

**LAPORAN AKHIR PRAKTIKUM
PEMROGRAMAN II**



Oleh:

Muhammad Ryan Rizky Rahmadi NIM. 2210817310001

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

LEMBAR PENGESAHAN

LAPORAN AKHIR PRAKTIKUM PEMROGRAMAN II

Laporan Praktikum Pemrograman II

Modul 0 : Input, Output, dan Logika Sederhana.

Modul 1 : Struktur Dasar Kode Java

Modul 2 : Class dan Objek

Modul 3 : Enkapulasi dan Collection

Modul 4 : Pewarisan/Inheritance & Diagram Class

Modul 5 : Polimorfisme

Modul 6 : Graphical User Interface

ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman II. Laporan Akhir Praktikum ini dikerjakan oleh:

Nama Praktikan : Muhammad Ryan Rizky Rahmadi

NIM : 2210817310001

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 TABEL	6
DAFTAR GAMBAR.....	8
MODUL 0: INPUT, OUTPUT, DAN LOGIKA SEDERHANA.....	9
SOAL 1	9
A. Source Code	9
B. Output Program	9
C. Pembahasan.....	9
SOAL 2.....	9
A. Source Code	10
B. Output Program	10
C. Pembahasan.....	10
SOAL 3.....	11
A. Source Code	11
B. Output Program	11
C. Pembahasan.....	12
SOAL 4.....	13
A. Source Code	13
B. Output Program	13
C. Pembahasan.....	14
SOAL 5.....	15
A. Source Code	15
B. Output Program	15
C. Pembahasan.....	15
MODUL 1: STRUKTUR DASAR KODE JAVA	17
SOAL 1	17
A. Source Code	17
B. Output Program	19
C. Pembahasan.....	19

SOAL 2.....	20
A. Source Code	20
B. Output Program	20
C. Pembahasan.....	21
SOAL 3.....	22
A. Source Code	22
B. Output Program	23
C. Pembahasan.....	23
SOAL 4.....	24
A. Source Code	24
B. Output Program	26
C. Pembahasan.....	26
SOAL 5.....	27
A. Source Code	27
B. Output Program	27
C. Pembahasan.....	27
MODUL 2: CLASS DAN OBJEK.....	29
SOAL 1	29
A. Source Code	29
B. Output Program	31
C. Pembahasan.....	31
SOAL 2.....	32
A. Source Code	32
B. Output Program	33
C. Pembahasan.....	33
SOAL 3.....	35
A. Source Code	35
B. Output Program	36
C. Pembahasan.....	36
MODUL 3: ENKAPULASI DAN COLLECTION.....	38
SOAL 1	38
A. Source Code	38
B. Output Program	40

C. Pembahasan.....	40
SOAL 2.....	41
A. Source Code	43
B. Output Program.....	45
C. Pembahasan.....	45
SOAL 3.....	47
A. Source Code	48
B. Output Program.....	52
C. Pembahasan.....	52
MODUL 4: PEWARISAN/INHERITANCE & DIAGRAM KELAS	54
SOAL 1	54
A. Source Code	54
B. Output Program.....	55
C. Pembahasan.....	55
SOAL 2.....	57
A. Source Code	59
B. Output Program.....	61
C. Pembahasan.....	61
MODUL 5: POLIMORFISME.....	63
SOAL63	
A. Source Code	67
B. Output Program.....	70
C. Pembahasan.....	70
MODUL 6: GRAPHICAL USER INTEFACE.....	73
SOAL73	
A. Source Code	74
B. Output Program.....	77
C. Pembahasan.....	77

DAFTAR TABEL

MODUL 0: HOW TO PROGRAM

Tabel 1.1 Soal 1	9
Tabel 2.1 Source Code Soal 1	9
Tabel 3.2 Soal 2	10
Tabel 4.2 Source Code Soal 2	10
Tabel 5.3 Soal 3	11
Tabel 6.3 Source Code Soal 3	11
Tabel 7.4 Soal 3	13
Tabel 8.4 Source Code Soal 4	13
Tabel 9.5 Soal 5	15
Tabel 10.5 Source Code Soal 5	15

Modul 1 : Struktur Dasar Kode Java

Tabel 11.1 Soal 1	17
Tabel 12.1 Source Code Soal 1	19
Tabel 13.2 Soal 2	20
Tabel 14.2 Source Code Soal 2	20
Tabel 15.3 Soal 3	22
Tabel 16.3 Source Code Soal 3	22
Tabel 17.4 Soal 3	24
Tabel 18.4 Source Code Soal 4	26
Tabel 19.5 Soal 5	27
Tabel 20.5 Source Code Soal 5	27

Modul 2 : Class dan Objek

Tabel 21.1 Soal 1	29
Tabel 22.1 Source Code Soal 1 Buah	30
Tabel 23.1 Source Code Soal 1 Main	30
Tabel 24.2 Soal 2	32
Tabel 25.2 Source Code Soal 2 Kopi	33
Tabel 26.2 Source Code Soal 2 Main	33
Tabel 27.3 Soal 3	35
Tabel 28.3 Source Code Soal 3 Pegawai	35
Tabel 29.3 Source Code Soal 3 Main	36

Modul 3 : Enkapulasi dan Collection

Tabel 30.1 Soal 1	38
Tabel 31.1 Source Code Soal 1 Dadu	39
Tabel 32.1 Source Code Soal 1 Main	39

Tabel 33.2 Soal 2	42
Tabel 34.2 Source Code Soal 2 Negara.....	44
Tabel 35.2 Source Code Soal 2 Main	45
Tabel 36.3 Soal 3	48
Tabel 37.3 Source Code Soal 3 Kopi	49
Tabel 38.3 Source Code Soal 3 Main	51

Modul 4 : Pewarisan/Inheritance & Diagram Class

Tabel 39.1 Soal 1	54
Tabel 40.1 Source Code Soal 1 Peliharaan.....	55
Tabel 41.1 Source Code Soal 1 Man	55
Tabel 42.2 Soal 2	58
Tabel 43.2 Source Code Soal 2 HewanPeliharaan	59
Tabel 44.2 Source Code Soal 2 Anjing	60
Tabel 45.2 Source Code Soal 2 Kucing.....	60
Tabel 46.2 Source Code Soal 2 Main	61

Modul 5 : Polimorfisme

Tabel 47.1 Soal 1	67
Tabel 48.1 Source Code Soal 1 Spehere.....	67
Tabel 49.1 Source Code Soal 1 Shape.....	68
Tabel 50.1 Source Code Soal 1 Rectangle	68
Tabel 51.1 Source Code Soal 1 Cylinder	69
Tabel 52.1 Source Code Soal 1 Paint	69
Tabel 53.1 Source Code Soal 1 PaintThings	70

Modul 6 : Graphical User Interface

Tabel 54.1 Source Code Soal 1 Mahasiswa	75
Tabel 55.1 Source Code Soal 1 Main	77

DAFTAR GAMBAR

MODUL 0: HOW TO PROGRAM

Gambar 1.1 Screenshoot Output Soal 1.....	9
Gambar 2.2 Output Screenshoot Soal 2.....	10
Gambar 3.3 Output Screenshoot Soal 3.....	11
Gambar 4.4 Output Screenshoot Soal 4.....	13
Gambar 5.5 Output Screenshoot Soal 5.....	15

Modul 1 : Struktur Dasar Kode Java

Gambar 6.1 Screenshoot Output Soal 1.....	19
Gambar 7.2 Output Screenshoot Soal 2.....	20
Gambar 8.3 Output Screenshoot Soal 3.....	23
Gambar 9.4 Output Screenshoot Soal 4.....	26
Gambar 10.5 Output Screenshoot Soal 5.....	27

Modul 2 : Class dan Objek

Gambar 11.1 Screenshoot Output Soal 1.....	31
Gambar 12.2 Output Screenshoot Soal 2.....	33
Gambar 13.3 Output Screenshoot Soal 3.....	36

Modul 3 : Enkapulasi dan Collection

Gambar 14.1 Screenshoot Output Soal 1.....	40
Gambar 15.2 Output Screenshoot Soal 2.....	45
Gambar 16.3 Output Screenshoot Soal 3.....	52

Modul 4 : Pewarisan/Inheritance & Diagram Class

Gambar 17.1 Screenshoot Output Soal 1.....	55
Gambar 18.2 Output Screenshoot Soal 2.....	61

Modul 5 : Polimorfisme

Gambar 1.1 Screenshoot Output Soal 1.....	70
---	----

Modul 6 : Graphical User Interface

Gambar 1.1 Screenshoot Output Soal 1.....	77
---	----

MODUL 0: INPUT, OUTPUT, DAN LOGIKA SEDERHANA

SOAL 1

Buatlah program yang dapat menghasilkan output sebagaimana berikut:

Output
Selamat Pagi, Nama Anda
Selamat Siang, Nama Anda
Selamat Malam, Nama Anda

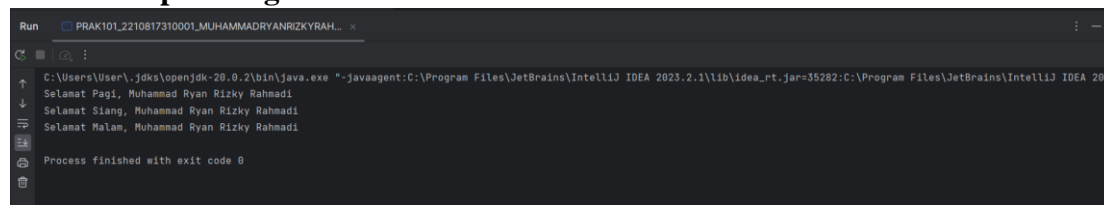
Tabel 1.1 Soal 1

A. Source Code

1	Public class PRAK101_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
2	public static void main(String[] args){
3	System.out.println("Selamat Pagi, Muhammad Ryan Rizky
4	Rahmadi");
5	System.out.println("Selamat Siang, Muhammad Ryan
6	Rizky Rahmadi");
7	System.out.println("Selamat Malam, Muhammad Ryan
8	Rizky Rahmadi");
9	}
10	}
11	

Tabel 2.1 Source Code Soal 1

B. Output Program



Gambar 1.1 Screenshoot Output Soal 1

C. Pembahasan

Line 1 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 2 dan 11 : { }, penanda awal dan akhir *block code*.

