

**LAPORAN PRAKTIKUM
PEMROGRAMAN II
MODUL 1**



STRUKTUR DASAR KODE JAVA

Oleh:

Indra Suryadilaga NIM. 2410817310014

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
SEPTEMBER 2025**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN II
MODUL 1

Laporan Praktikum Pemrograman II Modul 1: Struktur Dasar Kode Java Sederhana ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman II. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Indra Suryadilaga
NIM : 2410817310014

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Galih Aji Sabdaraya
NIM. 2310817210005

Andreyan Rizky Baskara, S.Kom.,
M.Kom.
NIP. 19930703 201903 1 011

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI.....	3
DAFTAR GAMBAR	4
DAFTAR TABEL	5
SOAL 1	6
A. Source Code	6
B. Output Program.....	9
C. Pembahasan.....	9
SOAL 2	12
A. Source Code	12
B. Output Program.....	13
C. Pembahasan.....	13
SOAL 3	15
A. Source Code	15
B. Output Program.....	16
C. Pembahasan.....	16
SOAL 4	18
A. Source Code	18
B. Output Program.....	20
C. Pembahasan.....	20
SOAL 5	22
A. Source Code	22
B. Output Program.....	23
C. Pembahasan.....	23
GITHUB.....	25

DAFTAR GAMBAR

Gambar 1. Screenshot Hasil Jawaban Soal 1	9
Gambar 2. Screenshot Hasil Jawaban Soal 2	13
Gambar 3. Screenshot Hasil Jawaban Soal 3	16
Gambar 4. Screenshot Hasil Jawaban Soal 4	20
Gambar 5. Screenshot Hasil Jawaban Soal 5	23

DAFTAR TABEL

Tabel 1.1 Soal No. 1.....	6
Tabel 1.2 Source Code	6
Tabel 2.1 Soal No. 2.....	12
Tabel 2.2 Source Code	12
Tabel 3.1 Soal No. 3.....	15
Tabel 3.2 Source Code	15
Tabel 4.1 Soal No. 4.....	18
Tabel 4.2 Source Code	18
Tabel 5.1 Soal No.5.....	22
Tabel 5.2 Source Code	22

SOAL 1

Buatlah program yang dapat menerima input dan menghasilkan output seperti berikut. Gunakan tipe data yang tepat dalam menyimpan data yang diinputkan. Program harus bersifat dinamis, artinya input dapat diubah-ubah dan output akan menyesuaikan, bukan hardcoded.

Tabel 1.1 Soal No. 1

Input
Masukkan Nama Lengkap: Lucy Drien Masukkan Tempat Lahir: Jerman Masukkan Tanggal Lahir: 7 Masukkan Bulan Lahir: 9 Masukkan Tahun Lahir: 1878 Masukkan Tinggi Badan: 174 Masukkan Berat Badan: 54.89
Output
Nama Lengkap Lucy Drien, Lahir di Jerman pada Tanggal 7 September 1878 Tinggi Badan 174 cm dan Berat Badan 54.89 kilogram

Simpan dengan nama file: **PRAK101-NIM-Nama.java**

A. Source Code

Tabel 1.2 Source Code

1	<code>package modulSatu;</code>
2	<code>import java.util.Scanner;</code>
3	
4	<code>public class PRAK101_2410817310014_IndraSuryadilaga</code>
5	<code>{</code>
6	
7	<code> public static void main(String[] args) {</code>
8	<code> Scanner input = new Scanner(System.in);</code>
9	
10	<code> // --- Input <u>Nama</u> dan <u>Tempat Lahir</u> ---</code>
11	<code> System.out.print("Masukkan Nama Lengkap: ");</code>
12	<code> String namaLengkap = input.nextLine();</code>
13	
14	<code> System.out.print("Masukkan Tempat Lahir: ");</code>

```

15         String tempatLahir = input.nextLine();
16
17         // --- Variabel untuk Data Kelahiran ---
18         int tanggalLahir;
19         int bulanLahir;
20         int tahunLahir;
21         boolean tanggalValid = false;
22
23         // --- Perulangan untuk Validasi Data
24         Kelahiran ---
25         do {
26             System.out.print("Masukkan Tanggal
27 Lahir: ");
28             tanggalLahir = input.nextInt();
29
30             System.out.print("Masukkan Bulan Lahir:
31 ");
32             bulanLahir = input.nextInt();
33
34             System.out.print("Masukkan Tahun Lahir:
35 ");
36             tahunLahir = input.nextInt();
37
38             // Memeriksa validitas dasar bulan dan
39             tahun terlebih dahulu
40             if (bulanLahir < 1 || bulanLahir > 12 ||
41 tahunLahir <= 0) {
42                 System.out.println("Bulan atau
43 Tahun tidak valid. Silakan masukkan kembali data
44 kelahiran Anda.");
45                 continue;
46             }
47
48             // Menentukan jumlah hari maksimum dalam
49             sebulan (termasuk tahun kabisat)
50             int maxTanggal;
51             if (bulanLahir == 2) {
52                 // Cek tahun kabisat (leap year)
53                 boolean isKabisat = (tahunLahir % 4
54 == 0 && tahunLahir % 100 != 0) || (tahunLahir % 400
55 == 0);
56                 maxTanggal = isKabisat ? 29 : 28;
57             } else if (bulanLahir == 4 || bulanLahir
58 == 6 || bulanLahir == 9 || bulanLahir == 11) {
59                 maxTanggal = 30;

