**LAPORAN PRAKTIKUM**

**PEMROGRAMAN II**

**MODUL 1**

****

**STRUKTUR DASAR KODE JAVA**

**Oleh:**

**Daniel Noprianto NIM. 2410817110010**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**OKTOBER 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 : Daniel Noprianto

NIM : 2410817110010

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Galih Aji Sabdaraya  NIM. 2310817210005 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom.  NIP. 19930703 201903 1 011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 2](#_Toc210238477)

[DAFTAR ISI 3](#_Toc210238478)

[DAFTAR GAMBAR 5](#_Toc210238479)

[DAFTAR TABEL 6](#_Toc210238480)

[SOAL 1 7](#_Toc210238481)

[A. Source Code 7](#_Toc210238482)

[B. Output Program 10](#_Toc210238483)

[C. Pembahasan 11](#_Toc210238484)

[SOAL 2 14](#_Toc210238485)

[A. Source Code 14](#_Toc210238486)

[B. Output Program 15](#_Toc210238487)

[C. Pembahasan 15](#_Toc210238488)

[SOAL 3 18](#_Toc210238489)

[A. Source Code 18](#_Toc210238490)

[B. Output Program 19](#_Toc210238491)

[C. Pembahasan 19](#_Toc210238492)

[SOAL 4 21](#_Toc210238493)

[A. Source Code 21](#_Toc210238494)

[B. Output Program 23](#_Toc210238495)

[C. Pembahasan 23](#_Toc210238496)

[SOAL 5 26](#_Toc210238497)

[A. Source Code 26](#_Toc210238498)

[B. Output Program 27](#_Toc210238499)

[C. Pembahasan 27](#_Toc210238500)

[TAUTAN GIT HUB 30](#_Toc210238501)

# DAFTAR GAMBAR

[Gambar 1. Screenshot Hasil Jawaban Soal 1 10](#_Toc210238502)

[Gambar 2. Screenshot Hasil Jawaban Soal 2 15](#_Toc210238503)

[Gambar 3. Screenshot Hasil Jawaban Soal 3 19](#_Toc210238504)

[Gambar 4. Screenshot Hasil Jawaban Soal 4 23](#_Toc210238505)

[Gambar 5. Screenshot Hasil Jawaban Soal 5 27](#_Toc210238506)

# DAFTAR TABEL

[Tabel 1. 1 Soal No. 1 7](#_Toc210238713)

[Tabel 1. 2 Source Code Soal 1 7](#_Toc210238714)

[Tabel 1. 3 Soal No. 2 14](#_Toc210238715)

[Tabel 1. 4 Source Code Soal 2 14](#_Toc210238716)

[Tabel 1. 5 Soal No. 3 18](#_Toc210238717)

[Tabel 1. 6 Source Code Soal 3 18](#_Toc210238718)

[Tabel 1. 7 Soal No. 4 21](#_Toc210238719)

[Tabel 1. 8 Source Code Soal 4 21](#_Toc210238720)

[Tabel 1. 9 Soal No. 5 26](#_Toc210238721)

[Tabel 1. 10 Source Code Soal 5 26](#_Toc210238722)

# 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 |

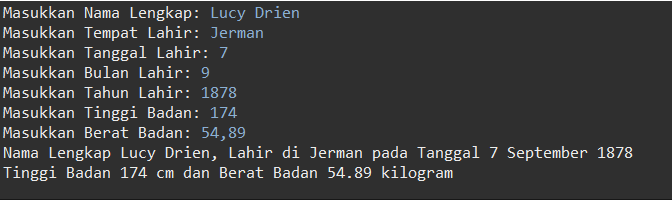
Simpan dengan nama file: **PRAK101-NIM-Nama.java**