Line 3 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 4 – 9 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

SOAL 2

Buatlah program yang dapat menghasilkan output sebagaimana berikut:

Output	
	#####
#	#
#	#
	#####

Tabel 3.2 Soal 2

```

1 public class PRAK102_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
2     public static void main(String[] args) {
3         System.out.println
4         ("#####");
5         System.out.println ("#");
6         System.out.println ("# MUHAMAMAD RYAN RIZKY RAHMADI
7         #");
8         System.out.println ("# 2210817310001
9         #");
10        System.out.println ("#");
11        System.out.println
12        ("#####");
13    }
14 }
15
16
17

```

B. Output Program

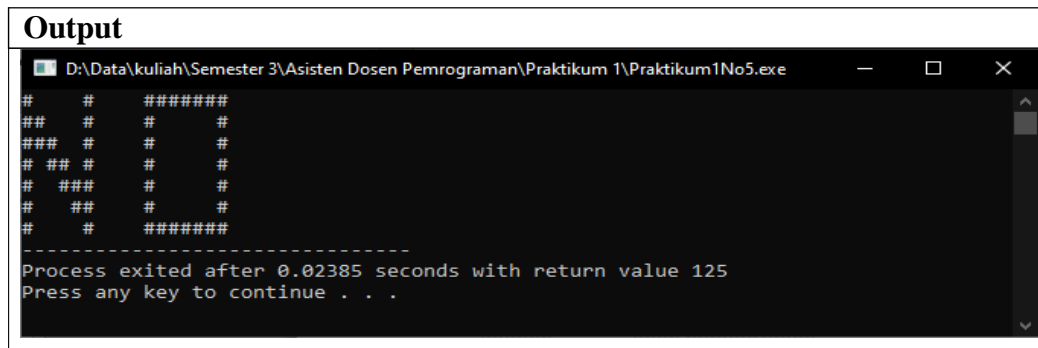
Gambar 2.2 Output Screenshoot Soal 2

Line 1 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

SOAL 3

Buatlah program yang dapat menghasilkan output sebagaimana berikut:



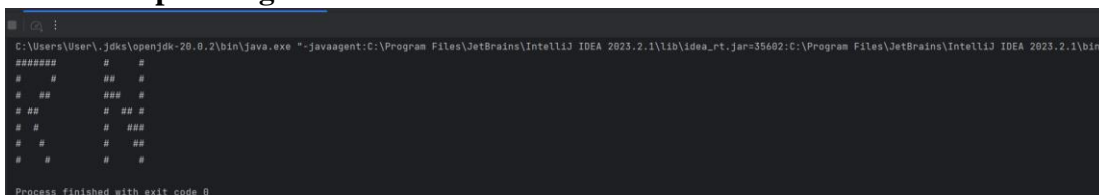
Tabel 5.3 Soal 3

A. Source Code

```
1 public class PRAK103_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
2     public static void main(String[] args) {
3         System.out.println ("##### # #");
4         System.out.println ("# # ## #");
5         System.out.println ("# ## ### #");
6         System.out.println ("# ## # ## #");
7         System.out.println ("# # # ###");
8         System.out.println ("# # # ##");
9         System.out.println ("# # # #");
10    }
11
12 }
13 }
```

Tabel 6.3 Source Code Soal 3

B. Output Program



Gambar 3.3 Output Screenshoot Soal 3

C. Pembahasan

Line 1 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 2 dan 13 : { }, penanda awal dan akhir *block code*.

Line 3 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 4 – 10 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

SOAL 4

Buatlah program yang dapat menghasilkan output sebagaimana berikut:

Output
Perkenalkan, Nama Saya <Nama>, Umur Saya <Umur>

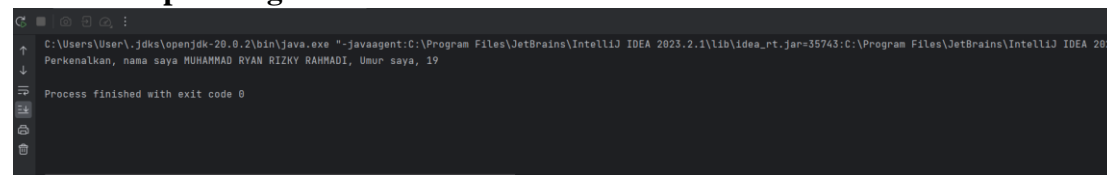
Tabel 7.4 Soal 3

A. Source Code

1	public class PRAK104_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
2	public static void main(String[] args) {
3	String namy = ("MUHAMMAD RYAN RIZKY RAHMADI");
4	int umur = (19);
5	System.out.println ("Perkenalkan, nama saya " +
6	(namy) + ", Umur saya, " + (umur));
7	}
8	}
9	
10	

Tabel 8.4 Source Code Soal 4

B. Output Program



Gambar 4.4 Output Screenshoot Soal 4

C. Pembahasan

Line 1 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 2 dan 10 : { }, penanda awal dan akhir *block code*.

Line 3 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 4 : *String*, tipe data teks yang berisikan karakter lebih.

Line 5 : *int*, untuk penulisan angka bulat.

Line 7: *System.out.println* mencetak sesuatu dari program sebagai *output*.

SOAL 5

Buatlah program yang dapat menghasilkan output sebagaimana berikut:

Input
Masukan Makanan Favorit: Bakso Masukan Hobi: Renang
Output
Aku Suka Makan Bakso, dan Hobiku Renang

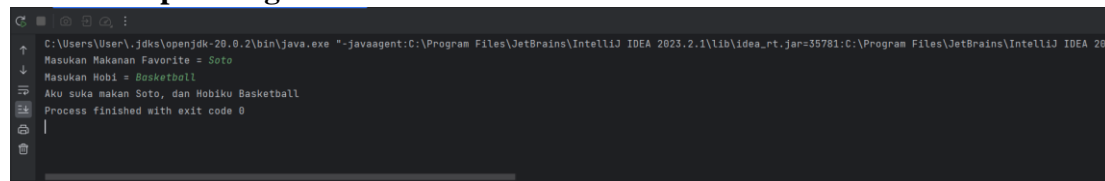
Tabel 9.5 Soal 5

A. Source Code

1	import	java.util.Scanner;
2		
3	public class	PRAK105_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
4	public static void	main(String[] args) {
5	String	makan;
6	String	hoby;
7	Scanner namy	= new Scanner(System.in);
8		
9	System.out.print("Masukan Makanan Favorite = ");	
10	makan	= namy.next();
11		
12	System.out.print("Masukan Hobi = ");	
13	hoby	= namy.next();
14		
15	System.out.print("Aku suka makan "+ (makan)+" , dan	
16	Hobiku	" + (hoby));
17	}	
18	}	
19		
20		

Tabel 10.5 Source Code Soal 5

B. Output Program



Gambar 5.5 Output Screenshoot Soal 5

C. Pembahasan

Line 1 : *Import*, perintah untuk program memasukkan bahasa java sehingga aktif.

Line 3 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 20 : { }, penanda awal dan akhir *block code*.

Line 5 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 6-7 : *String*, tipe data teks yang berisikan karakter lebih.

Line 10 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

MODUL 1:

STRUKTUR DASAR KODE JAVA

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 11.1 Soal 1

A. Source Code

1	import java.util.Scanner;
2	
3	public class
4	PRAK101_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
5	public static void main(String[] args){
6	String namapanjang;
7	String tempatlahir, namabulan = "";
8	int tanggallahir, bulanlahir, tahunlahir,
9	tinggibadan;
10	float beratbadan;
11	Scanner namy = new Scanner (System.in);
12	System.out.print("Masukan Nama Lengkap :
13	");
14	namapanjang = namy.nextLine();
15	System.out.print("Masukan Tempat Lahir :
16	");
17	tempatlahir = namy.next();
18	System.out.print("Masukan Tanggal Lahir :
20	");
21	tanggallahir = namy.nextInt();

```

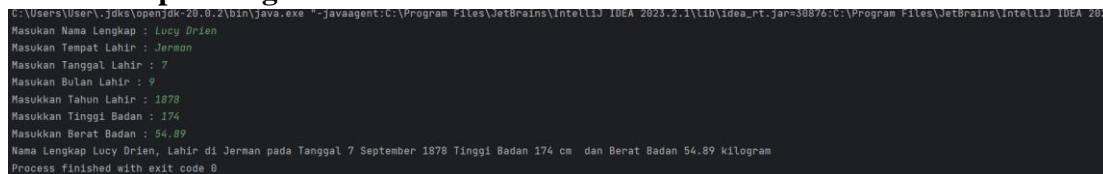
22         System.out.print("Masukan Bulan Lahir :
23     ");
24     bulanlahir = namy.nextInt();
25     switch (bulanlahir){
26         case 1:
27             namabulan = "Januari";
28             break;
29         case 2:
30             namabulan = "Febuari";
31             break;
32         case 3:
33             namabulan = "Maret" ;
34             break;
35         case 4:
36             namabulan = "April";
37             break;
38         case 5:
39             namabulan = "Mei";
40             break;
41         case 6:
42             namabulan = "Juni";
43             break;
44         case 7:
45             namabulan = "Juli";
46             break;
47         case 8:
48             namabulan = "Agustus";
49             break;
50         case 9:
51             namabulan = "September";
52             break;
53         case 10:
54             namabulan = "Oktober";
55             break;
56         case 11:
57             namabulan = "November";
58             break;
59         case 12:
60             namabulan = "Desember";
61             break;
62     }
63     System.out.print("Masukkan Tahun Lahir : ");
64     tahunlahir = namy.nextInt();
65     System.out.print("Masukkan Tinggi Badan : ");
66     tinggibadan = namy.nextInt();
67     System.out.print("Masukkan Berat Badan : ");
68     beratbadan = namy.nextFloat();
69
70

```

71	System.out.print("Nama Lengkap "+ (namapanjang)+",
72	Lahir di "+ (tempatlahir)
73	+ " pada Tanggal "+ (tanggallahir)+" "+
74	(namabulan)+" "+(tahunlahir)+ " Tinggi Badan
75	" +(tinggibadan)+" cm "+ " dan Berat Badan "+(beratbadan)+"
76	kilogram");
77	
78	}
79	}
80	
81	
82	
83	

