tipe data yang digunakan untuk menyimpan nilai kebenaran (true/false)

```

// true
// true digunakan untuk menyatakan nilai benar.

```



```

boolean isJavaFun = true;
System.out.println(isJavaFun); // output: true

// false
// false digunakan untuk menyatakan nilai salah.
boolean isFishTasty = false;
System.out.println(isFishTasty); // output: false

// Tipe data boolean digunakan untuk membuat keputusan dalam
program.
// Contoh:
// Jika nilai isJavaFun adalah true, maka cetak "Java is
fun!".
if (isJavaFun) {
    System.out.println("Java is fun!");
} else {
    System.out.println("Java is not fun!");
}

```

#### - Tipe data Karakter

Tipe data karakter adalah tipe data yang digunakan untuk menyimpan satu karakter. Dideklarasikan dengan menggunakan tanda petik satu (').

```

// Contoh tipe data karakter
char huruf = 'A';
System.out.println(huruf);

// Karakter khusus (Escape sequence)
// Karakter khusus yang dimulai dengan backslash (\) disebut escape
sequence. Digunakan untuk menampilkan karakter khusus yang sulit
System.out.println("Hello\nWorld!");
System.out.println("Hello\tWorld!");

// Menggunakan ASCII
// Tipe data karakter juga dapat menyimpan karakter ASCII.
char myVar1 = 65, myVar2 = 66, myVar3 = 67;
System.out.println(myVar1);
System.out.println(myVar2);
System.out.println(myVar3);

// Karakter kosong
char spasi = ' ';
System.out.println(spasi);

```

#### - Tipe data String

Tipe data string adalah tipe data yang digunakan untuk menyimpan teks. Dideklarasikan dengan menggunakan tanda petik dua (").

```
String nama = "John Doe";
System.out.println(nama);

// Concatenation
// Concatenation adalah proses menggabungkan dua atau lebih string.
// Concatenation dilakukan dengan menggunakan operator +.
// Contoh:
String firstName = "John";
String lastName = "Doe";
String fullName = firstName + " " + lastName;
System.out.println(fullName);
System.out.println("Hello, " + fullName + "!");
System.out.println("Hello, " + (firstName + " " + lastName) + "!");

// String length
// String length adalah jumlah karakter dalam string.
// String length dilakukan dengan menggunakan method length().
// Contoh:
String text = "Hello, World!";
System.out.println(text.length());
```

#### d. Konversi Tipe Data

Konversi tipe data di Java adalah proses mengubah satu tipe data menjadi tipe data lain. Konversi ini dapat terjadi secara implisit (otomatis) atau eksplisit (manual/casting).

##### 1) Widening Casting (Otomatis)

Konversi ini terjadi secara otomatis oleh Java tanpa perlu casting, saat data dari tipe kecil dikonversi ke tipe lebih besar.

Aturan Umum:

- Tidak ada kehilangan data karena tipe target lebih besar.
- Terjadi otomatis jika tidak ada risiko kehilangan presisi.

Urutan Konversi Otomatis:

byte -> short -> int -> long -> float -> double

```
byte numberByte = 100;
short numberShort = numberByte;
int numberInt = numberShort;
```

```

long numberLong = numberInt;
float numberFloat = numberLong;
double numberDouble = numberFloat;

System.out.println(numberByte);
System.out.println(numberShort);
System.out.println(numberInt);
System.out.println(numberLong);
System.out.println(numberFloat);
System.out.println(numberDouble);

```

## 2). Narrowing Casting (Manual)

Konversi ini dilakukan secara manual menggunakan type casting, karena bisa menyebabkan kehilangan data atau presisi.

Aturan Umum:

- Harus dilakukan secara manual dengan casting.
- Bisa menyebabkan kehilangan data jika nilai lebih besar dari kapasitas tipe target.

