

**LAPORAN PRAKTIKUM
PEMROGRAMAN II
MODUL 1**



STRUKTUR DASAR KODE JAVA

Oleh:

Ryan Muhammad Irfan NIM. 2210817310013

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
SEPTEMBER 2023**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN II
MODUL 1

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

Nama Praktikan : Ryan Muhammad Irfan
NIM : 2210817310013

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Bachrul Uluum
NIM. 2010817210025

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.....	7
C. Pembahasan.....	7
D. Tautan GIT.....	12
SOAL 2.....	13
A. Source Code	13
B. Output Program.....	14
C. Pembahasan.....	14
D. Tautan GIT.....	16
SOAL 3.....	17
A. Source Code	17
B. Output Program.....	18
C. Pembahasan.....	18
D. Tautan GIT.....	20
SOAL 4.....	21
A. Source Code	21
B. Output Program.....	23
C. Pembahasan.....	23
D. Tautan GIT.....	29
SOAL 5.....	30
A. Source Code	30
B. Output Program.....	31
C. Pembahasan.....	31
D. Tautan GIT.....	33

DAFTAR GAMBAR

Gambar 1. Screenshot Hasil Jawaban Soal 1	7
Gambar 2. Screenshot Hasil Jawaban Soal 2	14
Gambar 3. Screenshot Hasil Jawaban Soal 3	18
Gambar 4. Screenshot Hasil Jawaban Soal 4	23
Gambar 5. Screenshot Hasil Jawaban Soal 5	31

DAFTAR TABEL

Tabel 1. Soal 1	6
Tabel 2. Source Code Soal 1	7
Tabel 3. Soal 2	13
Tabel 4. Source Code Soal 2	13
Tabel 5. Soal 3	17
Tabel 6. Source Code Soal 3	17
Tabel 7. Soal 4	21
Tabel 8. Source Code Soal 4	22
Tabel 9. Soal 5	30
Tabel 10. Source Code Soal 5	30

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.

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

Tabel 1. Soal 1

Simpan coding anda dengan nama: **PRAK101-NIM-Nama.java**

A. Source Code

1	import java.util.Scanner;
2	
3	public class PRAK101_2210817310013_RyanMuhammadIrfan {
4	public static void main(String[] args) {
5	boolean inputvalid = false;
6	String month_name_selected = "";
7	
8	String [] month_name = {
9	"Januari", "Februari", "Maret", "April",
10	"Mei", "Juni", "Juli", "Agustus", "September",
11	"Oktober", "November", "Desember"
12	};
13	
14	Scanner userinput = new Scanner(System.in);
15	System.out.print("Masukkan Nama Lengkap: ");
16	String name = userinput.nextLine();
17	
18	System.out.print("Masukkan Tempat Lahir: ");
19	String place_of_birth = userinput.nextLine();

```

20
21     System.out.print("Masukkan Tanggal Lahir: ");
22     int date_of_birth = userInput.nextInt();
23
24     while (!inputvalid){
25         System.out.print("Masukkan Bulan Lahir: ");
26         int month_of_birth = userInput.nextInt();
27         if (month_of_birth >=1 && month_of_birth <=12){
28             inputvalid = true;
29             month_name_selected =
month_name[month_of_birth - 1];
30         }
31     }
32
33     System.out.print("Masukkan Tahun Lahir: ");
34     int year_of_birth = userInput.nextInt();
35
36     System.out.print("Masukkan Tinggi Badan: ");
37     int height = userInput.nextInt();
38
39     System.out.print("Masukkan Berat Badan: ");
40     float weight = userInput.nextFloat();
41
42     System.out.println("Nama Lengkap "+name+", Lahir di
"+place_of_birth+
43         "      pada      Tanggal      "+date_of_birth+"
"+month_name_selected+
44         " "+year_of_birth+" Tinggi Badan "+height+
45         " cm dan Berat Badan "+weight+" kilogram");
46 }
47 }

```

Tabel 2. Source Code Soal 1

B. Output Program

```

Run PRAK101_2210817310013_RyanMuhammadIrfan x
"C:\Users\R\yan muhammad irfan\.jdk\openjdk-20.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023
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
Process finished with exit code 0

```

Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Pada baris [1], import java.util.Scanner;

Mengimpor class scanner yang terdapat di dalam package java.util. Berfungsi untuk membaca input dari pengguna.

Pada baris [3-54],

```
public class PRAK101_2210817310013_RyanMuhammadIrfan "public"
```

berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

"class" berfungsi untuk membuat class yang dalam baris ini diberi nama PRAK101_2210817310013_RyanMuhammadIrfan.

"{"...}" berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [4-53],

```
public static void main(String[] args)
```

"public" berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

"static" berfungsi membuat suatu method tanpa perlu melakukan instansiasi terlebih dahulu.

"void" berfungsi untuk tidak mengembalikan nilai apapun.

"main" merupakan nama fungsi yang digunakan oleh java sebagai awal masuk ke program.

"String[] args" berfungsi sebagai parameter yang diperlukan oleh fungsi main. Parameter ini adalah array dari argument perintah yang bisa diteruskan ke program java.

"{"...}" berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [5],

```
boolean inputvalid = false;
```

"boolean" merupakan tipe datanya, "inputvalid" merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya false. <tipe data> <nama variabel> = <nilai variabel>.