Tabel 12.1 Source Code Soal 1

B. Output Program



```

C:\Users\User\jds\openjdk-20.0.2\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=36876:C:\Program Files\JetBrains\IntelliJ IDEA 202
Masukan Nama Lengkap : Lucy Orien
Masukan Tempat Lahir : Jerman
Masukan Tanggal Lahir : 7
Masukan Bulan Lahir : 9
Masukkan Tahun Lahir : 1878
Masukkan Tinggi Badan : 174
Masukkan Berat Badan : 54.89
Nama Lengkap Lucy Orien, 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 6.1 Screenshoot Output Soal 1

C. Pembahasan

Line 1 : *Import*, perintah untuk program memasukkan bahasa java sehingga aktif.

Line 2 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 83 : {}, penanda awal dan akhir *block code*.

Line 5 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 6-7 : *String*, tipe data teks yang berisikan karakter lebih.

Line 8-9: *int*, tipe data *integer* atau menerima bilangan bulat.

Line 10 : *float* tipe data *float* atau menerima bilangan desimal.

Line 11, : *Scanner* untuk menginput data.

Line 13,16,20,23,64,67,71,dan 75 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

Line 26-62 : *Switch*, membuat suatu menu yang di sebut *case* untuk membadandingkan sebuah nilai *variable*.

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	?

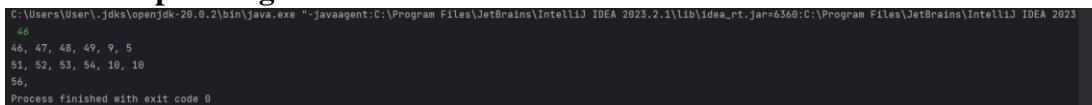
Tabel 13.2 Soal 2

A. Source Code

```
1  import java.util.Scanner;
2
3
4  public class PRAK102_2210817310001_MUHAMMADRYANRIZKYRAHMADI {
5      public static void main(String[] args) {
6          Scanner namy = new Scanner(System.in);
7          System.out.print(" ");
8          int angka = namy.nextInt();
9          int derets = 1;
10         while (derets <= 11) {
11             int bilangan;
12             if (angka % 5 == 0) {
13                 bilangan = (angka / 5) - 1;
14             } else {
15                 bilangan = angka;
16             }
17             System.out.print(bilangan + ", ");
18             if (derets % 5 == 0) {
19                 System.out.println(derets);
20             }
21             angka++;
22             derets++;
23         }
24     }
25 }
```

Tabel 14.2 Source Code Soal 2

B. Output Program



```
C:\Users\User\joks\openjdk-20.0.2\bin\java.exe --javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=6360:C:\Program Files\JetBrains\IntelliJ IDEA 2023
46
46, 47, 48, 49, 9, 5
51, 52, 53, 54, 10, 10
56,
Process finished with exit code 0
```

Gambar 7.2 Output Screenshoot Soal 2

C. Pembahasan

Line 1 : *Import*, perintah untuk program memasukkan bahasa java sehingga aktif.

Line 3 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 25 : {}, penanda awal dan akhir *block code*.

Line 5 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 6 : *Scanner* untuk menginput data.

Line 7 dan 19 : *System.out.print* mencetak sesuatu dari program sebagai *output*.

Line 8-9, dan 11: *int*, tipe data *integer* atau menerima bilangan bulat.

Line 10 : *While* untuk perulangan pada kondisi hingga terpenuhi.

Line 12,14 dan 18 : *If Else* untuk memberikan kondisi *true* atau *false*.

Line 21-22 : ++ atau *increment* untuk menambah variable satu angka.

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	?

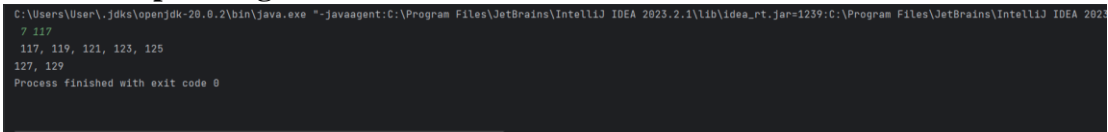
Tabel 15.3 Soal 3

A. Source Code

```
1 import java.util.Scanner;
2
3 public class PRAK103_2210817310001_MUHAMMADRYANRIZKYRAHMADI
4 {
5     public static void main(String[] args) {
6         Scanner namy = new Scanner(System.in);
7         System.out.print(" ");
8         int N = namy.nextInt();
9         System.out.print(" ");
10        int bilanganpertama = namy.nextInt();
11        int baris = 1;
12        int bilangan = bilanganpertama;
13        do {
14            if (bilangan % 2 != 0) {
15                System.out.print(bilangan);
16                if (baris % 5 != 0 && baris != N) {
17                    System.out.print(", ");
18                }
19                bilangan += 2;
20            }
21            if (baris % 5 == 0) {
22                System.out.println();
23            }
24            baris++;
25        } while (baris <= N);
26    }
27 }
```

Tabel 16.3 Source Code Soal 3

B. Output Program



```
C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=1239:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin" -Dfile.encoding=UTF-8
7 117
117, 119, 121, 123, 125
127, 129
Process finished with exit code 0
```

Gambar 8.3 Output Screenshoot Soal 3

C. Pembahasan

Line 1 : *Import*, perintah untuk program memasukkan bahasa java sehingga aktif.

Line 3 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 27 : *{ }*, penanda awal dan akhir *block code*.

Line 5 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 6 : *Scanner* untuk menginput data.

Line 7, 9, 15, 17, dan 22 : *System.out.print* mencetak sesuatu dari program sebagai *output*.

Line 8, 10 - 12: *int*, tipe data *integer* atau menerima bilangan bulat.

Line 13 dan 25 : *Do While* untuk perulangan terlebih dahulu, lalu memeriksa kondisi atau syaratnya

Line 14 dan 21 : *If Else* untuk memberikan kondisi *true* atau *false*.

Line 24 : *++* atau *increment* untuk menambah variable satu angka.

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 rondonya. 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 17.4 Soal 3

A. Source Code

```
1 import java.util.Scanner;
2
3 public class PRAK104_2210817310001_MUHAMMADRYANRIZKYRAHMADI
4 {
5     public static void main(String[] args) {
6         Scanner namy = new Scanner(System.in);
7
8         System.out.print(" Tangan Abu : ");
9         char abuRonde1 = namy.next().charAt(0);
10        System.out.print(" ");
11        char abuRonde2 = namy.next().charAt(0);
12        System.out.print(" ");
13        char abuRonde3 = namy.next().charAt(0);
14        System.out.print(" Tangan Bagas : ");
15        char bagasRonde1 = namy.next().charAt(0);
16        System.out.print(" ");
17        char bagasRonde2 = namy.next().charAt(0);
18        System.out.print(" ");
19        char bagasRonde3 = namy.next().charAt(0);
```



```

20
21     int abuScore = 0;
22     int bagasScore = 0;
23
24     if ((abuRonde1 == 'B' && bagasRonde1 == 'G') ||
25         (abuRonde1 == 'G' && bagasRonde1 == 'K') ||
26         (abuRonde1 == 'K' && bagasRonde1 == 'B')) {
27         abuScore++;
28     } else if ((abuRonde1 == 'G' && bagasRonde1 == 'B')
29 ||
30         (abuRonde1 == 'K' && bagasRonde1 == 'G') ||
31         (abuRonde1 == 'B' && bagasRonde1 == 'K')) {
32         bagasScore++;
33     }
34     if ((abuRonde2 == 'B' && bagasRonde2 == 'G') ||
35         (abuRonde2 == 'G' && bagasRonde2 == 'K') ||
36         (abuRonde2 == 'K' && bagasRonde2 == 'B')) {
37         abuScore++;
38     } else if ((abuRonde2 == 'G' && bagasRonde2 == 'B')
39 ||
40         (abuRonde2 == 'K' && bagasRonde2 == 'G') ||
41         (abuRonde2 == 'B' && bagasRonde2 == 'K')) {
42         bagasScore++;
43     }
44     if ((abuRonde3 == 'B' && bagasRonde3 == 'G') ||
45         (abuRonde3 == 'G' && bagasRonde3 == 'K') ||
46         (abuRonde3 == 'K' && bagasRonde3 == 'B')) {
47         abuScore++;
48     } else if ((abuRonde3 == 'G' && bagasRonde3 == 'B')
49 ||
50         (abuRonde3 == 'K' && bagasRonde3 == 'G') ||
51         (abuRonde3 == 'B' && bagasRonde3 == 'K')) {
52         bagasScore++;
53     }
54     if (abuScore > bagasScore) {
55         System.out.println("Abu");
56     } else if (abuScore < bagasScore) {
57         System.out.println("Bagas");
58     } else {
59         System.out.println("SERI");
60     }
61 }
62 }
63
64
65
66
67

```

68	
69	
70	
71	
72	
73	
74	
75	
76	
77	

Tabel 18.4 Source Code Soal 4

B. Output Program

```

C:\Users\User1\jdk8\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=32123:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin" -Dfile.encoding=UTF-8
Tangan Abu : G K B
Tangan Bagas : B K K
Bagas
Process finished with exit code 0

```

Gambar 9.4 Output Screenshoot Soal 4

C. Pembahasan

Line 1 : *Import*, perintah untuk program memasukkan bahasa java sehingga aktif.

Line 3 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 77 : {}, penanda awal dan akhir *block code*.

Line 5 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 6 : *Scanner* untuk menginput data.

Line 8, 10, 12, 14, 16, 18, 70, 72, dan 74: *System.out.print* mencetak sesuatu dari program sebagai *output*.

Line 10 : *char* tipe data *char* atau menerima bilangan dan symbol.

Line 24-69 : *If Else* untuk memberikan kondisi *true* atau *false*.

Line 30, 37, 45, 52, 60, dan 67 : ++ atau *increment* untuk menambah variable satu angka.

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	?

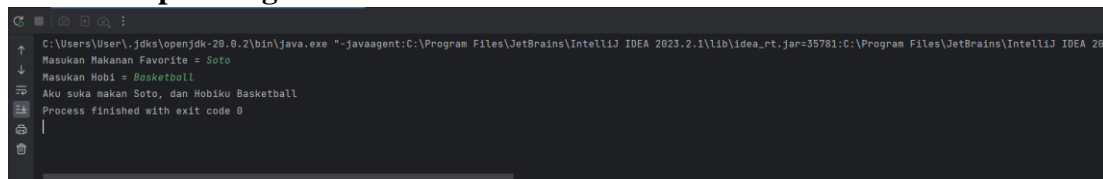
Tabel 19.5 Soal 5

A. Source Code