## Source Code

*Tabel 1. 2 Source Code Soal 1*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106 | package modulSatu;  import java.util.Scanner;  public class PRAK101\_2410817110010\_DanielNoprianto {  public static void main(String[] args) {  Scanner scan = new Scanner(System.***in***);  //Input Nama  System.***out***.print("Masukkan Nama Lengkap: ");  String namaLengkap = scan.nextLine();  //Input Tempat Lahir  System.***out***.print("Masukkan Tempat Lahir: ");  String tempatLahir = scan.nextLine();  //Validasi untuk Tanggal, Bulan, dan Tahun  int tanggalLahir, bulanLahir, tahunLahir;  boolean tanggalValid = false;  do {  System.***out***.print("Masukkan Tanggal Lahir: ");  tanggalLahir = scan.nextInt();  System.***out***.print("Masukkan Bulan Lahir: ");  bulanLahir = scan.nextInt();  System.***out***.print("Masukkan Tahun Lahir: ");  tahunLahir = scan.nextInt();  //Validasi Bulan  if (bulanLahir < 1 || bulanLahir > 12) {  System.***out***.println("ERROR: Bulan harus antara 1 - 12. Silakan ulangi.");  continue;  }  //Cek Tahun  int tahunSekarang = 2025;  if (tahunLahir > tahunSekarang || tahunLahir <= 0) {  System.***out***.println("ERROR: Tahun lahir tidak valid. Silakan ulangi.");  continue;  }  //Cek Tanggal sesuai Bulan dan Tahun Kabisat  int maksTanggal;  if (bulanLahir == 2) {  //Cek tahun kabisat  boolean isKabisat = (tahunLahir % 4 == 0 && tahunLahir % 100 != 0) || (tahunLahir % 400 == 0);  maksTanggal = isKabisat ? 29 : 28;  } else if (bulanLahir == 4 || bulanLahir == 6 || bulanLahir == 9 || bulanLahir == 11) {  maksTanggal = 30;  } else {  maksTanggal = 31;  }  if (tanggalLahir >= 1 && tanggalLahir <= maksTanggal) {  tanggalValid = true;  } else {  System.***out***.println("ERROR: Tanggal " + tanggalLahir + " tidak valid untuk bulan " + bulanLahir + " tahun " + tahunLahir + ". Maksimal tanggal adalah " + maksTanggal + ".");  System.***out***.println("Silakan masukkan ulang tanggal, bulan, dan tahun.");  }  } while (!tanggalValid);  //Validasi Tinggi & Berat Badan ---  int tinggiBadan;  do {  System.***out***.print("Masukkan Tinggi Badan: ");  tinggiBadan = scan.nextInt();  if (tinggiBadan <= 0) {  System.***out***.println("Tinggi badan tidak boleh nol atau negatif. Silakan coba lagi.");  }  } while (tinggiBadan <= 0);  double beratBadan;  do {  System.***out***.print("Masukkan Berat Badan: ");  beratBadan = scan.nextDouble();  if (beratBadan <= 0) {  System.***out***.println("Berat badan tidak boleh nol atau negatif. Silakan coba lagi.");  }  } while (beratBadan <= 0);  scan.close();  String namaBulan;  switch (bulanLahir) {  case 1: namaBulan = "Januari"; break;  case 2: namaBulan = "Februari"; break;  case 3: namaBulan = "Maret"; break;  case 4: namaBulan = "April"; break;  case 5: namaBulan = "Mei"; break;  case 6: namaBulan = "Juni"; break;  case 7: namaBulan = "Juli"; break;  case 8: namaBulan = "Agustus"; break;  case 9: namaBulan = "September"; break;  case 10: namaBulan = "Oktober"; break;  case 11: namaBulan = "November"; break;  case 12: namaBulan = "Desember"; break;  default: namaBulan = ""; break;  }  System.***out***.println("Nama Lengkap " + namaLengkap + ", Lahir di " + tempatLahir + " pada Tanggal " + tanggalLahir + " " + namaBulan + " " + tahunLahir);  System.***out***.println("Tinggi Badan " + tinggiBadan + " cm dan Berat Badan " + beratBadan + " kilogram");  }  } |

## Output Program



Gambar 1. Screenshot Hasil Jawaban Soal 1

## Pembahasan

Program ini diawali dengan deklarasi package modulSatu yang menandakan bahwa file Java ini termasuk ke dalam paket bernama modulSatu. Dengan adanya penanda ini, ini nantinya akan membantu dalam pengorganisasian kode. Selanjutnya, terdapat import java.util.Scanner;, yang digunakan untuk mengimpor kelas *Scanner* dari pustaka Java. Kelas *Scanner* dipakai untuk membaca input dari pengguna.