Urutan konversi manual:

double -> float -> long -> int -> char -> short -> byte

```

double numberDouble2 = 3.14;
float numberFloat2 = (float) numberDouble2;
long numberLong2 = (long) numberFloat2;
int numberInt2 = (int) numberLong2;
short numberShort2 = (short) numberInt2;
byte numberByte2 = (byte) numberShort2;

System.out.println(numberDouble2);
System.out.println(numberFloat2);
System.out.println(numberLong);
System.out.println(numberInt2);
System.out.println(numberShort2);
System.out.println(numberByte2);

```

## 3) Konversi String

Konversi juga bisa terjadi antara tipe data yang tidak sejenis, seperti String ke angka atau sebaliknya.

```

// String ke angka
String teks = "100";
int angka = Integer.parseInt(teks); // String → int
double desimal = Double.parseDouble("3.14"); // String → double
System.out.println(angka + 50); // Output: 150

// Angka ke String

```

```
int umur = 20;
String teksUmur = String.valueOf(umur); // int → String
System.out.println(teksUmur + " tahun"); // Output: "20
tahun"
```

## e. Operators

Operator dalam Java adalah simbol khusus yang digunakan untuk melakukan operasi pada variabel dan nilai. Java memiliki beberapa jenis operator yang digunakan untuk berbagai keperluan, seperti perhitungan matematika, perbandingan, logika, dan manipulasi bit.

### 1). Operator Aritmatika (+ \* / - % ++)

Operator ini digunakan untuk melakukan operasi matematika seperti penjumlahan, pengurangan, perkalian, dll.

Operator	Fungsi	Contoh
+	Penjumlahan	a + b
-	Pengurangan	a - b
*	Perkalian	a * b
/	Pembagian	a / b
%	Modulus (Sisa bagi)	a % b

```
int a = 10, b = 5;
System.out.println(a + b); // Penjumlahan
System.out.println(a - b); // Pengurangan
System.out.println(a * b); // Perkalian
System.out.println(a / b); // Pembagian
System.out.println(a % b); // Modulus (sisa bagi)
```

### 2) Operator Penugasan (Assignment Operators)

Operator ini digunakan untuk memberikan nilai ke dalam variabel.

Operator	Fungsi	Contoh
=	Menetapkan nilai	a = 10
+=	Menambahkan dan menetapkan	a += 5 (sama dengan a = a + 5)
-=	Mengurangi dan menetapkan	a -= 3 (sama dengan a = a - 3)
*=	Mengalikan dan menetapkan	a *= 2 (sama dengan a = a * 2)
/=	Membagi dan menetapkan	a /= 2 (sama dengan a = a / 2)

%=	Modulus menetapkan	dan a %= 3 (sama dengan a = a % 3)
----	-----------------------	---------------------------------------

### 3). Operator Perbandingan (Comparison Operators)

Operator ini digunakan untuk membandingkan dua nilai dan menghasilkan true atau false.

Operator	Fungsi	Contoh
==	Sama dengan	a == b
!=	Tidak sama dengan	a != b
>	Lebih besar	a > b
<	Lebih kecil	a < b
>=	Lebih besar atau sama dengan	a >= b
<=	Lebih kecil atau sama dengan	a <= b

### 4). Operator Logika (Logical Operators)

Operator ini digunakan dalam operasi logika, biasanya untuk pengambilan keputusan dengan tipe boolean.

Operator	Fungsi	Contoh
&&	Logika AND (true jika kedua operand true)	a && b
	Menghasilkan true jika salah satu operand true	true    false, hasilnya: false
!	Logika NOT (membalik nilai boolean)	!a

## f. Input

Dalam Java, input digunakan untuk menerima data dari pengguna saat program dijalankan. Java menyediakan beberapa cara untuk mengambil input, dengan metode yang paling umum adalah menggunakan kelas Scanner, BufferedReader, dan Console.

- Menggunakan Scanner (Paling Umum): Kelas Scanner dari paket java.util adalah cara paling sering digunakan untuk membaca input dari pengguna.