```
1 import java.util.Scanner;
2 public class PRAK105_2210817310001_MUHAMMADRYANRIZKYRAHMADI
3 {
4     public static void main(String [] args){
5         float jarijari, tinggi, rumus;
6         double pi = 3.14;
7         Scanner namy =new Scanner(System.in);
8         System.out.print("Masukkan Jari-Jari : ");
9         jarijari = namy.nextFloat();
10        System.out.print("Masukkan Tinggi : ");
11        tinggi = namy.nextFloat();
12        rumus = (float) (pi * (jarijari *jarijari*tinggi));
13
14        System.out.print("Volume tabung dengan jari-jari
15 "+jarijari+" dan tinggi "+tinggi+" cm adalah "+rumus+" m3");
16
17    }
18 }
19
20
```

Tabel 20.5 Source Code Soal 5

B. Output Program



Gambar 10.5 Output Screenshoot Soal 5

C. Pembahasan

Line 1 : *Import*, perintah untuk program memasukkan bahasa java sehingga aktif.

Line 2 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 22 : { }, penanda awal dan akhir *block code*.

Line 4 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 5 : *float* tipe data *float* atau menerima bilangan desimal.

Line 6 : *double* tipe data *double* atau menerima bilangan desimal.

Line 7, : *Scanner* untuk menginput data.

Line 8, 10, dan 15 : *System.out.print* mencetak sesuatu dari program sebagai *output*.

MODUL 2: CLASS DAN OBJEK

SOAL 1

Reno membeli buah-buahan di toko buah. Setiap buah memiliki diskon atau potongan harga 2% per 4kg. Apel memiliki harga Rp7.000 per 0.4kg. Mangga memiliki harga Rp3500 per 0.2kg. Alpukat memiliki harga Rp10.000 per 0.25kg. Dari informasi tersebut, buatlah sebuah program dalam bahasa java yang mengimplementasikan konsep **class** dan **object**. Ketentuan : Di **class main** terdapat 3 buah instansiasi objek dan **tidak boleh ada operasi apapun selain** instansiasi objek dan memanggil method

Output
Nama Buah: Apel Berat: 0.4 Harga: 7000.0 Jumlah Beli: 40.0kg Harga Sebelum Diskon: Rp700000.00 Total Diskon: Rp5600.00 Harga Setelah Diskon: Rp694400.00
Nama Buah: mangga Berat: 0.2 Harga: 3500.0 Jumlah Beli: 15.0kg Harga Sebelum Diskon: Rp262500.00 Total Diskon: Rp840.00 Harga Setelah Diskon: Rp261660.00
Nama Buah: alpukat Berat: 0.25 Harga: 10000.0 Jumlah Beli: 12.0kg Harga Sebelum Diskon: Rp480000.00 Total Diskon: Rp2400.00 Harga Setelah Diskon: Rp477600.00

Tabel 21.1 Soal 1

A. Source Code

1	package PRAK201_2210817310001_MUHAMMADRYANRIZKYRAHMADI;
2	
3	public class Buah {
4	String nama;
5	double berat,harga,
6	jumlahBeli,hargaSebelumDiskon,hargaSesudahDiskon,totalDisko
7	n ;
8	
9	public Buah(String namaBuah, double beratBuah, int
10	hargaBuah, double jumlahBeli){
11	this.nama = namaBuah;
12	this.berat = beratBuah;
13	this.harga = hargaBuah;

14	this.jumlahBeli = jumlahBeli;
15	this.hargaSebelumDiskon =
16	harga*(jumlahBeli/beratBuah);
17	this.totalDiskon =
18	(int) (jumlahBeli/4)*(4*harga)*0.02;
20	this.hargaSesudahDiskon = hargaSebelumDiskon-
21	totalDiskon;
22	}
23	
24	public void display(){
25	System.out.println("Nama Buah: "+nama);
26	System.out.println("Berat: "+berat+" kg");
27	System.out.println("Harga: Rp"+harga);
28	System.out.println("Harga Sebelum Diskon: Rp"+
29	String.format("%.2f",hargaSebelumDiskon));
30	System.out.println("Total Diskon: Rp"+totalDiskon);
31	System.out.println("Harga Setelah Diskon:
32	Rp"+String.format("%.2f",hargaSesudahDiskon)+"\n");
33	}
34	}

Tabel 22.1 Source Code Soal 1 Buah

1	package PRAK201_2210817310001_MUHAMMADRYANRIZKYRAHMADI;
2	public class Main {
3	public static void main(String[] args) {
4	Buah Apel1= new Buah("Apel", 0.4, 7000, 40);
5	Buah Mangga1 = new Buah("Mangga", 0.2, 3500, 15);
6	Buah Alpukat1 = new Buah("Alpukat", 0.25, 10000,
7	12);
8	
9	Apel1.display();
10	Mangga1.display();
11	Alpukat1.display();
12	}
13	}

Tabel 23.1 Source Code Soal 1 Main

B. Output Program

```
C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=13338:C:\Program Files\JetBrains\IntelliJ IDEA 20
Nama Buah: Apel
Berat: 0.4 kg
Harga: Rp7000.0
Harga Sebelum Diskon: Rp700000.00
Total Diskon: Rp5600.0
Harga Setelah Diskon: Rp694400.00

Nama Buah: Mangga
Berat: 0.2 kg
Harga: Rp3500.0
Harga Sebelum Diskon: Rp262500.00
Total Diskon: Rp840.0
Harga Setelah Diskon: Rp261660.00

Nama Buah: Alpukat
Berat: 0.25 kg
Harga: Rp10000.0
Harga Sebelum Diskon: Rp480000.00
Total Diskon: Rp2400.0
Harga Setelah Diskon: Rp477600.00

Process finished with exit code 0
```

Gambar 11.1 Screenshot Output Soal 1

C. Pembahasan

a. Buah

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2, 9, dan 24 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 2 dan 13 : {}, penanda awal dan akhir *block code*.

Line 4 : *String*, tipe data teks yang berisikan karakter lebih.

Line 5 : *double* tipe data *double* atau menerima bilangan desimal.

Line 25-31 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

Line 28 dan 31 : *String.format()*, sebuah format untuk meng-edit sebuah *string*.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 13 : {}, penanda awal dan akhir *block code*.

Line 3 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 9-11 : *Display*, merupakan fungsi menampilkan data yang diprogram.

SOAL 2

Diberikan sebuah class Soal2Main. Soal ini berisi sebuah kode program dalam bahasa java yang digunakan untuk menentukan jumlah pajak yang harus dibayar oleh pelanggan ketika berbelanja di coffe shop. Lengkapilah kode pada program berikut sehingga dapat menghasilkan keluaran seperti yang ditunjukkan pada tabel di bawah.

Ketentuan : Class Soal2Main **tidak boleh** diubah isinya. Pajak dari tiap pembelian adalah 11%.

File: Soal2Main.java

```
package praktikum2.soal2;

public class Soal2Main {
    public static void main(String[] args) {
        Kopi kopi1 = new Kopi();
        kopi1.namaKopi = "Espresso";
        kopi1.ukuran = "Medium";
        kopi1.harga = 25000;

        kopi1.info();
        kopi1.setPembeli("Alice");
        System.out.println("Pembeli Kopi: " + kopi1.getPembeli());
        System.out.println("Pajak Kopi: Rp. " + kopi1.getPajak());
    }
}
```

Output

```
Nama Kopi: Espresso
Ukuran: Medium
Harga: Rp. 25000.0
Pembeli Kopi: Alice
Pajak Kopi: Rp. 2750.0
```

Tabel 24.2 Soal 2

A. Source Code

```
1 package PRAK202_2210817310001_MUHAMMADRYANRIZKYRAHMADI;
2 public class Kopi {
3     String namaKopi, ukuran, namaPembeli;
4     int harga;
5     double pajak;
6     public void info(){
7         System.out.println("Nama Kopi: "+namaKopi);
8         System.out.println("Ukuran: "+ukuran);
9         System.out.println("Harga: Rp"+harga);
10    }
11    public void setPembeli(String namaPembeli){
12        this.namaPembeli=namaPembeli;
13    }
14    public String getPembeli(){
15        return namaPembeli;
16    }
17    public double pajak(){
18        pajak=harga*0.11;
19        return pajak;
20    }
```


21	}
22	
23	}

Tabel 25.2 Source Code Soal 2 Kopi

1	package PRAK202_2210817310001_MUHAMMADRYANRIZKYRAHMADI;
2	public class Soal2Main {
3	public static void main(String[] args) {
4	Kopi kopi1 = new Kopi();
5	kopi1.namaKopi = "Espresso";
6	kopi1.ukuran = "Medium";
7	kopi1.harga= 25000;
8	
9	kopi1.info();
10	kopi1.setPembeli("Alice");
11	System.out.println("Pembeli Kopi:
12	"+kopi1.getPembeli());
13	System.out.println("Pajak Kopi: "+ kopi1.pajak());
14	}
15	}

Tabel 26.2 Source Code Soal 2 Main

B. Output Program