Baris ini mendeklarasikan sebuah kelas publik yang memiliki nama PRAK101\_2410817110010\_DanielNoprianto. Nama pada kelas biasanya sesuai dengan nama file .java. Kelas ini nantinya akan menjadi tempat utama di mana seluruh kode program akan ditulis.

public static void main(String[] args) { Scanner scan = new Scanner(System.in);. Pada kelas ini terdapat method main, yaitu yang di mana titik awal eksekusi program. Program Java akan selalu memulai jalannya dari method ini. Kemudian objek *Scanner* *scan* dibuat agar bisa menerima input dari pengguna melalui System.in (melalui *keyboard*).

System.out.print("Masukkan Nama Lengkap: "); String namaLengkap = scan.nextLine(); dan System.out.print("Masukkan Tempat Lahir: "); String tempatLahir = scan.nextLine();. Pada bagian ini, program akan meminta kepada pengguna untuk bisa memasukkan Nama Lengkap dan Tempat Lahir. Method nextLine() dipakai pada program agar program bisa membaca input berupa string satu baris penuh (termasuk spasi).

int tanggalLahir, bulanLahir, tahunLahir; boolean tanggalValid = false;. Deklarasi tiga variabel tanggalLahir, bulanLahir, dan tahunLahir yang nantinya berfungsi untuk menyimpan data tanggal lahir. Variabel tanggalValid digunakan sebagai penanda apakah tanggal yang dimasukkan oleh pengguna sudah valid atau belum.

do {...} while (!tanggalValid);. Program akan menggunakan perulangan do-while untuk terus meminta input tanggal, bulan, dan tahun lahir hingga data yang dimasukkan sudah benar (valid). Perulangan ini nantinya akan selalu dieksekusi minimal satu kali, lalu akan dicek kondisi pada !tanggalValid (yang di mana selama tanggal belum valid, pengecekan akan terus diulang).

if (bulanLahir < 1 || bulanLahir > 12) {System.out.println("ERROR: Bulan harus antara 1 - 12. Silakan ulangi."); continue;}. Pada bagian ini, program akan mengecek apakah bulan yang dimasukkan berada dalam rentang bulan 1–12. Jika tidak, akan muncul pesan error dan continue dipakai untuk mengulang perulangan tanpa melanjutkan proses validasi lainnya.

int tahunSekarang = 2025; if (tahunLahir > tahunSekarang || tahunLahir <= 0) { System.out.println("ERROR: Tahun lahir tidak valid. Silakan ulangi."); continue;}. Pada bagian ini, program akan melakukan pengecekan terhadap tahun lahir. Tahun tidak boleh melebihi tahun sekarang (2025) dan tidak boleh <= nol atau negatif. Jika salah, pengguna nantinya akan diminta untuk mengulang input.

Selanjutnya ada validasi tanggal sesuai bulan & tahun kabisat. Program akan menentukan jumlah maksimal hari dalam bulan yang diinput. Untuk Februari (bulan 2), dicek apakah tahun tersebut kabisat. Jika kabisat, maksimal tanggal adalah 29, jika tidak maka 28. Untuk bulan dengan 30 hari (April, Juni, September, November), nilai maksTanggal = 30. Selain itu (bulan lain), maksTanggal = 31.

Selanjutnya ada pengecekan validitas tanggal, if (tanggalLahir >= 1 && tanggalLahir <= maksTanggal) {…} else {…}. Bagian ini memastikan bahwa tanggal yang dimasukkan sesuai dengan batas maksimal hari dalam bulan tersebut. Jika valid, variabel tanggalValid diubah menjadi true agar keluar dari perulangan. Jika tidak, program menampilkan pesan kesalahan dan meminta input ulang.

int tinggiBadan; do {…} while (tinggiBadan <= 0);. Pada bagian ini pengguna diminta untuk memasukkan tinggi badan. Adanya validasi dilakukan agar tinggi badan tidak boleh nol atau negatif. Jika salah, input akan diminta ulang.

double beratBadan; do {…} while (beratBadan <= 0);. Serupa dengan bagian tinggi badan, berat badan juga divalidasi agar nilainya positif. Input akan diulang sampai valid.

scan.close();. Setelah semua input yang diminta dari pengguna sudah selesai, objek Scanner akan ditutup agar tidak terjadi kebocoran *resource* (*resource leak*).

String namaBulan; switch (bulanLahir) { case 1: namaBulan = "Januari"; break; …default: namaBulan = ""; break; }. Bagian ini mengubah inputan angka bulan (1–12) menjadi nama bulan dalam bahasa Indonesia dengan menggunakan switch-case.