```

```

60         } else {
61             maxTanggal = 31;
62         }
63
64         // Memeriksa apakah tanggal yang
65 dimasukkan sesuai dengan rentang bulan dan tahun
66         if (tanggalLahir >= 1 && tanggalLahir <=
67 maxTanggal) {
68             tanggalValid = true;
69         } else {
70             System.out.println("=> ERROR:
71 Tanggal " + tanggalLahir + " tidak ada pada bulan
72 ke-" + bulanLahir + " tahun " + tahunLahir + ".
73 Silakan masukkan kembali data kelahiran Anda.");
74         }
75
76         } while (!tanggalValid);
77
78
79         // --- Validasi Input Tinggi dan Berat Badan
80 ---
81         int tinggiBadan;
82         do {
83             System.out.print("Masukkan Tinggi
84 Badan: ");
85             tinggiBadan = input.nextInt();
86             if (tinggiBadan <= 0) {
87                 System.out.println("=> ERROR:
88 Tinggi badan harus angka positif.");
89             }
90         } while (tinggiBadan <= 0);
91
92         double beratBadan;
93         do {
94             System.out.print("Masukkan Berat Badan:
95 ");
96             beratBadan = input.nextDouble();
97             if (beratBadan <= 0) {
98                 System.out.println("=> ERROR: Berat
99 badan harus angka positif.");
100             }
101         } while (beratBadan <= 0);
102
103         // --- Proses dan Output ---
104         String[] namaBulanArray = {

```

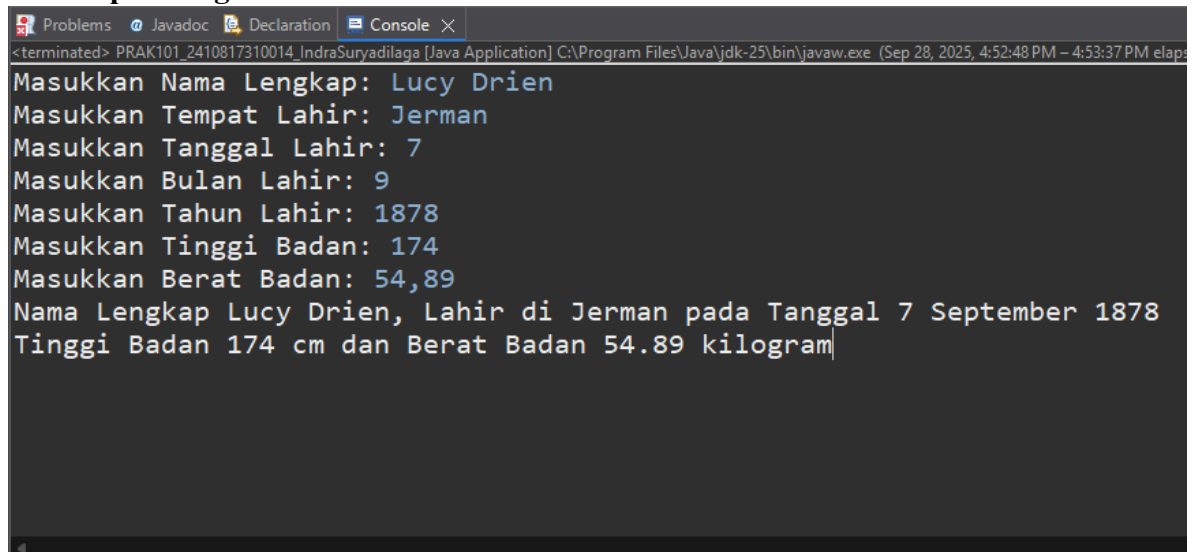


```

105         "", "Januari", "Februari", "Maret",
106 "April", "Mei", "Juni",
107         "Juli", "Agustus", "September",
108 "Oktober", "November", "Desember"
109     };
110
111     String namaBulan =
112 namaBulanArray[bulanLahir];
113
114     System.out.println("Nama Lengkap " +
115 namaLengkap + ", Lahir di " + tempatLahir + " pada
116 Tanggal " + tanggalLahir + " " + namaBulan + " " +
117 tahunLahir);
118     System.out.println("Tinggi Badan " +
119 tinggiBadan + " cm dan Berat Badan " + beratBadan +
120 " kilogram");
121
122     System.out.println();
123     input.close();
124 }
125 }