```
C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar-13350:C:\Program Files\JetBrains\IntelliJ IDEA 202
Name Kopi: Espresso
Ukuran: Medium
Harga: Rp25000
Pembeli Kopi: Alice
Pajak Kopi: 2750.0
Process finished with exit code 0
```

Gambar 12.2 Output Screenshoot Soal 2

C. Pembahasan

a. Kopi

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2, 11, 14, dan 17 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 2 dan 23 : {}, penanda awal dan akhir *block code*.

Line 3 : *String*, tipe data teks yang berisikan karakter lebih.

Line 4 : *int*, tipe data *integer* atau menerima bilangan bulat.

Line 5 : *double* tipe data *double* atau menerima bilangan desimal.

Line 7-9 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

Line 15 dan 20 : *return* untuk mengembalikan nilai.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 15 : {}, penanda awal dan akhir *block code*.

Line 3 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 12 dan 13 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

SOAL 3

Diberikan 2 buah file yaitu Soal3Main.java dan Pegawai.java. Temukan dan Perbaiki kesalahan yang terdapat pada kedua file tersebut sehingga program dapat dijalankan dan menghasilkan keluaran seperti yang diminta. Baris kode yang terindikasi memiliki **error** dijadikan komentar dan berikan penjelasan **di atas** baris kode yang terdapat kesalahan. Isi penjelasan berupa penyebab/ alasan kenapa terjadi **error**. Baris berikutnya merupakan hasil **perbaikan** untuk mengatasi **error** yang terjadi. Misal pada salah satu baris kode,

//Pada baris ini terjadi **error** karena kurangnya titik koma (;)
//public String nama; public String nama;

Catatan: Jika tidak ada komentar/penjelasan tentang **error** yang terjadi, meskipun program akan lancar dan menghasilkan keluaran sesuai yang diminta, maka tidak akan dinilai.

+	<p>File: Pegawai.java</p> <pre> package praktikum2.soa3; public class Employee { public String nama; public char asal; public String jabatan; public int umur; public String getNama() { return nama; } public String getAsal() { return asal; } public void setJabatan() { this.jabatan = j; } } </pre>
	<p>File: Soal3Main.java</p> <pre> package praktikum2.soa3; public class Soal3Main { public static void main(String[] args) { Pegawai p1 = new Pegawai(); p1.nama = "Roi"; p1.asal = "Kingdom of Orvel"; p1.setJabatan("Assasin"); System.out.println("Nama Pegawai: " + p1.getNama()); } } </pre>

Tabel 27.3 Soal 3

A. Source Code

1	package PRAK203_2210817310001_MUHAMMADRYANRIZKYRAHMADI;
2	public class Pegawai {
3	public String nama;
4	public String asal; // tipe data berbeda yang harusnya
5	string tetapi char.
6	public String jabatan;
7	public int umur;
8	public String getNama() {
9	return nama;
10	}
11	// tipe data yang harusnya string tetapi char.
12	public String getAsal() {
13	return asal;
14	}
15	public void setJabatan(String j) {
16	this.jabatan = j;
17	}
18	}

Tabel 28.3 Source Code Soal 3 Pegawai

1	package PRAK203_2210817310001_MUHAMMADRYANRIZKYRAHMADI;
2	public class Soal3Main {
3	public static void main(String[] args){
4	Pegawai p1 = new Pegawai();
5	//p1.nama = "Roi" erornya karena tidak ada ;
6	p1.nama = "Roi";
7	p1.asal = "Kingdom of Orvel";
8	p1.setJabatan ("Assasin");
9	//p1.jabatan = "Assasin"; terjadi error karena
10	deklarasi tidak sesuai dengan yang ada pada class Pegawai,
11	jadi ditambahkan method setJabatan
12	p1.umur = 17; //Belum terdapat pendeklarasian
13	pada property umur
14	
15	//Output yang diperlukan adalah "Nama: Roi", jadi
16	kata Pegawai dihapus aja.
17	System.out.println("Nama: " + p1.getNama());
18	System.out.println("Asal: " + p1.getAsal());
20	System.out.println("Jabatan: " + p1.jabatan);
21	//Untuk mencetak kata tahun diakhir kalimat, jadi
22	tambahkan (+ " tahun")
23	System.out.println("Umur: " + p1.umur + " tahun");
24	}
25	}
26	

Tabel 29.3 Source Code Soal 3 Main

B. Output Program

```

C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=13354:C:\Program Files\JetBrains\IntelliJ IDEA 202
Nama: Roi
Asal: Kingdom of Orvel
Jabatan: Assasin
Umur: 17 tahun
Process finished with exit code 0

```

Gambar 13.3 Output Screenshoot Soal 3

C. Pembahasan

a. Pegawai

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2-8, 12, dan 15 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 2 dan 18 : {}, penanda awal dan akhir *block code*.

Line 3-8 : *String*, tipe data teks yang berisikan karakter lebih.

Line 7 : *int*, tipe data *integer* atau menerima bilangan bulat.

Line 4 dan 11: // atau *comments* membuat note atau komentar pada *workspace*.

Line 9 dan 13: *return* untuk mengembalikan nilai.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 26 : {}, penanda awal dan akhir *block code*.

Line 4 : *Static*, fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. *Void*, tidak ada nilai balik atau *return*. *String[] args*, adalah parameters.

Line 6, 10, 16, dan 22 : // atau *comments* membuat note atau komentar pada *workspace*.

Line 18-20, dan 24 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

MODUL 3: ENKAPULASI DAN COLLECTION

SOAL 1

Buatlah program yang mengimplementasikan enkapsulasi dan collection dengan ketentuan seperti berikut.

- a. Buatlah kelas dengan nama Dadu.
- b. Terdapat sebuah method acakNilai() yang akan memberikan nilai acak ketika objek dadu diinisiasi dengan nilai antara 1 – 6.
- c. Gunakan collection dengan tipe LinkedList untuk menyimpan objek dadu yang diinputkan.
- d. Program bersifat dinamis
- e. Input baris pertama adalah banyaknya jumlah dadu.
- f. Output adalah nilai tiap objek dadu yang telah diinisiasi
- g. Output paling akhir adalah total jumlah nilai semua objek dadu.
- h. Karena nilai dadu yang dihasilkan acak, maka nilai dadu output tidak harus sama dengan contoh table dibawah.

Input	Output
3	Dadu ke-1 bernilai 2 Dadu ke-2 bernilai 3 Dadu ke-3 bernilai 5 Total nilai dadu keseluruhan 10
4	Dadu ke-1 bernilai 1 Dadu ke-2 bernilai 3 Dadu ke-3 bernilai 6 Dadu ke-4 bernilai 1 Total nilai dadu keseluruhan 11
1	Dadu ke-1 bernilai 2 Total nilai dadu keseluruhan 2

Tabel 30.1 Soal 1

A. Source Code

1	package PRAK301_soal1;
2	
3	import java.util.LinkedList;
4	
5	public class Dadu {
6	public int input;
7	public int min = 1;
8	public int max = 6;
9	int total;
10	
11	public void setInput(int input){this.input = input;}
12	
13	void acakNilai(){
14	LinkedList<Integer> ll = new LinkedList<Integer>();
15	

16	ll.add(1);ll.add(2);ll.add(3);ll.add(4);ll.add(5);ll.add(6);
17	for (int i = 0; i < input; i++){
18	int random_int =
20	(int)Math.floor(Math.random()*(max-min+1)+min);
21	System.out.println("Dadu ke-" + (i + 1) + "
22	bernilai "+ random_int);
23	total += random_int;
24	}
25	System.out.println("Total nilai dadu keseluruhan "
26	+total);
27	}
28	}

Tabel 31.1 Source Code Soal 1 Dadu

1	package PRAK301_soal1;
2	
3	import java.util.*;
4	
5	public class Main {
6	public static void main(String[] args) {
7	Scanner input = new Scanner(System.in);
8	Dadu dd = new Dadu();
9	int masukkan = input.nextInt();
10	
11	dd.setInput(masukkan);
12	dd.acakNilai();
13	}
14	}

Tabel 32.1 Source Code Soal 1 Main

B. Output Program

```
C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=21070:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin"
J
Dadu ke-1 bernilai 4
Dadu ke-2 bernilai 3
Dadu ke-3 bernilai 4
Total nilai dadu keseluruhan 11
Process finished with exit code 0
```

Gambar 14.1 Screenshoot Output Soal 1

C. Pembahasan

a. Dadu

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5-9, dan 11: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 5 dan 28 : { }, penanda awal dan akhir *block code*.

Line 6-9 : *int*, tipe data *integer* atau menerima bilangan bulat.

Line 14 : *LinkedList*, suatu cara untuk menyimpan data dengan struktur.

Line 20 : *Math.floor*, sebuah metode mengembalikan nilai *double* sama dengan bilangan bulat terbesar.

Line 25 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5-6 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 5 dan 14 : { }, penanda awal dan akhir *block code*.

Line 7, : *Scanner* untuk menginput data.

Line 9 : *int*, tipe data *integer* atau menerima bilangan bulat.

SOAL 2

Buatlah program yang mengimplementasikan enkapsulasi dan collection dengan ketentuan seperti berikut.

- a. Buatlah kelas dengan nama Negara.
- b. Terdapat 5 attribute pada kelas Negara, yaitu nama, jenis kepemimpinan, nama pemimpin, tanggal kemerdekaan, bulan kemerdekaan, tahun kemerdekaan
- c. Inisiasi nilai attribute dilakukan pada constructor.
- d. Gunakan collection dengan tipe LinkedList untuk menyimpan objek Negara
- e. Gunakan collection dengan tipe HashMap untuk menyimpan daftar nama bulan. Nama bulan diambil dari hashmap berdasarkan angka bulan yang diinputkan.
- f. Program bersifat dinamis.
- g. Input baris pertama adalah banyaknya negara. Input baris berikutnya adalah data negara. Jika jenis kepemimpinan adalah monarki maka tidak perlu menginputkan tanggal kemerdekaan.
- h. Output adalah detail dari setiap objek negara yang telah diinputkan

Input
2 Indonesia presiden Joko Widodo 17 8 1945 Palestina presiden Mahmoud Abbas 15 11 1988
Output
Negara Indonesia mempunyai Presiden bernama Joko Widodo Deklarasi Kemerdekaan pada Tanggal 17 Agustus 1945 Negara Palestina mempunyai Presiden bernama Mahmoud Abbas Deklarasi Kemerdekaan pada Tanggal 15 November 1988
Input
3 Thailand monarki Maha Vajiralongkorn Indonesia presiden Joko Widodo 17 8 1945 Malaysia perdana menteri Ismail Sabri Yaakob 31 8 1957
Output
Negara Thailand mempunyai Raja bernama Maha Vajiralongkorn Negara Indonesia mempunyai Presiden bernama Joko Widodo Deklarasi Kemerdekaan pada Tanggal 17 Agustus 1945 Negara Malaysia mempunyai Perdana Menteri bernama Ismail Sabri Yaakob Deklarasi Kemerdekaan pada Tanggal 31 Agustus 1957

Tabel 33.2 Soal 2

A. Source Code