System.out.println(…);. Pada akhir dari program, program akan menampilkan atau mencetak informasi lengkap pengguna, termasuk nama, tempat dan tanggal lahir, serta tinggi dan berat badan. Dengan formatnya sudah disusun agar rapi dan mudah dibaca.

# 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 1. 3 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 | ? |

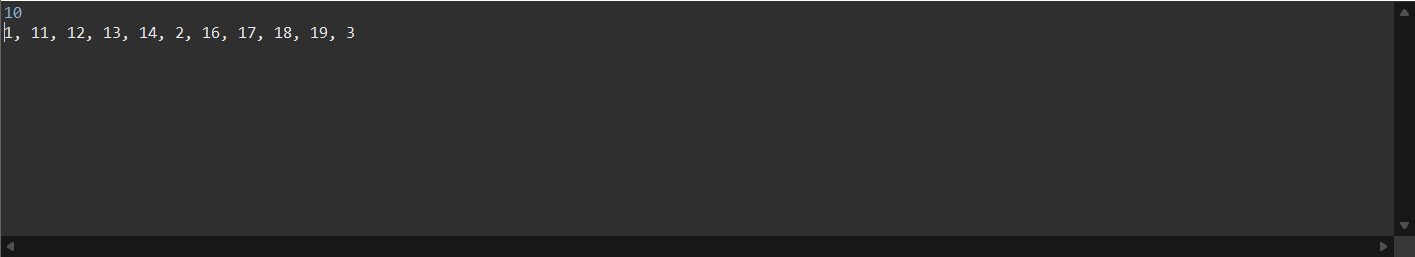
Simpan dengan nama file: **PRAK102-NIM-Nama.java**

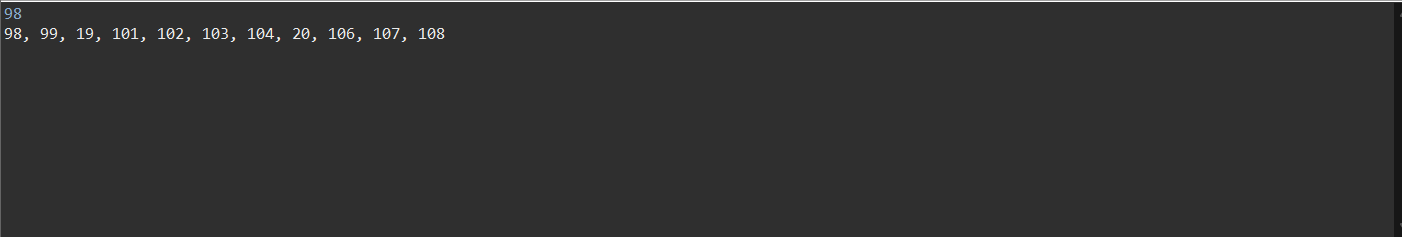
## Source Code

*Tabel 1. 4 Source Code Soal 2*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35 | package modulSatu;  import java.util.Scanner;  public class PRAK102\_2410817110010\_DanielNoprianto {  public static void main(String[] args) {  Scanner scan = new Scanner(System.***in***);    System.***out***.print("");  int angkaAwal = scan.nextInt();  scan.close();  System.***out***.println("");    int i = 0;  while (i <= 10) {  if (angkaAwal % 5 == 0) {  System.***out***.print(((angkaAwal / 5) - 1));  } else {  System.***out***.print(angkaAwal);  }  if (i < 10) {  System.***out***.print(", ");  }  angkaAwal++;  i++;  }  System.***out***.println();  }  } |

## Output Program







Gambar 2. Screenshot Hasil Jawaban Soal 2

## Pembahasan

Pertama - pertama ada package modulSatu; import java.util.Scanner;. Program ditempatkan dalam package bernama modulSatu agar lebih terstruktur. Kemudian ada perintah import java.util.Scanner; yang digunakan agar kelas Scanner bisa dipakai untuk membaca input dari pengguna melalui keyboard.

public class PRAK102\_2410817110010\_DanielNoprianto {…} . Baris ini akan mendeklarasikan sebuah kelas publik dengan nama PRAK102\_2410817110010\_DanielNoprianto. Nama kelas harus sesuai dengan nama file .java yang digunakan.

public static void main(String[] args) { Scanner scan = new Scanner(System.in);. Di dalam kelas, method main didefinisikan sebagai titik awal eksekusi program. Kemudian objek Scanner dibuat agar bisa membaca input dari pengguna.