"," berfungsi untuk mengakhiri sebuah perintah.

Pada baris [6],

```
String month_name_selected = "";
```

"String" merupakan tipe datanya, "month_name_selected" merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya "kosong". <tipe data> <nama variabel> = <nilai variabel>.

"," berfungsi untuk mengakhiri sebuah perintah.

Pada baris [8-12],

```
String [] month_name = {
```

"String" merupakan tipe datanya, "month_name" merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu diisi valuenya array. <tipe data> <nama variabel> = <nilai variabel>.

"[]" berfungsi untuk mendeklarasikan array.

"{"...}" berfungsi untuk memulai dan mengakhiri blok kode.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [9], "Januari", "Februari", "Maret", "April",
Merupakan isi dari value array pada variabel baris ke 8.

Pada baris [10], "Mei", "Juni", "Juli", "Agustus", "September",
Merupakan isi dari value array pada variabel baris ke 8.

Pada baris [11], "Oktober", "November", "Desember"
Merupakan isi dari value array pada variabel baris ke 8.

Pada baris [14], `Scanner userInput = new Scanner(System.in);`
Berfungsi untuk membuat sebuah objek dari class Scanner atau dalam hal ini variabel userInput akan berisi object dari Scanner class dengan cara memanggil konstruktor `Scanner(System.in)`. `System in` adalah input stream yang digunakan untuk menerima input dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [15], `System.out.print("Masukkan Nama Lengkap: ");`
“`System.out.print(...)`” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Nama Lengkap: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [16], `String name = userInput.nextLine();`
“String” merupakan tipe datanya, “name” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userinput.nextline()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“`nextLine()`” berfungsi untuk membaca nilai String dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [18], `System.out.print("Masukkan Tempat Lahir: ");`
“`System.out.print(...)`” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Tempat Lahir: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [19], `String place_of_birth = userInput.nextLine();`
“String” merupakan tipe datanya, “place_of_birth” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userinput.nextline()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextLine()” berfungsi untuk membaca nilai String dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [21], `System.out.print("Masukkan Tanggal Lahir: ");`
“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Tanggal Lahir: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [22], `int date_of_birth = userInput.nextInt();`
“int” merupakan tipe datanya, “date_of_birth” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.nextInt()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextInt()” berfungsi untuk membaca nilai Int dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [24-31], `while (!inputvalid) {...}`
“while” merupakan sebuah perulangan yang mengulang sebuah blok kode selama kondisi yang ditentukan benar. “!inputvalid” merupakan sebuah kondisi. Dalam hal ini berarti kondisinya tidak false(value dari variabel inputvalid) artinya perulangan akan terus berjalan sampai kondisinya terpenuhi (true).

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [25], `System.out.print("Masukkan Bulan Lahir: ");`
“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Bulan Lahir: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [26], `int month_of_birth = userInput.nextInt();`
“int” merupakan tipe datanya, “month_of_birth” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.nextInt()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextInt()” berfungsi untuk membaca nilai Int dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [27-30], `if (month_of_birth >=1 && month_of_birth <=12) {...}`
“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara

kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “month_of_birth >=1 && month_of_birth <=12” merupakan kondisi yang artinya value dari month_of_birth lebih dari sama dengan 1 dan kurang dari sama dengan 12.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [28], `inputvalid = true;`

Menyatakan value dari variabel “inputvalid” bernilai true yang berfungsi untuk menghentikan perulangan dari while.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [29], `month_name_selected = month_name[month_of_birth - 1];`

Menyatakan value dari variabel “month_name_selected” bernilai array dari month_name

“[month_of_birth - 1]” berfungsi untuk menentukan indeks array. Merupakan value dari variabel month_of_birth dikurang 1 karena array dimulai dari indeks 0.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [33], `System.out.print("Masukkan Tahun Lahir: ");`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Tahun Lahir: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [34], `int year_of_birth = userInput.nextInt();`

“int” merupakan tipe datanya, “year_of_birth” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.nextInt()`. <tipe data> <nama variabel> = <nilai variabel>.

“userInput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextInt()” berfungsi untuk membaca nilai Int dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [36], `System.out.print("Masukkan Tinggi Badan: ");`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Tinggi Badan: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [37], `int height = userInput.nextInt();`

“int” merupakan tipe datanya, “height” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.nextInt()`. <tipe data> <nama variabel> = <nilai variabel>.

Pada baris [39], `System.out.print("Masukkan Berat Badan: ");`

“System.out.print(…)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan Berat Badan: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [40], `float weight = userinput.nextFloat();`

“float” merupakan tipe datanya, “weight” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userinput.nextFloat()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextFloat()” berfungsi untuk membaca nilai float dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [42-45],

```
System.out.println("Nama    Lengkap    "+name+",    Lahir    di  
"+place_of_birth+  
"    pada Tanggal "+date_of_birth+" "+month_name_selected+  
" "+year_of_birth+" Tinggi Badan "+height+  
" cm dan Berat Badan "+weight+" kilogram");
```

“System.out.println(…)” berfungsi untuk mencetak angka atau karakter dengan pindah ke baris berikutnya. Dalam baris ini mencetak "Nama Lengkap "+ value dari variabel name+", Lahir di "+ value dari variabel place_of_birth+ " pada Tanggal "+ value dari variabel date_of_birth+" "+ value dari variabel month_name_selected+ " "+ value dari variabel year_of_birth+" Tinggi Badan "+ value dari variabel height+ " cm dan Berat Badan "+ value dari variabel weight+" kilogram".