```
1 package PRAK302_soal2;
2
3 import java.util.*;
4
5 public class Negara {
6     private String nama, jenis, nama_p;
7     private int tanggal, bulan, tahun;
8
9     //    Konstruktor
10    public Negara(String nama, String jenis, String
11 nama_p, int tanggal, int bulan, int tahun){
12        this.nama = nama;
13        this.jenis = jenis;
14        this.nama_p = nama_p;
15        this.tanggal = tanggal;
16        this.bulan = bulan;
17        this.tahun = tahun;
18    }
19
20
21    //    Encapsulation
22    public int getBulan(){
23        return bulan;
24    }
25
26    void getData(){
27        Scanner scn = new Scanner(System.in);
28        nama=scn.nextLine();
29        jenis=scn.nextLine();
30        nama_p=scn.nextLine();
31        if
32 (jenis.equalsIgnoreCase("presiden")||jenis.equalsIgnoreCase
33 noreCase("perdana menteri")) {
34            tanggal = scn.nextInt();
35            bulan = scn.nextInt();
36            tahun = scn.nextInt();
37        }
38    }
39
40    void tampilkan(){
41        HashMap<Integer, String> month = new
42 HashMap<Integer,String>();
43        month.put(1, "Januari");
44        month.put(2, "Februari");
45        month.put(3, "Maret");
46        month.put(4, "April");
47        month.put(5, "Mei");
48        month.put(6, "Juni");
```

49	month.put(7, "Juli");
50	month.put(8, "Agustus");
51	month.put(9, "September");
52	month.put(10, "Oktober");
53	month.put(11, "November");
54	month.put(12, "Desember");
55	if
56	(jenis.equalsIgnoreCase("presiden") jenis.equalsIg
57	noreCase("perdana menteri")) {
58	System.out.println("Negara "+nama+"
59	mempunyai "+jenis+" bernama "+nama_p);}
60	else if
61	(jenis.equalsIgnoreCase("monarki")){
62	System.out.println("Negara "+nama+"
63	mempunyai Raja"+" bernama "+nama_p+"\n");
64	}
65	if
66	(jenis.equalsIgnoreCase("presiden") jenis.equalsIg
67	noreCase("perdana menteri")) {
68	System.out.println("Deklarasi
69	Kemerdekaan pada Tanggal "+tanggal+"
70	" "+month.get(getBulan())+" "+tahun+"\n");}
	}
	}

Tabel 34.2 Source Code Soal 2 Negara

1	package PRAK302_soal2;
2	
3	import java.util.*;
4	import java.util.Scanner;
5	
6	public class Main {
7	public static void main(String[] args) {
8	String nama = "", jenis="", nama_p="";
9	int tanggal=0, bulan=0, tahun=0;
10	Scanner input = new Scanner(System.in);
11	Negara negara = new Negara(nama, jenis, nama_p,
12	tanggal, bulan, tahun);
13	LinkedList<Negara> llNegara = new
14	LinkedList<Negara>();
15	
16	int ulang;
17	ulang = input.nextInt();
18	
19	for (int i=0; i<ulang; i++){
20	Negara c = new Negara(nama, jenis, nama_p,
21	tanggal, bulan, tahun);

22	<code>c.getData();</code>
23	<code>llNegara.add(c);</code>
24	<code>}</code>
25	
26	<code>for (int i = 0; i < llNegara.size(); i++){</code>
27	<code>Negara nIndex = llNegara.get(i);</code>
28	<code>nIndex.tampilkan();</code>
29	<code>}</code>
30	<code>}</code>
31	<code>}</code>

Tabel 35.2 Source Code Soal 2 Main

B. Output Program

```

C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar-20846:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin\
2
Indonesia
Presiden
Joko Widodo
17
8
1945
Palestina
Presiden
Mahmoud Abbas
15
13
1988
Negara Indonesia mempunyai Presiden bernama Joko Widodo
Deklarasi Kemerdekaan pada Tanggal 17 Agustus 1945
Negara Palestina mempunyai Presiden bernama Mahmoud Abbas
Deklarasi Kemerdekaan pada Tanggal 15 November 1988
Process finished with exit code 0

```

Gambar 15.2 Output Screenshoot Soal 2

C. Pembahasan

a. Negara

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5, 10, dan 22 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6-7 : *Private*, kunci pada java yang dapat diakses oleh kelas itu sendiri.

Line 5 dan 70 : `{ }`, penanda awal dan akhir *block code*.

Line 10 : *String*, tipe data teks yang berisikan karakter lebih.

Line 11 : *int*, tipe data *integer* atau menerima bilangan bulat.

Line 41 : *HashMap*, adalah *class* yang berisi sekumpulan pasangan nilai atau *value* dan *key*.

Line 43-54 : *month.put*, metode menambahkan *key* bulan.

Line 55 dan 60 : *If Else* untuk memberikan kondisi *true* atau *false*.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3-4: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 6-7 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 31 : {}, penanda awal dan akhir *block code*.

Line 8 : *String*, tipe data teks yang berisikan karakter lebih.

Line 9 : *int*, tipe data *integer* atau menerima bilangan bulat.

Line 10, : *Scanner* untuk menginput data.

Line 13-14 : *LinkedList*, suatu cara untuk menyimpan data dengan struktur.

Line 19 dan 26 : *for*, untuk perulangan pada *statement* secara berulang – ulang selama terpenuhi.

SOAL 3

Buatlah program yang mengimplementasikan enkapsulasi dan collection dengan ketentuan seperti berikut.

- a. Buatlah kelas dengan nama Mahasiswa
- b. Terdapat 2 atribut pada kelas, yaitu nama dan nim
- c. Inisiasi nilai atribut dilakukan pada konstruktor
- d. Terdapat getter untuk 2 atribut tadi
- e. Gunakan collection dengan tipe ArrayList untuk menyimpan objek Mahasiswa
- f. Program bersifat dinamis dan interaktif
- g. Program dapat melakukan operasi seperti berikut:
 - Tambah Mahasiswa, menambahkan objek baru ke ArrayList
 - Hapus Mahasiswa, menghapus data mahasiswa dari ArrayList berdasarkan NIM
 - Cari Mahasiswa, menampilkan data mahasiswa berdasarkan NIM yang di input oleh pengguna
 - Tampilkan seluruh data Mahasiswa, menampilkan seluruh data mahasiswa dari ArrayList. Tampilkan nama dan NIM mahasiswa
 - Keluar, program berhenti dan seluruh data pada ArrayList dihapus, ketika program dijalankan ulang, ArrayList masih kosong.

Output
Menu: 1. Tambah Mahasiswa 2. Hapus Mahasiswa berdasarkan NIM 3. Cari Mahasiswa berdasarkan NIM 4. Tampilkan Daftar Mahasiswa 0. Keluar Pilihan: 1 Masukkan Nama Mahasiswa: Bachrul Uluum Masukkan NIM Mahasiswa (harus unik): 2010817210025 Mahasiswa Bachrul Uluum ditambahkan. Menu: 1. Tambah Mahasiswa 2. Hapus Mahasiswa berdasarkan NIM

<p>3. Cari Mahasiswa berdasarkan NIM</p> <p>4. Tampilkan Daftar Mahasiswa</p> <p>0. Keluar</p> <p>Pilihan: 1</p> <p>Masukkan Nama Mahasiswa: Muhammad Aulia Akbar</p> <p>Masukkan NIM Mahasiswa (harus unik): 2010817210023</p> <p>Mahasiswa Muhammad Aulia Akbar ditambahkan.</p> <p>Menu:</p> <p>1. Tambah Mahasiswa</p> <p>2. Hapus Mahasiswa berdasarkan NIM</p> <p>3. Cari Mahasiswa berdasarkan NIM</p> <p>4. Tampilkan Daftar Mahasiswa</p> <p>0. Keluar</p> <p>Pilihan: 4</p> <p>Daftar Mahasiswa:</p> <p>NIM: 2010817210025, Nama: Bachrul Uluum</p> <p>NIM: 2010817210023, Nama: Muhammad Aulia Akbar</p> <p>1. Tambah Mahasiswa</p> <p>2. Hapus Mahasiswa berdasarkan NIM</p> <p>3. Cari Mahasiswa berdasarkan NIM</p> <p>4. Tampilkan Daftar Mahasiswa</p> <p>0. Keluar</p> <p>Pilihan: 2</p> <p>Masukkan NIM Mahasiswa yang akan dihapus: 2010817210025</p> <p>Mahasiswa dengan NIM 2010817210025 dihapus.</p> <p>Menu:</p> <p>1. Tambah Mahasiswa</p> <p>2. Hapus Mahasiswa berdasarkan NIM</p> <p>3. Cari Mahasiswa berdasarkan NIM</p> <p>4. Tampilkan Daftar Mahasiswa</p> <p>0. Keluar</p>
<p>Pilihan: 0</p> <p>Terima kasih!</p>

Tabel 36.3 Soal 3

A. Source Code

1	package PRAK303_soal3;
2	
3	class Mahasiswa {
4	private String nama;
5	private String nim;
6	
7	public Mahasiswa(String nama, String nim) {
8	this.nama = nama;
9	this.nim = nim;
10	}
11	
12	public String getNama() {
13	return nama;

14	}
15	
16	public String getNim() {
17	return nim;
18	}
19	}

Tabel 37.3 Source Code Soal 3 Kopi

1	package PRAK303_soal3;
2	
3	import java.util.ArrayList;
4	import java.util.Scanner;
5	
6	public class Main {
7	public static void main(String[] args) {
8	Scanner sc = new Scanner(System.in);
9	ArrayList<Mahasiswa> mahasiswaList = new
10	ArrayList<>();
11	
12	while (true) {
13	System.out.println("Menu:");
14	System.out.println("1. Tambah Mahasiswa");
15	System.out.println("2. Hapus Mahasiswa
16	berdasarkan NIM");
17	System.out.println("3. Cari Mahasiswa
18	berdasarkan NIM");
20	System.out.println("4. Tampilkan Daftar
21	Mahasiswa");
22	System.out.println("0. Keluar");
23	System.out.print("Pilihan: ");
24	int pilihan = sc.nextInt();
25	
26	if (pilihan == 0) {
27	mahasiswaList.clear();
28	System.out.println("Terima kasih!");
29	break;
30	} else if (pilihan == 1) {
31	System.out.print("Masukkan Nama Mahasiswa:
32	");
33	sc.nextLine();
34	String nama = sc.nextLine();
35	System.out.print("Masukkan NIM Mahasiswa
36	(harus unik): ");
37	String nim = sc.nextLine();
38	
39	boolean isNimExist = false;
40	for (Mahasiswa mahasiswa : mahasiswaList) {