```

B. Output Program



The screenshot shows a Java IDE window with the 'Console' tab active. The output of the program is as follows:

```

<terminated> PRAK101_2410817310014_IndraSuryadilaga [Java Application] C:\Program Files\Java\jdk-25\bin\javaw.exe (Sep 28, 2025, 4:52:48 PM – 4:53:37 PM elapsed)
Masukkan Nama Lengkap: Lucy Drien
Masukkan Tempat Lahir: Jerman
Masukkan Tanggal Lahir: 7
Masukkan Bulan Lahir: 9
Masukkan Tahun Lahir: 1878
Masukkan Tinggi Badan: 174
Masukkan Berat Badan: 54,89
Nama Lengkap Lucy Drien, Lahir di Jerman pada Tanggal 7 September 1878
Tinggi Badan 174 cm dan Berat Badan 54.89 kilogram

```

Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Pada program ini, baris pertama `package modulSatu;` digunakan untuk mendefinisikan lokasi file Java berada di dalam sebuah package bernama *modulSatu*.

Lalu, pada baris `import java.util.Scanner;`, kita mengimpor library bawaan Java yang menyediakan kelas `Scanner`, yaitu komponen yang dipakai untuk membaca input dari keyboard.

Selanjutnya, bagian `public class PRAK101_2410817310014_IndraSuryadilaga` { mendefinisikan kelas utama dengan nama sesuai aturan praktikum. Di dalam kelas ini, terdapat `public static void main(String[] args)`, yaitu titik masuk utama program Java yang akan dijalankan pertama kali.

Di dalam method `main`, `Scanner input = new Scanner(System.in);` berfungsi membuat objek bernama *input* yang bisa digunakan untuk membaca data dari pengguna. Baris `String namaLengkap = input.nextLine();` dan `String tempatLahir = input.nextLine();` menyimpan input teks berupa nama lengkap dan tempat lahir.

`int tanggalLahir;; int bulanLahir;; int tahunLahir;; boolean tanggalValid = false;` Variabel ini digunakan untuk menyimpan data tanggal lahir dengan tipe `int`, sementara `tanggalValid` sebagai indikator apakah tanggal yang dimasukkan sudah sesuai aturan atau belum.

Proses validasi dilakukan pada blok `do { ... } while (!tanggalValid);`. Di dalamnya, program meminta tanggal, bulan, dan tahun lahir. Baris `boolean isKabisat = (tahunLahir % 4 == 0 && tahunLahir % 100 != 0) || (tahunLahir % 400 == 0);` digunakan untuk menentukan apakah sebuah tahun termasuk tahun kabisat atau tidak, sehingga bulan Februari bisa memiliki 28 atau 29 hari. Selain itu, kondisi seperti `if (bulanLahir == 4 || bulanLahir == 6 || bulanLahir == 9 || bulanLahir == 11)` digunakan untuk memastikan bulan dengan 30 hari ditangani secara tepat.

Validasi tinggi badan ditangani oleh blok `do { ... } while (tinggiBadan <= 0);`. Di sini digunakan tipe `int` karena tinggi badan tidak memerlukan angka desimal. Hal yang sama juga berlaku untuk berat badan, hanya saja dipakai tipe `double` agar dapat menampung nilai dengan pecahan desimal seperti 54.89.

Akhir program menyiapkan array `String[] namaBulanArray = { ... }`; untuk menyimpan daftar nama bulan dari Januari sampai Desember. Nantinya, `String namaBulan = namaBulanArray[bulanLahir];` mengambil nama bulan sesuai angka yang dimasukkan. Output diformat dengan `System.out.println(...)` yang menyusun kalimat lengkap berisi nama, tempat lahir, tanggal lahir dalam format natural, serta tinggi dan berat badan beserta satuannya.

SOAL 2

Buatlah program yang dapat menampilkan deret bilangan sebanyak 10 baris menggunakan perulangan **while**. Tampilkan deret sesuai kondisi, jika bilangan pada deret tersebut adalah kelipatan 5, maka bilangan tersebut harus dibagi 5 kemudian dikurangi 1, dan ditampilkan. Input adalah angka awal memulai deret.

Tabel 2.1 Soal No. 2

Input	Output
10	1,11,12,13,14,2,16,17,18,19,3
98	98, 99, 19, 101, 102, 103, 104, 20, 106, 107, 108
46	?