“;” berfungsi untuk mengakhiri sebuah perintah.

D. Tautan GIT

https://github.com/ryanmi04/Praktikum-Pemrograman-2-Paralel-1/blob/main/PRAKTIKUM%201/src/PRAK101_2210817310013_RyanMuhammadIrfan.java

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.

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	?

Tabel 3. Soal 2

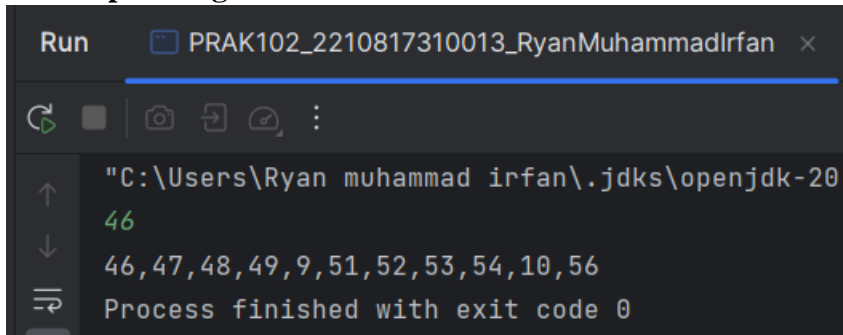
Simpan coding anda dengan nama: **PRAK102-NIM-Nama.java**

A. Source Code

```
1  import java.util.Scanner;
2
3  public class PRAK102_2210817310013_RyanMuhammadIrfan {
4      public static void main(String[] args) {
5
6          int i = 1;
7          Scanner userInput = new Scanner(System.in);
8
9          int number = userInput.nextInt();
10         while (i<=11){
11             if (number % 5 == 0){
12                 System.out.print(number / 5 - 1);
13             }
14             else {
15                 System.out.print(number);
16             }
17             if (i<11){
18                 System.out.print(",");
19             }
20             i++;
21             number++;
22         }
23     }
24 }
```

Tabel 4. Source Code Soal 2

B. Output Program



```
Run PRAK102_2210817310013_RyanMuhammadIrfan x
C:\Users\Ryan muhammad irfan\.jdk\openjdk-20.
46
46, 47, 48, 49, 9, 51, 52, 53, 54, 10, 56
Process finished with exit code 0
```

Gambar 2. Screenshot Hasil Jawaban Soal 2

C. Pembahasan

Pada baris [1], `import java.util.Scanner;`

Mengimpor class scanner yang terdapat di dalam package `java.util`. Berfungsi untuk membaca input dari pengguna.

Pada baris [3-24],

`public class PRAK102_2210817310013_RyanMuhammadIrfan` “public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“class” berfungsi untuk membuat class yang dalam baris ini diberi nama `PRAK102_2210817310013_RyanMuhammadIrfan`.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [4-23], `public static void main(String[] args)`

“public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“static” berfungsi membuat suatu method tanpa perlu melakukan instansiasi terlebih dahulu.

“void” berfungsi untuk tidak mengembalikan nilai apapun.

“main” merupakan nama fungsi yang digunakan oleh java sebagai awal masuk ke program.

“String[] args” berfungsi sebagai parameter yang diperlukan oleh fungsi main. Parameter ini adalah array dari argument perintah yang bisa diteruskan ke program java.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [6], `int i = 1;`

“int” merupakan tipe datanya, “i” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya 1. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [7], `Scanner userInput = new Scanner(System.in);`
Berfungsi untuk membuat sebuah objek dari class `Scanner` atau dalam hal ini variabel `userInput` akan berisi object dari `Scanner` class dengan cara memanggil konstruktor `Scanner(System.in)`. `System.in` adalah input stream yang digunakan untuk menerima input dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah..

Pada baris [9], `int number = userInput.nextInt();`

“int” merupakan tipe datanya, “number” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.nextInt()`. <tipe data> <nama variabel> = <nilai variabel>.

“userInput” merupakan object dari class `scanner` yang berfungsi mengambil input dari pengguna.

“nextInt()” berfungsi untuk membaca nilai `Int` dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [10-22], `while (i<=11) {...}`

“while” merupakan sebuah perulangan yang mengulang sebuah blok kode selama kondisi yang ditentukan benar. “i<=11” merupakan sebuah kondisi. Dalam hal ini berarti kondisinya i<=11 apabila lewat dari kondisi tersebut maka perulangan akan berhenti.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [11-13], `if (number % 5 == 0) {...}`

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan `else if` atau `else`. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “number % 5 == 0” merupakan kondisi yang artinya value dari number modulus 5 sama dengan 0.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [12], `System.out.print(number / 5 - 1);`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak hasil dari nilai value number dibagi 5 dikurang 1.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [14-16], `else{...}`

“else” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya dari if sebelumnya salah/tidak terpenuhi.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [15], `System.out.print(number);`

“System.out.print(…)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak hasil dari nilai value number.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [17-19], `if (i<11) {...}`

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. Cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “i<11” merupakan kondisi yang artinya value dari i<11.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [18], `System.out.print(“,”);`

“System.out.print(…)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak “,”.