System.out.print(""); int angkaAwal = scan.nextInt(); scan.close(); System.out.print("");. Program menyiapkan perintah kosong (System.out.print("")) lalu membaca sebuah bilangan integer yang dimasukkan oleh pengguna menggunakan nextInt(), dan menyimpannya dalam variabel angkaAwal. Setelah itu, *scanner* ditutup dengan scan.close() untuk menghindari kebocoran *resource*. Ada satu lagi yaitu System.out.print("") setelah *scanner* akan ditutup, tetapi tidak menampilkan teks apa pun hanya untuk *statement* kosong yang sebenarnya tidak terlalu berpengaruh.

int i = 0; while (i <= 10) {…}. Program akan mendeklarasikan variabel penghitung i dengan nilai awal 0. Kemudian dilakukan perulangan while selama i <= 10, artinya perulangan akan berjalan sebanyak 11 kali (mulai dari i = 0 sampai 10).

if (angkaAwal % 5 == 0) {…} else {…}. Di dalam perulangan, program mengecek apakah angkaAwal merupakan kelipatan 5 dengan angkaAwal % 5 == 0. Jika iya, maka yang dicetak adalah (angkaAwal / 5) - 1. Artinya nilai dibagi 5, kemudian dikurangi oleh 1. Jika tidak, maka langsung mencetak nilai asli dari angkaAwal. Jadi, output akan berbeda tergantung apakah kondisi bilangan tersebut kelipatan 5 atau bukan.

if (i < 10) { System.out.print(", "); }. Setelah mencetak angka, program akan menambahkan tanda koma dan spasi (, ) selama i masih lebih kecil dari 10. Hal ini untuk memastikan bahwa angka terakhir (saat i = 10) tidak diikuti koma lagi, agar output sesuai dengan yang diminta.

angkaAwal++; i++;. Pada akhir dari setiap iterasi perulangan, nilai angkaAwal dinaikkan 1 (++) agar di langkah berikutnya bisa memproses angka selanjutnya. Demikian juga, i ditambah 1 agar perulangan bergerak maju menuju kondisi yang telah ditentukan.

System.out.println();. Setelah perulangan selesai, program mencetak baris baru agar kursor pindah ke bawah.

# 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 1. 5 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**

## Source Code

*Tabel 1. 6 Source Code Soal 3*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | package modulSatu;  import java.util.Scanner;  public class PRAK103\_2410817110010\_DanielNoprianto {  public static void main(String[] args) {  Scanner scan = new Scanner(System.***in***);  System.***out***.print("");  int jumlahBaris = scan.nextInt();  int angka = scan.nextInt();  scan.close();  int i = 0;    do {  if (angka % 2 != 0) {  System.***out***.print(angka);  i++;  if (i < jumlahBaris) {  System.***out***.print(", ");  }  }  angka++;  } while (i < jumlahBaris);  System.***out***.println();  }  } |

## Output Program





**

Gambar 3. Screenshot Hasil Jawaban Soal 3

## Pembahasan

Pada awal program terdapat package modulSatu; dan import java.util.Scanner;. Program berada di dalam package modulSatu untuk pengelompokan kode. Import java.util.Scanner digunakan supaya kelas Scanner bisa dipakai untuk menerima input.

public class PRAK103\_2410817110010\_DanielNoprianto {…}. Baris ini mendeklarasikan kelas publik bernama PRAK103\_2410817110010\_DanielNoprianto. Nama kelas mengikuti nama file .java.

public static void main(String[] args) { Scanner scan = new Scanner(System.in);. Program akan dimulai dari method main. *Scanner* dibuat agar bisa membaca input dari pengguna nantinya.