Simpan dengan nama file: **PRAK102-NIM-Nama.java**

A. Source Code

Tabel 2.2 Source Code

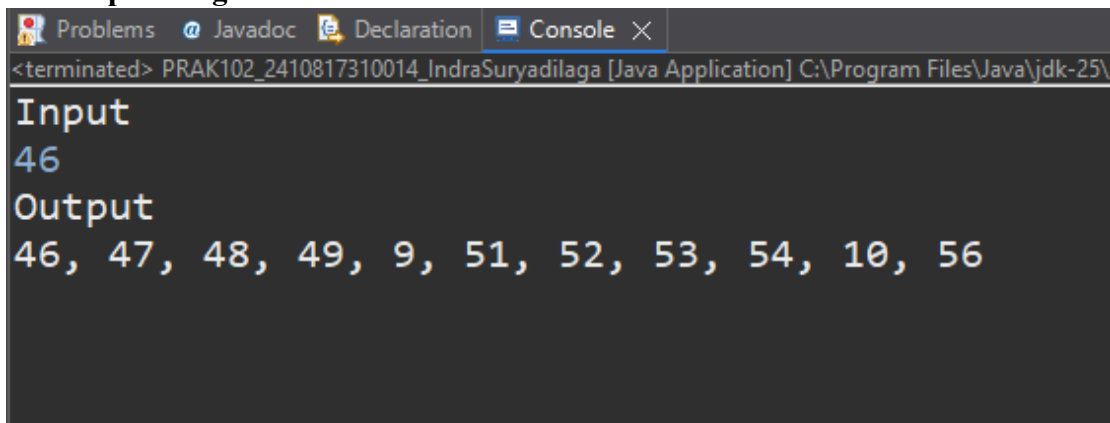
```
1 package modulSatu;
2 import java.util.Scanner;
3
4 public class PRAK102_2410817310014_IndraSuryadilaga {
5
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.println("Input");
10        int angkaAwal = input.nextInt();
11
12        System.out.println("Output");
13        int i = 0;
14
15        while (i <= 10) {
16
17            if (angkaAwal % 5 == 0) {
18                int hasil = (angkaAwal / 5) - 1;
19                System.out.print(hasil);
20            } else {
21                System.out.print(angkaAwal);
22            }
23        }
```

```

24         if (i < 10) {
25             System.out.print(", ");
26         }
27
28         angkaAwal++;
29         i++;
30     }
31
32     System.out.println();
33     input.close();
34 }
35
36 }

```

B. Output Program



The screenshot shows a Java IDE window with the 'Console' tab selected. The title bar indicates the application is 'PRAK102_2410817310014_IndraSuryadilaga [Java Application]'. The console output shows the program has terminated. The input '46' is shown in blue, and the output '46, 47, 48, 49, 9, 51, 52, 53, 54, 10, 56' is shown in yellow.

```

<terminated> PRAK102_2410817310014_IndraSuryadilaga [Java Application] C:\Program Files\Java\jdk-25\
Input
46
Output
46, 47, 48, 49, 9, 51, 52, 53, 54, 10, 56

```

Gambar 2. Screenshot Hasil Jawaban Soal 2

C. Pembahasan

Pada program ini, baris `package modulSatu;` dan `import java.util.Scanner;` sama seperti soal sebelumnya, dipakai untuk mengatur lokasi file dalam package serta mengimpor library Scanner agar bisa membaca input dari pengguna. Lalu, kelas utama didefinisikan dengan `public class PRAK102_2410817310014_IndraSuryadilaga {` dan titik awal eksekusi ada pada method `public static void main(String[] args).`

Input angka awal dibaca dengan perintah `int angkaAwal = input.nextInt();`. Nilai ini menjadi titik awal deret bilangan yang akan ditampilkan. Setelah itu, dibuat variabel penghitung `int i = 0;` untuk mengontrol jumlah bilangan yang dicetak.

Bagian utama program menggunakan perulangan `while (i <= 10) { ... }`. Kondisi ini memastikan proses berjalan sampai indeks `i` mencapai 10, sehingga total ada 11 bilangan yang dicetak.

Di dalam perulangan, terdapat kondisi `if (angkaAwal % 5 == 0 { ... } else {`. berfungsi untuk mengecek apakah bilangan sekarang merupakan kelipatan 5. Jika iya, bilangan tersebut diproses dengan cara dibagi 5 lalu dikurangi 1 sebelum ditampilkan. Contohnya, jika angka awal 10, maka $10 / 5 = 2$, lalu dikurangi 1 menghasilkan output **1**. Jika bukan kelipatan 5, bilangan ditampilkan apa adanya.