Cara Menggunakan Scanner:

- Import kelas Scanner
- Buat objek Scanner dengan System.in
- Gunakan metode next() atau nextLine() untuk membaca input

```
import java.util.Scanner;

public class Input {
    public static void main(String[] args) {
```

```

        System.out.print("Masukkan nama Anda: ");
        Scanner input = new Scanner(System.in);
        String name = input.nextLine();
        System.out.print("Masukkan umur Anda: ");
        int age = input.nextInt();
        System.out.print("Masukkan tinggi badan Anda: ");
        double height = input.nextDouble();

        System.out.println("\nData yang Anda masukkan:");
        System.out.println("Nama Anda: " + name);
        System.out.println("Umur Anda: " + age);
        System.out.println("Tinggi badan Anda: " + height);

        input.close();
    }
}

```

Metode Scanner yang Sering Digunakan:

Metode	Tipe Data yang Dibaca	Contoh Penggunaan
nextLine()	String (teks)	input.nextLine();
next()	String (satu kata)	input.next();
nextInt()	Integer	input.nextInt();
nextDouble()	Double (desimal)	input.nextDouble();
nextBoolean()	Boolean (true/false)	input.nextBoolean();

Perbedaan next() dan nextLine():

- next() hanya membaca satu kata (berhenti saat ada spasi).
- nextLine() membaca seluruh baris teks (termasuk spasi).

### g. Java Math

Java menyediakan kelas Math di dalam package java.lang yang berisi berbagai metode dan konstanta untuk operasi matematika seperti perhitungan pangkat, akar kuadrat, trigonometri, nilai absolut, pembulatan, dan banyak lagi.

```

public class Math {
    public static void main(String[] args) {
        System.out.println(Math.max(10, 20));
        System.out.println(Math.min(10, 20));
        System.out.println(Math.sqrt(64));
        System.out.println(Math.abs(-4.7));
        System.out.println(Math.random());
    }
}

```

### 1). Konstanta dalam Math

Kelas Math memiliki beberapa konstanta matematika yang sering digunakan, seperti:

Konstanta	Nilai
Math.PI	3.141592653589793 (nilai $\pi$ )
Math.E	2.718281828459045 (bilangan Euler)

### 2) Metode Matematika Dasar

Metode	Fungsi	Contoh
Math.abs(x)	Mengembalikan nilai absolut	Math.abs(-5) $\rightarrow$ 5
Math.max(a, b)	Mengembalikan nilai terbesar	Math.max(10, 20) $\rightarrow$ 20
Math.min(a, b)	Mengembalikan nilai terkecil	Math.min(10, 20) $\rightarrow$ 10
Math.sqrt(x)	Mengembalikan akar kuadrat	Math.sqrt(25) $\rightarrow$ 5.0
Math.cbrt(x)	Mengembalikan akar kubik	Math.cbrt(27) $\rightarrow$ 3.0
Math.pow(a, b)	Pangkat ( $a^b$ )	Math.pow(2, 3) $\rightarrow$ 8.0

### 3) Metode Pembulatan

Metode	Fungsi	Contoh
Math.round(x)	Pembulatan ke bilangan terdekat	Math.round(4.6) $\rightarrow$ 5
Math.floor(x)	Pembulatan ke bawah	Math.floor(4.9) $\rightarrow$ 4.0
Math.ceil(x)	Pembulatan ke atas	Math.ceil(4.1) $\rightarrow$ 5.0

### 4) Metode Random Angka Acak

Metode	Fungsi	Contoh
Math.random()	Menghasilkan angka acak antara 0.0 dan 1.0	Math.random()
(int) (Math.random() * n)	Angka acak antara 0 dan n-1	(int) (Math.random() * 10) (0-9)

## Latihan

Kerjakan soal dibawah ini!

Latihan	
Durasi Pengerjaan	30 menit
Start	15:40
End	16:10

Soal	<p>Buatlah sebuah program Java yang meminta pengguna memasukkan nama, NIM, usia, dan tinggi badan, lalu menampilkan kembali informasi tersebut dengan format yang rapi.</p> <p>Setelah itu, program harus melakukan hal berikut:</p> <ol style="list-style-type: none"> <li>1) Menggunakan operator aritmatika untuk menghitung hasil dari <math>(\text{usia} * 2) + 10 / 5 - 3</math>, kemudian menampilkan hasilnya.</li> <li>2) Menggunakan operator perbandingan untuk mengecek apakah usia lebih besar dari 18 dan menampilkan hasilnya (true atau false).</li> <li>3) Menggunakan operator logika untuk mengecek apakah usia lebih besar dari 18 dan tinggi badan lebih dari 160 cm, lalu menampilkan hasilnya.</li> <li>4) Menggunakan konversi tipe data, yaitu: <ul style="list-style-type: none"> <li>• Mengonversi usia (int) menjadi double (casting).</li> <li>• Mengonversi tinggi badan (double) menjadi int (casting).</li> <li>• Menampilkan hasil dari konversi tersebut.</li> </ul> </li> </ol>
Link Pengumpulan	

## Posttest

Kerjakan soal dibawah ini!