System.out.print(""); int jumlahBaris = scan.nextInt(); int angka = scan.nextInt(); scan.close();. Program pertama kali menampilkan perintah kosong. Lalu pengguna diminta memasukkan dua bilangan *integer*, variabel jumlahBaris akan berisi banyaknya bilangan yang ingin ditampilkan dan variabel angka berisi bilangan awal yang akan diproses. Setelah itu, objek *Scanner* ditutup dengan scan.close().

int i = 0;. Sebuah variabel penghitung i disiapkan dengan nilai awal 0. Variabel ini akan digunakan untuk menghitung berapa banyak bilangan ganjil yang sudah ditampilkan.

do { ... } while (i < jumlahBaris);. Program menggunakan perulangan do-while, yang berarti blok kode di dalamnya akan selalu dijalankan minimal sekali, kemudian dicek kondisi di bagian while. Perulangan ini berhenti jika i sudah sama dengan jumlahBaris.

if (angka % 2 != 0) {…} angka++;, yaitu pengecekan untuk bilangan ganjil. if (angka % 2 != 0) mengecek apakah bilangan saat ini ganjil. Jika ganjil, bilangan tersebut akan dicetak (System.out.print(angka);) dan variabel penghitung i ditambah 1. Setelah mencetak, jika i masih kurang dari jumlahBaris, maka program menambahkan tanda koma (,). Terakhir, angka++ akan menambah nilai angka satu per satu agar bisa mengecek bilangan berikutnya.

System.out.println();. Setelah perulangan selesai, program mencetak baris baru agar kursor pindah ke bawah.

# SOAL 4

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

1. Batu (B) mengalahkan Gunting (G).
2. Gunting (G) mengalahkan Kertas(K).
3. Kertas (K) mengalahkan Batu (B).
4. Jika kedua pemain memilih tangan yang sama, hasilnya adalah seri.
5. Terdapat tiga ronde dalam sekali permainan. Pemain melakukan suit sebanyak tiga kali.
6. Setiap ronde akan ditentukan pemenangnya berdasarkan aturan yang telah ditentukan sebelumnya.
7. Pemain yang mendapatkan poin terbanyak dialah pemenangnya.

Bantulah Abu dan Bagas 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 Bagas tiap rondenya. Hasilnya antara “Abu”, “Bagas” atau “Seri”.

Tabel 1. 7 Soal No. 4

|  |  |
| --- | --- |
| **Input** | **Output** |
| Tangan Abu: G G K  Tangan Bagas: B K G | Bagas |
| Tangan Abu: G K G  Tangan Bagas: B K K | Seri |
| Tangan Abu: G K B  Tangan Bagas: B K K | ? |

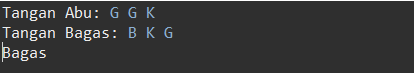
Simpan dengan nama file: **PRAK104-NIM-Nama.java**

## Source Code

*Tabel 1. 8 Source Code Soal 4*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47 | package modulSatu;  import java.util.Scanner;  public class PRAK104\_2410817110010\_DanielNoprianto {  public static void main(String[] args) {  Scanner scan = new Scanner(System.***in***);  System.***out***.print("Tangan Abu: ");  String inputAbu = scan.nextLine();  String tanganAbu = inputAbu.replaceAll("\\s+", "");  System.***out***.print("Tangan Bagas: ");  String inputBagas = scan.nextLine();  String tanganBagas = inputBagas.replaceAll("\\s+", "");  scan.close();  if (tanganAbu.length() != 3 || tanganBagas.length() != 3) {  System.***out***.println("\nInput tidak valid! Masing-masing pemain harus memasukkan 3 gerakan (B/G/K).");  } else {    int skorAbu = 0;  int skorBagas = 0;  for (int i = 0; i < 3; i++) {  char abuGerakan = tanganAbu.charAt(i);  char bagasGerakan = tanganBagas.charAt(i);  if ((abuGerakan == 'B' && bagasGerakan == 'G') ||  (abuGerakan == 'G' && bagasGerakan == 'K') ||  (abuGerakan == 'K' && bagasGerakan == 'B')) {  skorAbu++;  } else if (abuGerakan != bagasGerakan) {  skorBagas++;  }  }    if (skorAbu > skorBagas) {  System.***out***.println("Abu");  } else if (skorBagas > skorAbu) {  System.***out***.println("Bagas");  } else {  System.***out***.println("Seri");  }  }  }  } |

## Output Program







Gambar 4. Screenshot Hasil Jawaban Soal 4

## Pembahasan

package modulSatu; import java.util.Scanner;. Merupakan bagian awal program, yaitu sebagai pemberitahu bahwa file ditempatkan dalam package modulSatu agar lebih terstruktur. Import java.util.Scanner digunakan supaya kelas Scanner bisa dipakai untuk membaca input dari pengguna.

public class PRAK104\_2410817110010\_DanielNoprianto {…}. Program ini ada dalam kelas publik PRAK104\_2410817110010\_DanielNoprianto. Nama kelas biasanya sesuai dengan nama file .java.

public static void main(String[] args) {Scanner scan = new Scanner(System.in);…}. Program dimulai dari method main, titik eksekusi pertama saat program dijalankan. Objek Scanner dibuat dalam program agar program bisa menerima input dari pengguna.