Pada baris [20], `i++;`

“i++” berfungsi sebagai increment yang digunakan untuk menambah value i 1 angka. Dalam hal ini digunakan untuk mengerjakan iterasi loop menuju kondisi berhenti.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [21], `number++;`

“number++” berfungsi sebagai increment yang digunakan untuk menambah value number 1 angka. Dalam hal ini digunakan untuk mengerjakan iterasi loop menuju kondisi berhenti.

“;” berfungsi untuk mengakhiri sebuah perintah.

D. Tautan GIT

https://github.com/ryanmi04/Praktikum-Pemrograman-2-Paralel-1/blob/main/PRAKTIKUM%201/src/PRAK102_2210817310013_RyanMuhammadIrfan.java

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

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

Tabel 5. Soal 3

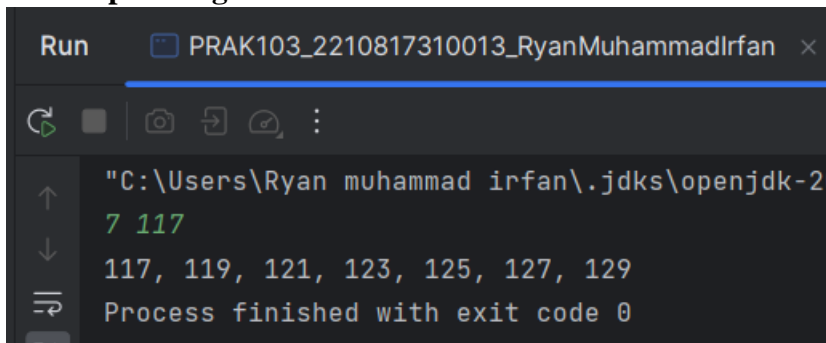
Simpan coding anda dengan nama: **PRAK103-NIM-Nama.java**

A. Source Code

```
1  import java.util.Scanner;
2
3  public class PRAK103_2210817310013_RyanMuhammadIrfan {
4      public static void main(String[] args) {
5          int i = 1;
6          Scanner userInput= new Scanner(System.in);
7          int number1 = userInput.nextInt();
8          int number2 = userInput.nextInt();
9
10         do {
11             if (number2 % 2 !=0){
12                 System.out.print(number2);
13                 if (i<number1){
14                     System.out.print(", ");
15                 }
16                 i++;
17             }
18             number2++;
19         } while (i <= number1);
20     }
21 }
```

Tabel 6. Source Code Soal 3

B. Output Program



```
Run PRAK103_2210817310013_RyanMuhammadIrfan x
C:\Users\Ryan muhammad irfan\.jdk\openjdk-20
7 117
117, 119, 121, 123, 125, 127, 129
Process finished with exit code 0
```

Gambar 3. Screenshot Hasil Jawaban Soal 3

C. Pembahasan

Pada baris [1], `import java.util.Scanner;`

Mengimpor class scanner yang terdapat di dalam package `java.util`. Berfungsi untuk membaca input dari pengguna.

Pada baris [3-21],

`public class PRAK103_2210817310013_RyanMuhammadIrfan` “public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“class” berfungsi untuk membuat class yang dalam baris ini diberi nama `PRAK103_2210817310013_RyanMuhammadIrfan`.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [4-20], `public static void main(String[] args)`

“public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“static” berfungsi membuat suatu method tanpa perlu melakukan instansiasi terlebih dahulu.

“void” berfungsi untuk tidak mengembalikan nilai apapun.

“main” merupakan nama fungsi yang digunakan oleh java sebagai awal masuk ke program.

“`String[] args`” berfungsi sebagai parameter yang diperlukan oleh fungsi main. Parameter ini adalah array dari argument perintah yang bisa diteruskan ke program java.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [5], `int i = 1;`

“int” merupakan tipe datanya, “i” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya 1. <tipe data> <nama variabel> = <nilai variabel>.

Pada baris [6], `Scanner userInput = new Scanner(System.in);`

Berfungsi untuk membuat sebuah objek dari class Scanner atau dalam hal ini variabel userinput akan berisi object dari Scanner class dengan cara memanggil konstruktor Scanner(System.in). System in adalah input stream yang digunakan untuk menerima input dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [7], `int number1 = userinput.nextInt();`

“int” merupakan tipe datanya, “number1” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userinput.Int()`. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [8], `int number2 = userinput.nextInt();`

“int” merupakan tipe datanya, “number2” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userinput.Int()`. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [9], `do {...}` berfungsi sebagai kombinasi dengan “while” untuk membuat loop “do-while”. do-while merupakan modifikasi dari perulangan while dengan memindahkan posisi pemeriksaan kondisi ke akhir perulangan.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [11-17], `if (number2 % 2 != 0) {...}`

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “number2 % 2 != 0” merupakan kondisi yang artinya value dari number2 modulus 2 tidak sama dengan 0 (bilangan genap).

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [12], `System.out.print(number2);`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak hasil dari nilai value number2.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [13-15], `if (i < number1) {...}`

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “i < number1” merupakan kondisi yang artinya value dari i kurang dari number 1.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [14], `System.out.print(", ");`

“`System.out.print(...)`” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak “, ”.