Pengerjaan Posttest	
Durasi Pengerjaan	30 menit
Start	16:10
End	16:40
Soal	<p>Studi Kasus: Manajemen Keuangan Pribadi</p> <p>Seorang individu bernama Budi ingin mengembangkan aplikasi manajemen keuangan pribadi untuk membantunya mengelola pengeluaran sehari-hari. Aplikasi ini harus dapat melakukan berbagai perhitungan berdasarkan pendapatan, pengeluaran, dan status keuangan Budi.</p> <p>Sebagai seorang programmer, tugas Anda adalah membuat program Java yang memenuhi kebutuhan berikut:</p> <ol style="list-style-type: none"> <li>1) Menerima Input dari Pengguna Program harus meminta pengguna memasukkan nama, usia, jumlah uang yang dimiliki, dan rata-rata pengeluaran harian, menggunakan Scanner.</li> <li>2) Konversikan usia (int) ke double, Konversikan jumlah uang (double) ke int.</li> <li>3) Perhitungan Keuangan: <ul style="list-style-type: none"> <li>• Estimasi sisa uang dalam 30 hari</li> <li>• Estimasi bulan bertahan dengan rumus : <math>\text{Bulan bertahan} = \frac{\text{uang yang dimiliki}}{(\text{pengeluaran harian} * 30)}</math>, Jika</li> </ul> </li> </ol>



	<p>hasilnya kurang dari 1 bulan maka PERINGATAN: Keuangan Anda kurang stabil!</p> <ul style="list-style-type: none"> <li>• Jika lebih dari 6 bulan, maka Keuangan Anda dalam kondisi aman.</li> </ul> <p>4) Operator Perbandingan &amp; Logika:</p> <ul style="list-style-type: none"> <li>• Tentukan apakah usia lebih dari 30 tahun.</li> <li>• Tentukan apakah usia lebih dari 30 DAN memiliki uang lebih dari Rp10.000.000.</li> <li>• Tentukan apakah usia kurang dari 30 ATAU memiliki uang lebih dari Rp5.000.000.</li> </ul> <p>Contoh Output :</p> <p>Apakah usia lebih dari 30? false  Apakah usia &gt; 30 dan uang &gt; 10 juta? false  Apakah usia &lt; 30 atau uang &gt; 5 juta? true</p> <p>5) Program meminta pengguna memasukkan jumlah hutang. Gunakan Math untuk menampilkan:</p> <ul style="list-style-type: none"> <li>• Nilai absolut dari hutang.</li> <li>• Pembulatan ke atas untuk pengeluaran harian menggunakan Math.ceil().</li> <li>• Bilangan acak antara Rp100.000 - Rp1.000.000 sebagai "bonus tak terduga".</li> </ul> <p>6) Menampilkan Rangkuman Keuangan</p> <p>Contoh :</p> <p>=== LAPORAN KEUANGAN PRIBADI ===  Nama: Budi  Usia: 25 tahun  Uang yang dimiliki: Rp5000000  Pengeluaran harian rata-rata: Rp150000  Sisa uang dalam 30 hari: Rp500000  Estimasi bulan bertahan: 1.1 bulan  Status Keuangan: PERINGATAN: Keuangan Anda kurang stabil!  Nilai absolut dari hutang: Rp1500000  Pengeluaran harian setelah pembulatan: Rp150000  Total uang yang dimiliki setelah pengeluaran dan bonus: Rp1220000  Bonus tak terduga: Rp720000</p>
Link Pengumpulan	

### Take Home Task

Kerjakan soal dibawah ini!