```

41         if (mahasiswa.getNim().equals(nim)) {
42             isNimExist = true;
43             break;
44         }
45     }
46
47     if (isNimExist) {
48         System.out.println("NIM sudah
49 digunakan.");
50     } else {
51         Mahasiswa mahasiswa = new
52 Mahasiswa(nama, nim);
53         mahasiswaList.add(mahasiswa);
54         System.out.println("Mahasiswa " + nama +
55 " ditambahkan.");
56     }
57     } else if (pilihan == 2) {
58         System.out.print("Masukkan NIM Mahasiswa
59 yang akan dihapus: ");
60         String nim = sc.next();
61
62         boolean isNimExist = false;
63         for (Mahasiswa mahasiswa : mahasiswaList) {
64             if (mahasiswa.getNim().equals(nim)) {
65                 isNimExist = true;
66                 mahasiswaList.remove(mahasiswa);
67                 System.out.println("Mahasiswa dengan
68 NIM " + nim + " dihapus.");
69                 break;
70             }
71         }
72
73         if (!isNimExist) {
74             System.out.println("NIM tidak
75 ditemukan.");
76         }
77     } else if (pilihan == 3) {
78         System.out.print("Masukkan NIM Mahasiswa
79 yang akan dicari: ");
80         String nim = sc.next();
81
82         boolean isNimExist = false;
83         for (Mahasiswa mahasiswa : mahasiswaList) {
84             if (mahasiswa.getNim().equals(nim)) {
85                 isNimExist = true;
86                 System.out.println("NIM: " +
87 mahasiswa.getNim() + ", Nama: " + mahasiswa.getNama());
88                 break;

```

89	}
90	}
91	if (!isNimExist) {
92	System.out.println("NIM tidak
93	ditemukan.");
94	}
95	} else if (pilihan == 4) {
96	System.out.println("Daftar Mahasiswa:");
97	for (Mahasiswa mahasiswa : mahasiswaList) {
98	System.out.println("NIM: " +
99	mahasiswa.getNim() + ", Nama: " + mahasiswa.getNama());
100	}
101	} else {
102	System.out.println("Pilihan tidak valid.");
103	}
104	System.out.println();
105	}
106	}
107	}
108	
109	
110	
111	
112	
113	
114	
115	
116	
117	
118	
119	
120	
121	
122	

Tabel 38.3 Source Code Soal 3 Main

B. Output Program

```
C:\Users\User\jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=20884:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath C:\Users\User\IdeaProjects\PENROGRAMAN-1\out\production\MODUL3_PRAK303_soa13\Main
Menu:
1. Tambah Mahasiswa
2. Hapus Mahasiswa berdasarkan NIM
3. Cari Mahasiswa berdasarkan NIM
4. Tampilkan Daftar Mahasiswa
5. Keluar
Pilihan: 1
Masukkan Nama Mahasiswa: Bachrul Ulum
Masukkan NIM Mahasiswa (harus unik): 2010817210025
Mahasiswa Bachrul Ulum ditambahkan.

Menu:
1. Tambah Mahasiswa
2. Hapus Mahasiswa berdasarkan NIM
3. Cari Mahasiswa berdasarkan NIM
4. Tampilkan Daftar Mahasiswa
5. Keluar
Pilihan: 1
Masukkan Nama Mahasiswa: Muhammad Aulia Akbar
Masukkan NIM Mahasiswa (harus unik): 2010817210023
Mahasiswa Muhammad Aulia Akbar ditambahkan.

Menu:
1. Tambah Mahasiswa
2. Hapus Mahasiswa berdasarkan NIM
3. Cari Mahasiswa berdasarkan NIM
4. Tampilkan Daftar Mahasiswa
5. Keluar
Pilihan: 4
Daftar Mahasiswa:
NIM: 2010817210025, Nama: Bachrul Ulum
NIM: 2010817210023, Nama: Muhammad Aulia Akbar

Menu:
1. Tambah Mahasiswa
2. Hapus Mahasiswa berdasarkan NIM
3. Cari Mahasiswa berdasarkan NIM
4. Tampilkan Daftar Mahasiswa
5. Keluar
Pilihan: 2
Masukkan NIM Mahasiswa yang akan dihapus: 2010817210025
Mahasiswa dengan NIM 2010817210025 dihapus.

Menu:
1. Tambah Mahasiswa
2. Hapus Mahasiswa berdasarkan NIM
3. Cari Mahasiswa berdasarkan NIM
4. Tampilkan Daftar Mahasiswa
5. Keluar
Pilihan: 0
Terima kasih!

Process finished with exit code 0
```

Gambar 16.3 Output Screenshoot Soal 3

C. Pembahasan

a. Kopi

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 4-5 : *Private*, kunci pada java yang dapat diakses oleh kelas itu sendiri.

Line 7,12, dan 16 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 5 dan 70 : {}, penanda awal dan akhir *block code*.

Line 4-5, 7, 12, dan 16 : *String*, tipe data teks yang berisikan karakter lebih.

Line 13 dan 17 : *return* untuk mengembalikan nilai.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3-4: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 6-7 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 107 : {}, penanda awal dan akhir *block code*.

Line 8, : *Scanner* untuk menginput data.

Line 9: *ArrayList*, untuk menyimpan beberapa nilai dalam satu variable.

Line 12 : *While*, perulangan yang selalu dilakukan, jika perulangan tidak diketahui.

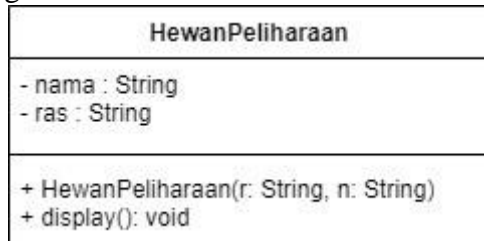
Line 13-23, 28, 31, 35, 48, 54, 58, 67, 74, 78, 86, 98, 102, dan 104 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

Line 26 dan 30 : *If Else* untuk memberikan kondisi *true* atau *false*.

MODUL 4: PEWARISAN/INHERITANCE & DIAGRAM KELAS

SOAL 1

Diberikan class diagram seperti berikut: (isi program harus sesuai dengan class diagram)Buatlah kelas dengan nama Dadu.



Implementasikan class diagram diatas menjadi program menggunakan Bahasa pemrogramanjava dengan ketentuan seperti berikut:

- i. Inisiasi objek **harus** menggunakan konstruktor.
- j. Output ditampilkan **dengan hanya** memanggil method display.
- k. Tidak boleh ada fungsi mencetak di class main.

Input
Nama Hewan Peliharaan: Leo Ras: Kucing Anggora
Output
Detail Hewan Peliharaan: Nama hewan peliharaanku adalah : Leo Dengan ras : Kucing Anggora

Tabel 39.1 Soal 1

A. Source Code

```

1 package soall;
2
3 public class peliharaan {
4     private String nama;
5     private String ras;
6
7     public peliharaan(String r, String n) {
8         ras = r;
9         nama = n;
10    }
11    public void display() {
12        System.out.println("Detail Hewan Peliharaan:");
13        System.out.println("Nama hewan peliharaanku adalah :
14    " + nama);
15        System.out.println("Dengan ras : " + ras);
  
```

16	}
17	
18	//ryann
20	}
21	

Tabel 40.1 Source Code Soal 1 Peliharaan

1	package soall;
2	
3	import java.util.Scanner;
4	public class Main {
5	public static void main(String[] args) {
6	Scanner isian = new Scanner(System.in);
7	
8	System.out.print("Nama Hewan Peliharaan: ");
9	String namaHewan = isian.nextLine();
10	
11	System.out.print("Ras: ");
12	String rasHewan = isian.nextLine();
13	
14	peliharaan hewanPeliharaan = new peliharaan(rasHewan,
15	namaHewan);
16	isian.close();
17	hewanPeliharaan.display();
18	}
20	}

Tabel 41.1 Source Code Soal 1 Man

B. Output Program