System.out.print("Tangan Abu: "); String inputAbu = scan.nextLine(); String tanganAbu = inputAbu.replaceAll("\\s+", "");. Program akan meminta input gerakan Abu. Input bisa mengandung spasi, misalnya B G K. Oleh karena itu, method replaceAll("\\s+", "") digunakan untuk menghapus semua spasi, sehingga tersisa *string* murni tiga karakter seperti "BGK".

System.out.print("Tangan Bagas: "); String inputBagas = scan.nextLine(); String tanganBagas = inputBagas.replaceAll("\\s+", "");. Serupa dengan Abu, program meminta input untuk Bagas dan juga menghapus spasi menggunakan replaceAll("\\s+", "").

scan.close();. *Scanner* ditutup untuk mencegah kebocoran *resource*, karena input sudah selesai diterima.

if (tanganAbu.length() != 3 || tanganBagas.length() != 3) {System.out.println("\nInput tidak valid! Masing-masing pemain harus memasukkan 3 gerakan (B/G/K).");}. Program akan mengecek apakah *string* hasil input dari Abu dan Bagas berisi tepat 3 karakter. Jika tidak, maka dianggap tidak valid dan program menampilkan pesan *error*.

else {int skorAbu = 0; int skorBagas = 0;}. Jika input valid, maka skor awal Abu dan Bagas diinisialisasi menjadi nol. Skor ini nantinya akan dihitung berdasarkan hasil 3 ronde permainan.

for (int i = 0; i < 3; i++) { char abuGerakan = tanganAbu.charAt(i); char bagasGerakan = tanganBagas.charAt(i);}. Program akan menggunakan perulangan for dari i = 0 sampai i < 3 untuk membandingkan ketiga Gerakan yang telah diinputkan. Pada setiap ronde, charAt(i) digunakan untuk mengambil karakter gerakan Abu dan Bagas di ronde ke-i. Simbol B = Batu, Simbol G = Gunting, Simbol K = Kertas

if ((abuGerakan == 'B' && bagasGerakan == 'G') || (abuGerakan == 'G' && bagasGerakan == 'K') || (abuGerakan == 'K' && bagasGerakan == 'B')) {skorAbu++; } else if (abuGerakan != bagasGerakan) {skorBagas++;}. Ini Adalah logika untuk mengikuti aturan permainan Batu Gunting Kertas. Jika Abu menang ronde, maka skorAbu++. Jika gerakan Abu dan Bagas berbeda tetapi Abu tidak menang, maka otomatis Bagas yang menang (skorBagas++). Jika keduanya sama, ronde dianggap seri dan skor tidak berubah.

if (skorAbu > skorBagas) {System.out.println("Abu");} else if (skorBagas > skorAbu) {System.out.println("Bagas");} else {System.out.println("Seri");}. Setelah 3 ronde dimainkan sudah selesai, skor akan dibandingkan. Jika skor Abu lebih besar maka Abu menang, jika skor Bagas lebih besar maka Bagas menang, dan jika sama maka hasil akhir akan seri.

# SOAL 5

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

Tabel 1. 9 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 cm3 |
| Masukkan jari-jari: 9.4  Masukkan tinggi: 5 | Volume tabung dengan jari-jari 9.4 cm dan tinggi 5.0 cm adalah 1387.252 cm3 |
| Masukkan jari-jari: 2  Masukkan tinggi: 3 | ? |

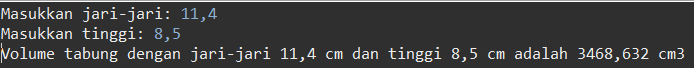
Simpan dengan nama file: **PRAK105-NIM-Nama.java**