“`;`” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [16], `i++;`

“`i++`” berfungsi sebagai increment yang digunakan untuk menambah value `i` 1 angka. Dalam hal ini digunakan untuk menggerakkan iterasi loop menuju kondisi berhenti.

“`;`” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [18], `number2++;`

“`number2++`” berfungsi sebagai increment yang digunakan untuk menambah value `number2` 1 angka. Dalam hal ini digunakan untuk menggerakkan iterasi loop menuju kondisi berhenti.

“`;`” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [19], `while (i <= number1);`

“`while`” merupakan pemeriksaan kondisi perulangan di akhir (do-while) yang mengulang sebuah blok kode selama kondisi yang ditentukan benar. “`i<=number1`” merupakan sebuah kondisi. Dalam hal ini berarti kondisinya value `i<= value number1` apabila lewat dari kondisi tersebut maka perulangan akan berhenti.

“`;`” berfungsi untuk mengakhiri sebuah perintah.

D. Tautan GIT

https://github.com/ryanmi04/Praktikum-Pemrograman-2-Paralel-1/blob/main/PRAKTIKUM%201/src/PRAK103_2210817310013_RyanMuhammadIrfan.java

SOAL 4

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

- Batu (B) mengalahkan Gunting (G).
- Gunting (G) mengalahkan Kertas(K).
- Kertas (K) mengalahkan Batu (B).
- Jika kedua pemain memilih tangan yang sama, hasilnya adalah seri
- Terdapat tiga ronde dalam sekali permainan. Pemain melakukan suit sebanyak tiga kali.
- Setiap ronde akan ditentukan pemenangnya berdasarkan aturan yang telah ditentukan sebelumnya
- 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 rondennya. Hasilnya antara “Abu”, “Bagas” atau “Seri”

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	?

Tabel 7. Soal 4

Simpan coding anda dengan nama: **PRAK104-NIM-Nama.java**

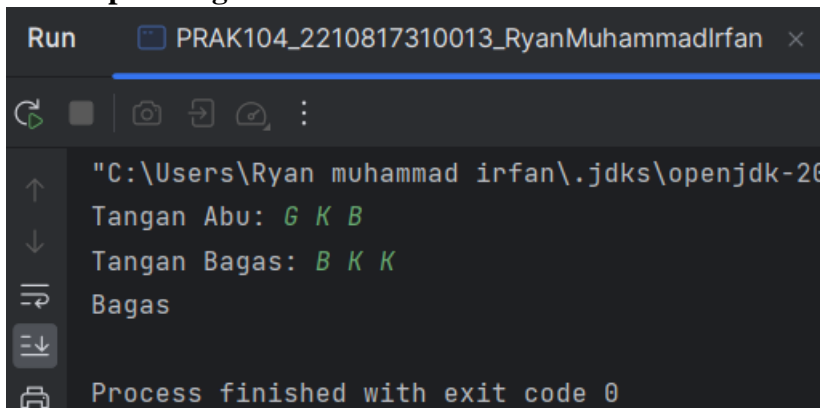
A. Source Code

1	import java.util.Scanner;
2	
3	public class PRAK104_2210817310013_RyanMuhammadIrfan {
4	public static void main(String[] args) {
5	int abu_point = 0;
6	int bagas_point = 0;
7	Scanner userinput = new Scanner(System.in);
8	System.out.print("Tangan Abu: ");
9	String abu_round1 = userinput.next().toUpperCase();
10	String abu_round2 = userinput.next().toUpperCase();
11	String abu_round3 = userinput.next().toUpperCase();
12	

13	System.out.print("Tangan Bagas: ");
14	String bagas_round1 = userInput.next().toUpperCase();
15	String bagas_round2 = userInput.next().toUpperCase();
16	String bagas_round3 = userInput.next().toUpperCase();
17	
18	
19	if (abu_round1.equals("B") && bagas_round1.equals("G")
20	abu_round1.equals("G") && bagas_round1.equals("K")
21	abu_round1.equals("K") && bagas_round1.equals("B")) {
22	abu_point++;
23	}
24	else if (abu_round1.equals("G") && bagas_round1.equals("B")
25	abu_round1.equals("K") && bagas_round1.equals("G")
26	abu_round1.equals("B") && bagas_round1.equals("K")) {
27	bagas_point++;
28	}
29	
30	if (abu_round2.equals("B") && bagas_round2.equals("G")
31	abu_round2.equals("G") && bagas_round2.equals("K")
32	abu_round2.equals("K") && bagas_round2.equals("B")) {
33	abu_point++;
34	}
35	else if (abu_round2.equals("G") && bagas_round2.equals("B")
36	abu_round2.equals("K") && bagas_round2.equals("G")
37	abu_round2.equals("B") && bagas_round2.equals("K")) {
38	bagas_point++;
39	}
40	
41	if (abu_round3.equals("B") && bagas_round3.equals("G")
42	abu_round3.equals("G") && bagas_round3.equals("K")
43	abu_round3.equals("K") && bagas_round3.equals("B")) {
44	abu_point++;
45	}
46	else if (abu_round3.equals("G") && bagas_round3.equals("B")
47	abu_round3.equals("K") && bagas_round3.equals("G")
48	abu_round3.equals("B") && bagas_round3.equals("K")) {
49	bagas_point++;
50	}
51	
52	if (abu_point>bagas_point){
53	System.out.println("Abu");
54	}
55	else if (bagas_point>abu_point) {
56	System.out.println("Bagas");
57	}
58	else{
59	System.out.println("Seri");
60	}
61	
62	}
63	}