Tugas	
Durasi Pengerjaan	6 hari
Start	Kamis
End	Rabu, 23:59
Template laporan	<a href="#">contoh template laporan tht</a>

Soal	<p><b>Studi Kasus: Sistem Pengelolaan Nilai Mahasiswa</b></p> <p>Seorang dosen ingin memiliki program sederhana untuk mengelola data nilai mahasiswa secara otomatis. Saat ini, ia kesulitan dalam mencatat nilai setiap mahasiswa secara manual dan ingin memiliki sistem yang dapat membantu menghitung rata-rata nilai, menentukan IPK berdasarkan nilai tersebut, serta mengevaluasi performa akademik mahasiswa secara otomatis.</p> <p>Dosen ingin sistem ini memiliki fitur yang dapat menerima data mahasiswa seperti nama, NIM, usia, dan jumlah mata kuliah yang diambil. Setiap mahasiswa akan memasukkan nilai mata kuliah yang kemudian dihitung rata-ratanya untuk menentukan IPK mereka berdasarkan skala 4.0.</p> <p>Selain itu, sistem juga harus dapat melakukan evaluasi akademik mahasiswa dengan cara:</p> <p>Menentukan apakah mahasiswa berusia lebih dari 22 tahun.  Menentukan apakah IPK mahasiswa lebih dari atau sama dengan 3.5 DAN jumlah mata kuliah lebih dari 4.  Menentukan apakah IPK mahasiswa kurang dari 2.5 ATAU jumlah mata kuliah kurang dari 4.</p> <p>Dosen juga menginginkan agar sistem ini dapat memberikan nomor antrian konsultasi akademik secara acak kepada mahasiswa serta menampilkan laporan akademik mahasiswa dalam format yang rapi.</p> <p>Sebagai seorang pengembang perangkat lunak, tugas Anda adalah membuat program Java berbasis konsol untuk memenuhi kebutuhan tersebut.</p> <p>HINT : Gunakan apa yang sudah dipelajari</p>
------	--

### Tugas Pendahuluan Modul 3

Link Pengumpulan:

Deadline: Rabu, 12 Maret 2025, pukul 23:59

#### Soal

1. Jelaskan perbedaan antara struktur kontrol if-else dan switch-case. Dalam kondisi seperti apa switch-case lebih direkomendasikan dibandingkan if-else?
2. Perhatikan kode berikut :

```
int nilai = 75;
if (nilai >= 80) {
    System.out.println("Grade: A");
}
```

```
} else if (nilai >= 70) {  
    System.out.println("Grade: B");  
} else {  
    System.out.println("Grade: C");  
}
```

- a) Jika nilai diubah menjadi 85, apa output yang dihasilkan?
  - b) Ubah kode di atas menggunakan switch-case (asumsikan nilai hanya bisa 60, 70, atau 80).
3. Apa perbedaan utama antara for-loop, while-loop, dan do-while-loop dalam Java?
  4. Berikan contoh penggunaan foreach loop pada sebuah array yang berisi nama-nama mahasiswa!
  5. Apa itu array dalam Java? Sebutkan perbedaan antara array satu dimensi dan array dua dimensi, serta berikan contoh deklarasi masing-masing!
  6. Diberikan array berikut :  
`int[] angka = {5, 10, 15, 20, 25};`  
Bagaimana cara menampilkan semua elemen array di atas menggunakan for-loop?
  7. Tuliskan kode program singkat untuk menghitung jumlah semua elemen dalam array berikut:  
`int[] data = {4, 8, 12, 16, 20};`
  8. Bagaimana cara menambahkan elemen baru ke dalam array? Apakah ukuran array bisa diubah setelah dibuat?
  9. Apa perbedaan antara String dan char dalam Java?
  10. Berikan contoh penggunaan metode String `length()`, `toUpperCase()`, dan `substring()` dalam Java!
  11. Jelaskan cara membalikkan urutan karakter dalam sebuah String tanpa menggunakan metode bawaan seperti `reverse()`!
  12. Tuliskan program untuk menghitung jumlah huruf vokal (a, e, i, o, u) dalam sebuah String yang dimasukkan oleh pengguna!