Setelah menampilkan angka, ada tambahan pengecekan `if (i < 10) {System.out.print(", ");}`. Bagian ini menjaga agar pemisah koma hanya muncul di antara angka, tetapi tidak di akhir deret. Setelah itu, `angkaAwal++` digunakan untuk menaikkan nilai bilangan berikutnya, dan `i++` menambah penghitung perulangan agar `while` bisa berhenti setelah mencapai jumlah yang ditentukan.

SOAL 3

Buatlah program yang dapat menampilkan deret bilangan sebanyak N baris (N diinputkan oleh pengguna) menggunakan perulangan **do-while**. Jika bilangan pada deret tersebut merupakan bilangan genap, maka bilangan tersebut tidak perlu ditampilkan. Input dari kiri ke kanan, N dan bilangan awal.

Tabel 3.1 Soal No. 3

Input	Output
10 5	5, 7, 9, 11, 13, 15, 17, 19, 21, 23
5 100	101, 103, 105, 107, 109
7 117	?

Simpan dengan nama file: **PRAK103-NIM-Nama.java**

A. Source Code

Tabel 3.2 Source Code

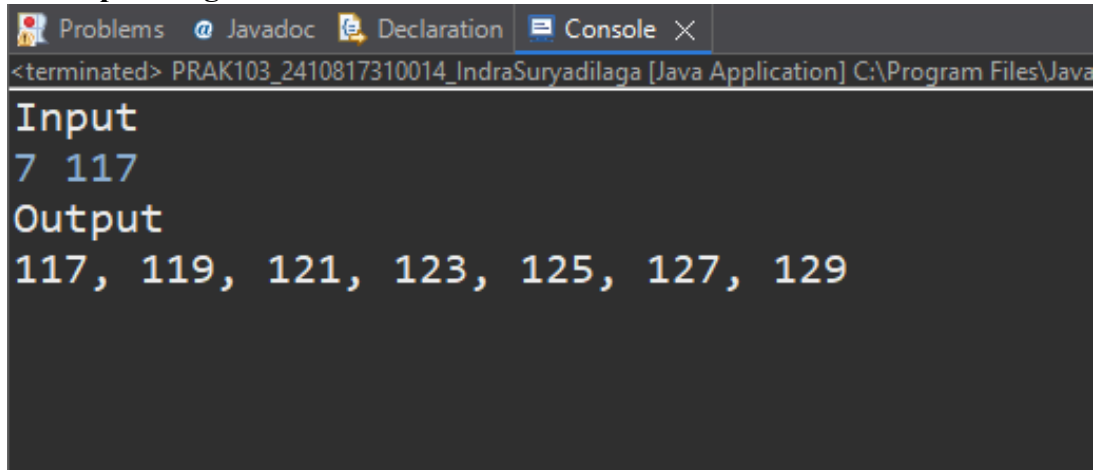
```
1 package modulSatu;
2 import java.util.Scanner;
3
4 public class PRAK103_2410817310014_IndraSuryadilaga {
5
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.println("Input");
10        int jumlahAngka = input.nextInt();
11        int angkaAwal = input.nextInt();
12
13        System.out.println("Output");
14        int i = 0;
15
16        do {
17            if (angkaAwal % 2 != 0) {
18                System.out.print(angkaAwal);
19
20                i++;
21
22                if (i < jumlahAngka) {
23                    System.out.print(", ");
24                }
25            }
26        } while (i < jumlahAngka);
27    }
28 }
```

```

25         }
26         angkaAwal++;
27
28     } while (i < jumlahAngka);
29
30     System.out.println();
31     input.close();
32 }
33 }

```

B. Output Program



The screenshot shows a Java IDE window with the 'Console' tab selected. The console output is as follows:

```

<terminated> PRAK103_2410817310014_IndraSuryadilaga [Java Application] C:\Program Files\Java
Input
7 117
Output
117, 119, 121, 123, 125, 127, 129

```

Gambar 3. Screenshot Hasil Jawaban Soal 3

C. Pembahasan

Pada program ini, baris `package modulSatu;` dan `import java.util.Scanner;` sama seperti soal sebelumnya, dipakai untuk mengatur lokasi file dalam package serta mengimpor library Scanner agar bisa membaca input dari pengguna. Lalu, kelas utama didefinisikan dengan `public class PRAK103_2410817310014_IndraSuryadilaga {` dan titik awal eksekusi ada pada method `public static void main(String[] args).`

Pertama, pengguna diminta untuk menginput dua nilai: banyaknya bilangan yang ingin ditampilkan (`jumlahAngka`) dan bilangan awal (`angkaAwal`). Keduanya dibaca dengan perintah `input.nextInt()`; . Nilai `jumlahAngka` berfungsi sebagai batas output, sedangkan `angkaAwal` sebagai titik mulai deret bilangan.