Tabel 8. Source Code Soal 4

B. Output Program



```
Run PRAK104_2210817310013_RyanMuhammadIrfan x
"C:\Users\Ryan muhammad irfan\.jdk\openjdk-20
Tangan Abu: G K B
Tangan Bagas: B K K
Bagas
Process finished with exit code 0
```

Gambar 4. Screenshot Hasil Jawaban Soal 4

C. Pembahasan

Pada baris [1], `import java.util.Scanner;`

Mengimpor class scanner yang terdapat di dalam package `java.util`. Berfungsi untuk membaca input dari pengguna.

Pada baris [3-63],

`public class PRAK104_2210817310013_RyanMuhammadIrfan` “public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“class” berfungsi untuk membuat class yang dalam baris ini diberi nama `PRAK104_2210817310013_RyanMuhammadIrfan`.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [4-62], `public static void main(String[] args)`

“public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“static” berfungsi membuat suatu method tanpa perlu melakukan instansiasi terlebih dahulu.

“void” berfungsi untuk tidak mengembalikan nilai apapun.

“main” merupakan nama fungsi yang digunakan oleh java sebagai awal masuk ke program.

“`String[] args`” berfungsi sebagai parameter yang diperlukan oleh fungsi main. Parameter ini adalah array dari argument perintah yang bisa diteruskan ke program java.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [5], `int abu_point = 0;`

“int” merupakan tipe datanya, “`abu_point`” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya 0. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [6], `int bagas_point = 0;`

“int” merupakan tipe datanya, “bagas_point” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya 0. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [7], `Scanner userInput = new Scanner(System.in);`

Berfungsi untuk membuat sebuah objek dari class Scanner atau dalam hal ini variabel userInput akan berisi object dari Scanner class dengan cara memanggil konstruktor Scanner(System.in). System in adalah input stream yang digunakan untuk menerima input dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [8], `System.out.print "Tangan Abu: " ;`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Tangan Abu: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [9], `String abu_round1 = userInput.next().toUpperCase();`

“String” merupakan tipe datanya, “abu_round1” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.next().toUpperCase()`. <tipe data> <nama variabel> = <nilai variabel>.

“userInput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“next()” berfungsi untuk membaca membaca String kata pertama dari input.

“toUpperCase()” berfungsi untuk mengubah semua huruf dalam string menjadi huruf kapital.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [10], `String abu_round2 = userInput.next().toUpperCase();`

“String” merupakan tipe datanya, “abu_round2” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.next().toUpperCase()`. <tipe data> <nama variabel> = <nilai variabel>.

“userInput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“next()” berfungsi untuk membaca membaca String kata pertama dari input.

“toUpperCase()” berfungsi untuk mengubah semua huruf dalam string menjadi huruf kapital.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [11], `String abu_round3 = userInput.next().toUpperCase();`
“String” merupakan tipe datanya, “abu_round3” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.next().toUpperCase()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“next()” berfungsi untuk membaca membaca String kata pertama dari input.

“toUpperCase()” berfungsi untuk mengubah semua huruf dalam string menjadi huruf kapital.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [13], `System.out.print "Tangan Bagas: ";`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Tangan Bagas: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [14], `String bagas_round1 = userInput.next().toUpperCase();`

“String” merupakan tipe datanya, “bagas_round1” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.next().toUpperCase()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“next()” berfungsi untuk membaca membaca String kata pertama dari input.

“toUpperCase()” berfungsi untuk mengubah semua huruf dalam string menjadi huruf kapital.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [15], `String bagas_round2 = userInput.next().toUpperCase();`

“String” merupakan tipe datanya, “bagas_round2” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya `userInput.next().toUpperCase()`. <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“next()” berfungsi untuk membaca membaca String kata pertama dari input.

“toUpperCase()” berfungsi untuk mengubah semua huruf dalam string menjadi huruf kapital.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [16], String bagas_round3 =
userinput.next().toUpperCase();

“String” merupakan tipe datanya, “bagas_round3” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya userinput.next().toUpperCase(). <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“next()” berfungsi untuk membaca membaca String kata pertama dari input.

“toUpperCase()” berfungsi untuk mengubah semua huruf dalam string menjadi huruf kapital.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [19-23],

```
if ( abu_round1.equals("B") && bagas_round1.equals("G") ||  
    abu_round1.equals("G") && bagas_round1.equals("K") ||  
    abu_round1.equals("K") && bagas_round1.equals("B")) {  
    abu_point++;  
}
```

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “|” merupakan operator (or) dan “&&” merupakan operator (and).

“abu_point++” berfungsi sebagai increment yang digunakan untuk menambah value abu_point 1 angka jika kondisi dari if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [24-28],

```
else if ( abu_round1.equals("G") && bagas_round1.equals("B")  
||  
    abu_round1.equals("K") && bagas_round1.equals("G") ||  
    abu_round1.equals("B") && bagas_round1.equals("K")) {  
    bagas_point++;  
}
```

“else if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisi dari if tidak terpenuhi. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “|” merupakan logical operator (or) dan “&&” merupakan logical operator (and).