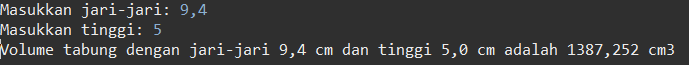
## Source Code

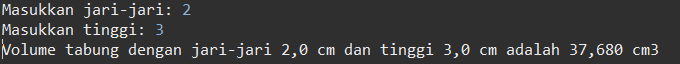
*Tabel 1. 10 Source Code Soal 5*

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42 | package modulSatu;  import java.util.Scanner;  public class PRAK105\_2410817110010\_DanielNoprianto {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.***in***);  double jariJari, tinggi;  final double PI = 3.14;  do {  System.***out***.print("Masukkan jari-jari: ");  jariJari = scanner.nextDouble();  if (jariJari <= 0) {  System.***out***.println("Input tidak valid. Jari-jari harus lebih besar dari 0.");  }  } while (jariJari <= 0);  do {  System.***out***.print("Masukkan tinggi: ");  tinggi = scanner.nextDouble();  if (tinggi <= 0) {  System.***out***.println("Input tidak valid. Tinggi harus lebih besar dari 0.");  }  } while (tinggi <= 0);  scanner.close();  double volume = PI \* jariJari \* jariJari \* tinggi;  System.***out***.printf(  "Volume tabung dengan jari-jari %.1f cm dan tinggi %.1f cm adalah %.3f cm3\n",  jariJari, tinggi, volume  );  }  } |

## Output Program







Gambar 5. Screenshot Hasil Jawaban Soal 5

## Pembahasan

Program ini akan diawali dengan mendeklarasikan package modulSatu dan mengimpor *library* Scanner dari Java. Package digunakan untuk mengelompokkan kelas agar lebih terstruktur, sedangkan *library* Scanner diperlukan untuk membaca input dari pengguna melalui keyboard. Setelah itu, kelas utama PRAK105\_2410817110010\_DanielNoprianto didefinisikan, yang berisi method main sebagai titik awal eksekusi dari program.

Di dalam method main, objek Scanner dibuat dengan perintah Scanner scanner = new Scanner(System.in);. Objek ini berfungsi untuk menangkap data yang dimasukkan oleh pengguna. Selanjutnya, aka nada deklarasi dua buah variabel bertipe double yaitu jariJari dan tinggi yang akan menyimpan input dari pengguna. Selain itu, didefinisikan juga konstanta PI dengan nilai 3.14 menggunakan keyword final, sehingga nilai PI tidak dapat diubah selama program berjalan.

Langkah berikutnya adalah proses validasi input untuk jari-jari tabung. Program menggunakan struktur perulangan do-while, program akan meminta pengguna untuk memasukkan data minimal sekali. Di dalam perulangan juga, program meminta pengguna untuk memasukkan nilai jari-jari. Jika pengguna memberikan nilai yang kurang dari atau sama dengan nol, program menampilkan pesan error “Input tidak valid. Jari-jari harus lebih besar dari 0.” dan mengulang kembali untuk perintah input. Perulangan ini terus berjalan sampai pengguna memasukkan nilai yang benar, yaitu lebih besar dari nol.

Setelah jari-jari berhasil divalidasi, program melanjutkan ke validasi tinggi tabung. Cara kerja yang digunakan sama dengan validasi jari-jari, yaitu menggunakan do-while. Program akan meminta pengguna memasukkan nilai untuk tinggi, kemudian memeriksa apakah nilai tersebut lebih besar dari nol. Jika tidak valid, maka akan muncul pesan error “Input tidak valid. Tinggi harus lebih besar dari 0.” dan pengguna diminta untuk mengulang input hingga memasukkan nilai yang benar.

Jika kedua input sudah valid, objek Scanner ditutup dengan perintah scanner.close(); untuk mencegah kebocoran *resource*. Selanjutnya, program melakukan perhitungan volume pada tabung dengan rumus π × r² × t. Rumus ini diimplementasikan dalam kode double volume = PI \* jariJari \* jariJari \* tinggi;. Perhitungan ini memanfaatkan konstanta PI serta nilai jariJari dan tinggi yang dimasukkan pengguna sebelumnya.

Hasil perhitungan kemudian ditampilkan ke layar menggunakan System.out.printf(). Fungsi ini memungkinkan hasil ditampilkan dengan format angka tertentu. Pada program, jari-jari dan tinggi ditampilkan dengan satu angka di belakang koma menggunakan format %.1f, sementara volume ditampilkan dengan tiga angka di belakang koma menggunakan format %.3f.

# TAUTAN GIT HUB

<https://github.com/Daniel-N0/Pemrograman_II/tree/3052f8b71640c59da891ac11c5569a3d7f436031/modulSatu>