Program ini menggunakan perulangan `do { ... } while (i < jumlahAngka);`. Pemilihan perulangan `do-while` memastikan bahwa blok kode di dalamnya akan dieksekusi minimal sekali, meskipun kondisi awal tidak langsung terpenuhi. Variabel penghitung `i` dipakai untuk memastikan jumlah bilangan ganjil yang dicetak sesuai dengan input `jumlahAngka`.

Di dalam perulangan, terdapat kondisi `if (angkaAwal % 2 == 0 {...; if (jumlahAngka){...}})`. Di sini dilakukan pengecekan apakah `angkaAwal` merupakan bilangan ganjil (`angkaAwal % 2 != 0`). Jika benar, bilangan tersebut dicetak ke layar dengan `System.out.print(angkaAwal);`. Setelah itu, variabel penghitung `i` ditambah satu agar jumlah bilangan yang ditampilkan bisa terkendali. Bagian `if (i < jumlahAngka)` digunakan untuk mencetak koma di antara bilangan, tetapi tidak menambahkan koma setelah angka terakhir.

Setelah setiap iterasi, `angkaAwal++` dijalankan agar program mengecek bilangan selanjutnya. Karena hanya bilangan ganjil yang dicetak, maka bilangan genap akan otomatis dilewati tanpa ditampilkan.

SOAL 4

Abu dan Bagus sedang memainkan suit Batu-Gunting-Kertas. Aturan permainannya adalah sebagai berikut:

- a. Batu (**B**) mengalahkan Gunting (**G**).
- b. Gunting (**G**) mengalahkan Kertas (**K**).
- c. Kertas (**K**) mengalahkan Batu (**B**).
- d. Jika kedua pemain memilih tangan yang sama, hasilnya adalah seri.
- e. Terdapat tiga ronde dalam sekali permainan. Pemain melakukan suit sebanyak tiga kali.
- f. Setiap ronde akan ditentukan pemenangnya berdasarkan aturan yang telah ditentukan sebelumnya.
- g. Pemain yang mendapatkan poin terbanyak dialah pemenangnya.

Bantulah Abu dan Bagus dengan cara membuat program dalam bahasa Java untuk menentukan siapa pemenangnya. Input baris pertama adalah pilihan tangan Abu,urut dari kiri ke kanan adalah ronde 1, ronde 2, ronde 3. Lalu baris kedua adalah pilihan tangan Bagus tiap rondanya. Hasilnya antara “Abu”, “Bagas” atau “Seri”.

Tabel 4.1 Soal No. 4

Input	Output
Tangan Abu: G G K Tangan Bagus: B K G	Bagas
Tangan Abu: G K G Tangan Bagus: B K K	Seri
Tangan Abu: G K B Tangan Bagus: B K K	?

Simpan dengan nama file: **PRAK104-NIM-Nama.java**

A. Source Code

Tabel 4.2 Source Code

```

1 package modulSatu;
2 import java.util.Scanner;
3
4 public class PRAK104_2410817310014_IndraSuryadilaga {
5
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.print("Tangan Abu: ");
10        String[] tanganAbu =
11 input.nextLine().split(" ");
12
13        System.out.print("Tangan Bagas: ");
14        String[] tanganBagas =
15 input.nextLine().split(" ");
16
17        int poinAbu = 0;
18        int poinBagas = 0;
19
20        for (int i = 0; i < 3; i++) {
21            String pilihanAbu = tanganAbu[i];
22            String pilihanBagas = tanganBagas[i];
23
24            if ((pilihanAbu.equals("B") &&
25 pilihanBagas.equals("G")) ||
26                (pilihanAbu.equals("G") &&
27 pilihanBagas.equals("K")) ||
28                (pilihanAbu.equals("K") &&
29 pilihanBagas.equals("B"))) {
30                poinAbu++;
31            }
32
33            else if ((pilihanBagas.equals("B") &&
34 pilihanAbu.equals("G")) ||
35                (pilihanBagas.equals("G") &&
36 pilihanAbu.equals("K")) ||
37                (pilihanBagas.equals("K") &&
38 pilihanAbu.equals("B"))) {
39                poinBagas++;
40            }
41        }
42
43        if (poinAbu > poinBagas) {
44            System.out.println("Abu");
45        } else if (poinBagas > poinAbu) {

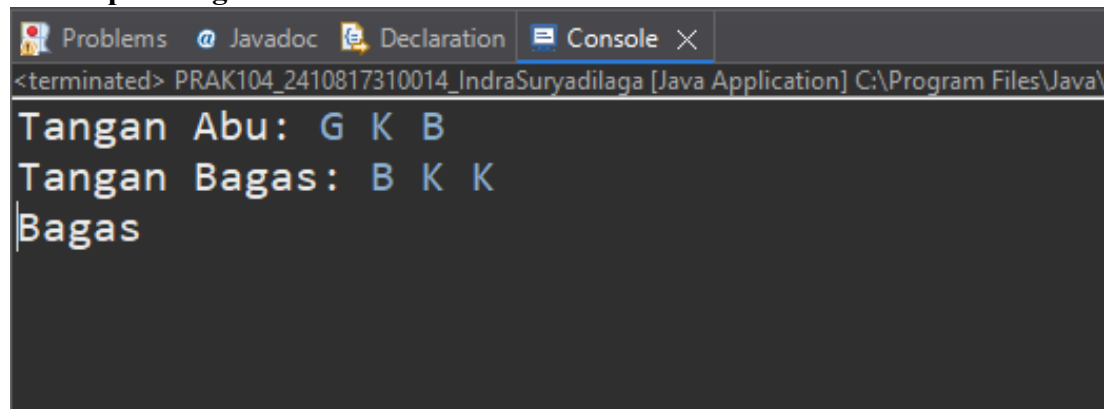
```

```

46         System.out.println("Bagas");
47     } else {
48         System.out.println("Seri");
49     }
50
51     System.out.println();
52     input.close();
53 }
56 }