“bagas_point++” berfungsi sebagai increment yang digunakan untuk menambah value bagas_point 1 angka jika kondisi dari else if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [30-34],

```
if ( abu_round2.equals("B") && bagas_round2.equals("G") ||
    abu_round2.equals("G") && bagas_round2.equals("K") ||
    abu_round2.equals("K") && bagas_round2.equals("B")) {
    abu_point++;
}
```

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “|” merupakan operator (or) dan “&&” merupakan operator (and).

“abu_point++” berfungsi sebagai increment yang digunakan untuk menambah value abu_point 1 angka jika kondisi dari if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [35-39],

```
else if ( abu_round2.equals("G") && bagas_round2.equals("B")
||
    abu_round2.equals("K") && bagas_round2.equals("G") ||
    abu_round2.equals("B") && bagas_round2.equals("K")) {
    bagas_point++;
}
```

“else if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisi dari if tidak terpenuhi. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “|” merupakan logical operator (or) dan “&&” merupakan logical operator (and).

“bagas_point++” berfungsi sebagai increment yang digunakan untuk menambah value bagas_point 1 angka jika kondisi dari else if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [41-45],

```
if ( abu_round3.equals("B") && bagas_round3.equals("G") ||
    abu_round3.equals("G") && bagas_round3.equals("K") ||
    abu_round3.equals("K") && bagas_round3.equals("B")) {
    abu_point++;
}
```

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “|” merupakan operator (or) dan “&&” merupakan operator (and).

“abu_point++” berfungsi sebagai increment yang digunakan untuk menambah value abu_point 1 angka jika kondisi dari if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [46-50],

```
else if ( abu_round3.equals("G") && bagas_round3.equals("B")  
||  
    abu_round3.equals("K") && bagas_round3.equals("G") ||  
    abu_round3.equals("B") && bagas_round3.equals("K")) {  
    bagas_point++;  
}
```

“else if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisi dari if tidak terpenuhi. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “||” merupakan logical operator (or) dan “&&” merupakan logical operator (and).

“bagas_point++” berfungsi sebagai increment yang digunakan untuk menambah value bagas_point 1 angka jika kondisi dari else if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [52-54], if (abu_point>bagas_point) {...}

“if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisinya benar. Jika salah maka tidak akan dieksekusi atau akan berlanjut ke percabangan else if atau else. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “abu_point>bagas_point” merupakan kondisi yang artinya value abu_point lebih dari bagas_point.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [53], System.out.print("Abu");

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Abu jika kondisi if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [55-57], else if (bagas_point>abu_point) {...}

“else if” berfungsi sebagai percabangan yang akan dieksekusi jika kondisi dari if tidak terpenuhi. cara kerjanya dengan membandingkan dengan Relational Operator atau Logical Operator. “abu_point>bagas_point” merupakan kondisi yang artinya value bagas_point lebih dari abu_point.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [56], System.out.print("Bagas");

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Bagas jika kondisi else if terpenuhi.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [58-60], `else{...}`

“else” berfungsi sebagai percabangan yang akan dieksekusi jika kondisi dari if dan else if tidak terpenuhi.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [59], `System.out.print("Seri");`

“`System.out.print(...)`” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Seri jika kondisi if dan else if tidak terpenuhi.

“`;`” berfungsi untuk mengakhiri sebuah perintah.

D. Tautan GIT

https://github.com/ryanmi04/Praktikum-Pemrograman-2-Paralel-1/blob/main/PRAKTIKUM%201/src/PRAK104_2210817310013_RyanMuhammadIrfan.java

SOAL 5

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

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	?

Tabel 9. Soal 5

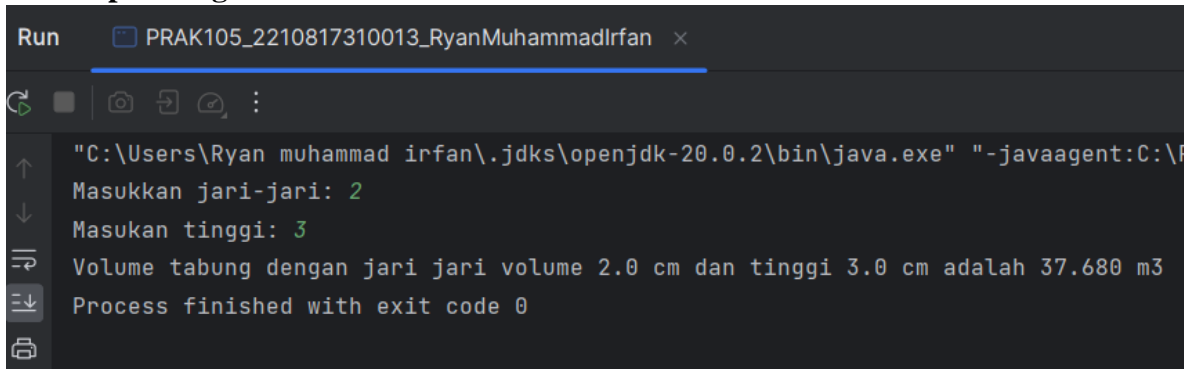
Simpan coding anda dengan nama: **PRAK105-NIM-Nama.java**