```

B. Output Program



The screenshot shows a Java IDE window with the 'Console' tab selected. The title bar indicates the application is 'PRAK104_2410817310014_IndraSuryadilaga [Java Application]'. The console output displays the following text:

```

Tangan Abu: G K B
Tangan Bagas: B K K
Bagas

```

Gambar 4. Screenshot Hasil Jawaban Soal 4

C. Pembahasan

Pada program ini, baris `package modulSatu;` dan `import java.util.Scanner;` sama seperti soal sebelumnya, dipakai untuk mengatur lokasi file dalam package serta mengimpor library Scanner agar bisa membaca input dari pengguna. Lalu, kelas utama didefinisikan dengan `public class PRAK104_2410817310014_IndraSuryadilaga {` dan titik awal eksekusi ada pada method `public static void main(String[] args).`

Pertama, program membaca pilihan tangan Abu dengan perintah `System.out.print("Tangan Abu: ");` `String[] tanganAbu = input.nextLine().split(" ");`. Hasil input akan dipecah berdasarkan spasi menggunakan `.split(" ")`, sehingga tiap pilihan (B, G, K) masuk ke dalam array `tanganAbu`. Proses yang sama dilakukan untuk Bagas pada baris

```
System.out.print("Tangan Bagas: "); String[] tanganBagas =  
input.nextLine().split(" ");
```

Setelah itu, dibuat variabel penghitung poinAbu dan poinBagas yang masing-masing awalnya bernilai 0. Kedua variabel ini akan bertambah sesuai dengan hasil setiap ronde.

Bagian inti program menggunakan perulangan for (int i = 0; i < 3; i++) karena ada 3 ronde yang harus dimainkan. Dalam setiap iterasi, pilihan Abu dan Bagas disimpan pada variabel pilihanAbu dan pilihanBagas. Kemudian dilakukan pengecekan kondisi sesuai aturan permainan dengan logika

```
((pilihanAbu.equals("B") && pilihanBagas.equals("G")) ||  
(pilihanAbu.equals("G") && pilihanBagas.equals("K")) ||  
(pilihanAbu.equals("K") && pilihanBagas.equals("B")))
```

Potongan ini berarti Abu mendapatkan poin ketika ia menang dari Bagas, misalnya Batu (B) lawan Gunting (G). Sedangkan kebalikannya, jika Bagas menang, maka poin Bagas bertambah melalui blok `else if`.

Setelah 3 ronde selesai, pemenang ditentukan dengan perbandingan nilai poinAbu dan poinBagas. Jika poinAbu > poinBagas, maka program mencetak "Abu", jika sebaliknya mencetak "Bagas", dan jika sama maka hasilnya "Seri".

SOAL 5

Buatlah program yang dapat menghitung volume tabung. Buatlah phi menjadi konstanta dengan aturan penulisan Bahasa pemrograman java.

Tabel 5.1 Soal No.5

Input	Output
Masukkan jari-jari: 11.4 Masukkan tinggi: 8.5	Volume tabung dengan jari-jari 11.4 cm dan tinggi 8.5 cm adalah 3468.632 m3
Masukkan jari-jari: 9.4 Masukkan tinggi: 5	Volume tabung dengan jari-jari 9.4 cm dan tinggi 5.0 cm adalah 1387.252 m3
Masukkan jari-jari: 2 Masukkan tinggi: 3	?

Simpan dengan nama file: **PRAK105-NIM-Nama.java**

A. Source Code

Tabel 5.2 Source Code

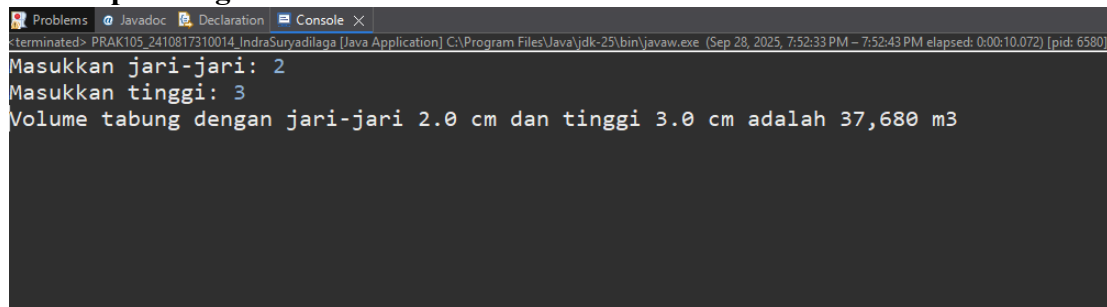
```
1 package modulSatu;
2 import java.util.Scanner;
3
4 public class PRAK105_2410817310014_IndraSuryadilaga {
5
6     public static void main(String[] args) {
7         Scanner input = new Scanner(System.in);
8
9         System.out.print("Masukkan jari-jari: ");
10        double jariJari = input.nextDouble();
11
12        System.out.print("Masukkan tinggi: ");
13        double tinggi = input.nextDouble();
14
15        double volume = jariJari * jariJari * tinggi
16 * 3.14;
17        String volumeFormat = String.format("%.3f",
18 volume);
19
20        System.out.print("Volume tabung dengan
21 jari-jari " + jariJari + " cm dan tinggi " + tinggi +
22 " cm adalah " + volumeFormat + " m3");
```

```

23
24         System.out.println();
25         input.close();
26     }
27 }

```

B. Output Program



The screenshot shows a Java IDE console window. The title bar includes 'Problems', 'Javadoc', 'Declaration', and 'Console'. The console output is as follows:

```

terminated> PRAK105_2410817310014_IndraSuryadilaga [Java Application] C:\Program Files\Java\jdk-25\bin\javaw.exe (Sep 28, 2025, 7:52:33 PM – 7:52:43 PM elapsed: 0:00:10.072) [pid: 6580]
Masukkan jari-jari: 2
Masukkan tinggi: 3
Volume tabung dengan jari-jari 2.0 cm dan tinggi 3.0 cm adalah 37,680 m3

```

Gambar 5. Screenshot Hasil Jawaban Soal 5

C. Pembahasan

Pada program ini, baris `package modulSatu;` dan `import java.util.Scanner;` sama seperti soal sebelumnya, dipakai untuk mengatur lokasi file dalam package serta mengimpor library Scanner agar bisa membaca input dari pengguna. Lalu, kelas utama didefinisikan dengan `public class PRAK105_2410817310014_IndraSuryadilaga {` dan titik awal eksekusi ada pada method `public static void main(String[] args).`

Input jari-jari dibaca dengan perintah `double jariJari = input.nextDouble();` Lalu dilanjutkan dengan membaca **tinggi** tabung menggunakan baris `double tinggi = input.nextDouble();`. Kedua variabel ini menggunakan tipe data `double` karena memungkinkan untuk menyimpan angka pecahan, sesuai dengan kebutuhan soal.

Rumus volume tabung dihitung pada baris `double volume = jariJari * jariJari * tinggi * 3.14;`. Di sini nilai π tidak ditulis sebagai variabel biasa, melainkan sebagai konstanta angka 3.14 yang langsung dipakai dalam perhitungan. Rumus tersebut mengikuti formula dasar volume tabung:

Hasil perhitungan kemudian diformat dengan tiga angka di belakang koma menggunakan `String volumeFormat = String.format("%.3f", volume);`. Terakhir, hasil ditampilkan dengan kalimat deskriptif melalui perintah `System.out.print("Volume tabung dengan jari-jari " + jariJari + " cm dan tinggi " + tinggi + " cm adalah " + volumeFormat + " m3");`. Sehingga output yang dihasilkan jelas menyebutkan nilai jari-jari, tinggi, dan volume tabung dalam satu kalimat yang lengkap.

GITHUB

https://github.com/IndraSuryadilaga/Pemrograman_II