A. Source Code

```
1 import java.util.Scanner;
2 public class PRAK105_2210817310013_RyanMuhammadIrfan {
3     public static void main(String[] args) {
4         final double phi = 3.14;
5         Scanner userInput = new Scanner(System.in);
6
7         System.out.print("Masukkan jari-jari: ");
8         double radius = userInput.nextDouble();
9         System.out.print("Masukan tinggi: ");
10        double height = userInput.nextDouble();
11
12        double volume = phi * (radius*radius) * height;
13
14        System.out.printf("Volume tabung dengan jari jari
15        volume %.1f cm " +
16        "dan tinggi %.1f cm adalah %.3f
17        m3",radius, height, volume);
18    }
19 }
```

Tabel 10. Source Code Soal 5

B. Output Program



```
Run PRAK105_2210817310013_RyanMuhammadIrfan x
"C:\Users\Ryan muhammad irfan\.jdk\openjdk-20.0.2\bin\java.exe" "-javaagent:C:\f
Masukkan jari-jari: 2
Masukkan tinggi: 3
Volume tabung dengan jari jari volume 2.0 cm dan tinggi 3.0 cm adalah 37.680 m3
Process finished with exit code 0
```

Gambar 5. Screenshot Hasil Jawaban Soal 5

C. Pembahasan

Pada baris [1], `import java.util.Scanner;`

Mengimpor class scanner yang terdapat di dalam package `java.util`. Berfungsi untuk membaca input dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [2-17],

`public class PRAK105_2210817310013_RyanMuhammadIrfan` “public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“class” berfungsi untuk membuat class yang dalam baris ini diberi nama `PRAK105_2210817310013_RyanMuhammadIrfan`.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [3-16], `public static void main(String[] args)`

“public” berfungsi sebagai penanda bahwa class, objek, method, atau atribut dapat diakses dari class lain.

“static” berfungsi membuat suatu method tanpa perlu melakukan instansiasi terlebih dahulu.

“void” berfungsi untuk tidak mengembalikan nilai apapun.

“main” merupakan nama fungsi yang digunakan oleh java sebagai awal masuk ke program.

“String[] args” berfungsi sebagai parameter yang diperlukan oleh fungsi main. Parameter ini adalah array dari argument perintah yang bisa diteruskan ke program java.

“{...}” berfungsi untuk memulai dan mengakhiri blok kode.

Pada baris [4], `final double phi = 3.14;`

“final” berfungsi untuk bahwa suatu variabel tidak dapat diubah.

“double” merupakan tipe datanya, “phi” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya 3.14. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [5], `Scanner userInput = new Scanner(System.in);`

Berfungsi untuk membuat sebuah objek dari class Scanner atau dalam hal ini variabel userInput akan berisi object dari Scanner class dengan cara memanggil konstruktor Scanner(System.in). System in adalah input stream yang digunakan untuk menerima input dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [7], `System.out.print("Masukkan jari-jari: ");`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan jari-jari: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [8], `double radius = userInput.nextDouble();`

“double” merupakan tipe datanya, “radius” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya userInput.nextDouble(). <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextDouble()” berfungsi untuk membaca nilai Double dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [9], `System.out.print("Masukkan tinggi: ");`

“System.out.print(...)” berfungsi untuk mencetak angka atau karakter. Dalam baris ini mencetak Masukan tinggi: .

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [10], `double height = userInput.nextDouble();`

“double” merupakan tipe datanya, “height” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya userInput.nextDouble(). <tipe data> <nama variabel> = <nilai variabel>.

“userinput” merupakan object dari class scanner yang berfungsi mengambil input dari pengguna.

“nextDouble()” berfungsi untuk membaca nilai Double dari pengguna.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [12], `double volume = phi * (radius*radius) * height;`
“double” merupakan tipe datanya, “volume” merupakan variabel yang diberi sebuah penamaan, Lalu setelah itu adalah valuenya yaitu value dari phi dikali (value radius dikali value radius) dikali value height. <tipe data> <nama variabel> = <nilai variabel>.

“;” berfungsi untuk mengakhiri sebuah perintah.

Pada baris [14-15],

```
System.out.printf("Volume tabung dengan jari jari volume %.1f  
cm " +
```

```
"dan tinggi %.1f cm adalah %.3f m3", radius, height, volume);
```

“System.out.printf(…)” berfungsi untuk mencetak angka atau karakter dengan format spesifik. Dalam baris ini mencetak " Volume tabung dengan jari jari volume %.1f cm " + "dan tinggi %.1f cm adalah %.3f m3", value radius, value height, value volume ".

“%.1f” berfungsi untuk mengatur tampilan bilangan decimal 1 angka dibelakang koma.

“%.3f” berfungsi untuk mengatur tampilan bilangan decimal 3 angka dibelakang koma.

“;” berfungsi untuk mengakhiri sebuah perintah.

D. Tautan GIT

https://github.com/ryanmi04/Praktikum-Pemrograman-2-Paralel-1/blob/main/PRAKTIKUM%201/src/PRAK105_2210817310013_RyanMuhammadIrfan.java