```
C:\Users\User\.jdk\openjdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=21070:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin"
J
Dadu ke-1 bernilai 4
Dadu ke-2 bernilai 3
Dadu ke-3 bernilai 6
Total nilai dadu keseluruhan 13
Process finished with exit code 0
```

Gambar 17.1 Screenshoot Output Soal 1

C. Pembahasan

a. Peliharaan

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 3 dan 11: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 21 : {}, penanda awal dan akhir *block code*.

Line 4 dan 5 : *String*, tipe data teks yang berisikan karakter lebih.

Line 12, 14, dan 16 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

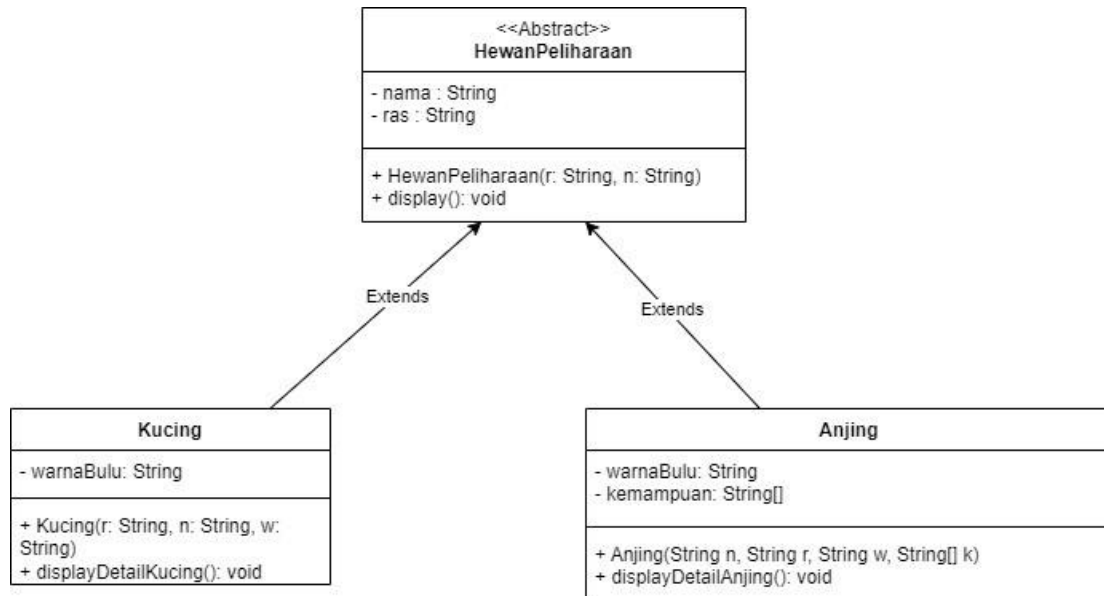
Line 5 dan 20 : { }, penanda awal dan akhir *block code*.

Line 7, : *Scanner* untuk menginput data.

Line 9 dan 12 : *String*, tipe data teks yang berisikan karakter lebih.

Line 9 : *int*, tipe data *integer* atau menerima bilangan bulat.

SOAL 2



Implementasikan class diagram diatas menjadi program menggunakan Bahasa pemrograman java dengan ketentuan seperti berikut:

- Inisiasi objek harus menggunakan konstruktor.
- Isi baris program harus sesuai dengan apa yang digambarkan di dalam kelas diagram.
- Menampilkan detail dari class Kucing maupun Anjing harus melalui method yang ada,tidak boleh menggunakan fungsi cetak dari java

Isi detail dari class Kucing dan Anjing harus memanggil method display pada parent classterlebih dahulu.

Input
Pilih jenis hewan yang ingin diinputkan: 1 = Kucing 2 = Anjing Masukkan pilihan: 1 Nama hewan peliharaan: lolo Ras: Anggora



Warna Bulu: abu-abu

Output
Detail Hewan Peliharaan: Nama hewan peliharaanku adalah : lolo Dengan ras : Anggora Memiliki warna bulu : abu-abu

Input
Pilih jenis hewan yang ingin diinputkan: 1 = Kucing 2 = Anjing Masukkan pilihan: 2 Nama hewan peliharaan: andi Ras: cihuahua Warna Bulu: putih Kemampuan : berenang, berjabat tangan, melompat

Output
Detail Hewan Peliharaan: Nama hewan peliharaanku adalah : andi Dengan ras : cihuahua Memiliki warna bulu : putih Memiliki kemampuan : berenang berjabat tangan melompat



Tabel 42.2 Soal 2

A. Source Code

```
1 package soal2;
2
3 public abstract class HewanPeliharaan {
4     private String nama;
5     private String ras;
6
7     public HewanPeliharaan(String r, String n) {
8         ras = r;
9         nama = n;
10    }
11
12    public void display() {
13        System.out.println("Detail Hewan Peliharaan:");
14        System.out.println("Nama hewan peliharaanku adalah :
15 " + nama);
16        System.out.println("Dengan ras : " + ras);
17    }
18 }
19
20
```

Tabel 43.2 Source Code Soal 2 HewanPeliharaan

```
1 package soal2;
2
3 public class Anjing extends HewanPeliharaan {
4     private String warnaBulu;
5     private String[] kemampuan;
6
7     public Anjing(String n, String r, String w, String[] k)
8     {
9         super(r, n);
10        warnaBulu = w;
11        kemampuan = k;
12    }
13
14    public void displayDetailAnjing() {
15        super.display();
16        System.out.println("Memiliki warna bulu : " +
17 warnaBulu);
18        System.out.print("Kemampuan : ");
19        for (String k : kemampuan) {
20            System.out.print(" "+k);
21        }
22    }
23 }
24
```

Tabel 44.2 Source Code Soal 2 Anjing

1	package soal2;
2	
3	public class Kucing extends HewanPeliharaan {
4	private String warnaBulu;
5	
6	public Kucing(String r, String n, String w) {
7	super(r, n);
8	warnaBulu = w;
9	}
10	
11	public void displayDetailKucing() {
12	super.display();
13	System.out.println("Memiliki warna bulu : " +
14	warnaBulu);
15	}
16	}

Tabel 45.2 Source Code Soal 2 Kucing

1	package soal2;
2	
3	import java.util.Scanner;
4	
5	public class Main {
6	public static void main(String[] args) {
7	Scanner masukan = new Scanner(System.in);
8	System.out.println("Pilih jenis hewan yang ingin
9	diinputkan:");
10	System.out.println("1 = Kucing");
11	System.out.println("2 = Anjing");
12	System.out.print("Masukkan pilihan: ");
13	int pilihan = masukan.nextInt();
14	
15	if (pilihan == 1) {
16	masukan.nextLine();
17	System.out.print("Nama hewan peliharaan: ");
18	String namaKucing = masukan.nextLine();
19	System.out.print("Ras: ");
20	String rasKucing = masukan.nextLine();
21	System.out.print("Warna Bulu: ");
22	String warnaBuluKucing = masukan.nextLine();
23	
24	Kucing kucing = new Kucing(rasKucing,
25	namaKucing, warnaBuluKucing);
26	kucing.displayDetailKucing();

27	} else if (pilihan == 2) {
28	masukan.nextLine();
29	System.out.print("Nama hewan peliharaan: ");
30	String namaAnjing = masukan.nextLine();
31	System.out.print("Ras: ");
32	String rasAnjing = masukan.nextLine();
33	System.out.print("Warna Bulu: ");
34	String warnaBuluAnjig = masukan.nextLine();
35	System.out.print("Kemampuan: ");
36	String kemampuanStr = masukan.nextLine();
37	String[] kemampuan = kemampuanStr.split(",");
38	
39	Anjing anjing = new Anjing(namaAnjing,
40	rasAnjing, warnaBuluAnjig, kemampuan);
41	anjing.displayDetailAnjing();
42	} else {
43	System.out.println("Pilihan tidak valid");
44	}
45	
46	masukan.close();
47	}
48	}
49	
50	
51	
52	
53	
54	
55	

Tabel 46.2 Source Code Soal 2 Main

B. Output Program

```

C:\Users\User\jdk-20.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\lib\idea_rt.jar=20846:C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.1\bin"
2
Indonesia
Presiden
Joko Widodo
17
8
1945
Palestina
Presiden
Mahmoud Abbas
15
11
1988
Negara Indonesia mempunyai Presiden bernama Joko Widodo
Deklarasi Kemerdekaan pada Tanggal 17 Agustus 1945

Negara Palestina mempunyai Presiden bernama Maheoud Abbas
Deklarasi Kemerdekaan pada Tanggal 15 November 1988

Process finished with exit code 0

```

Gambar 18.2 Output Screenshoot Soal 2

C. Pembahasan

a. HewanPeliharaan

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 3, 7, dan 11: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6-7 : *Private*, kunci pada java yang dapat diakses oleh kelas itu sendiri.

Line 5 dan 20 : { }, penanda awal dan akhir *block code*.

Line 10 : *String*, tipe data teks yang berisikan karakter lebih.

Line 20 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

b. Anjing

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3-4: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 3,7, dan 14 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 31 : { }, penanda awal dan akhir *block code*.

Line 8 : *String*, tipe data teks yang berisikan karakter lebih.

Line 7, 14, dan 21 : *System.out.println* mencetak sesuatu dari program sebagai *output*.

c. Kucing

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3, 6, dan 11 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 16 : { }, penanda awal dan akhir *block code*.

Line 8 : *String*, tipe data teks yang berisikan karakter lebih.

d. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3-4: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6, : *Scanner* untuk menginput data.

Line 5 dan 51 : { }, penanda awal dan akhir *block code*.

Line 7 - 49 : *String*, tipe data teks yang berisikan karakter lebih.

Line 7 - 49 : *int*, tipe data *integer* atau menerima bilangan bulat.

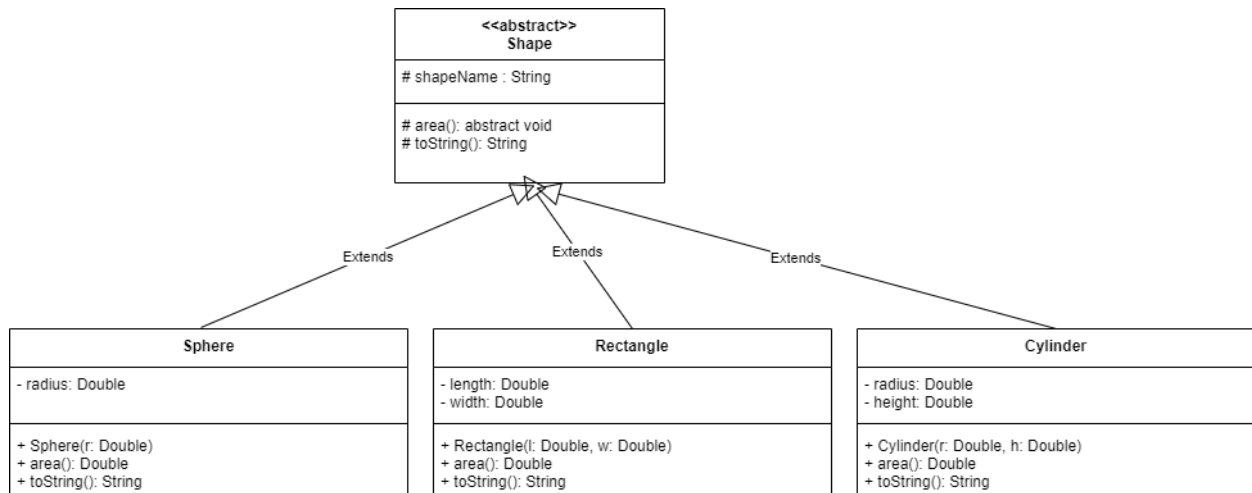
Line 15 dan 29 : *If Else* untuk memberikan kondisi *true* atau *false*.

MODUL 5: POLIMORFISME

SOAL

Pada praktikum kali ini anda akan diminta untuk membuat sebuah program yang dapat menghitung banyaknya liter cat yang digunakan untuk mewarnai bentuk ruang yang beragam.

Buatlah sebuah hierarki kelas abstrak Shape dimana memiliki 3 kelas anak yaitu Sphere, Rectangle, dan Cylinder seperti ditunjukkan oleh diagram kelas berikut.



Method `area()` digunakan untuk menghitung luas masing-masing objek. Berikut adalah formula yang digunakan untuk menghitung luas masing-masing bangun yang harus diimplementasikan.

Sphere: $4 \times \pi \times radius^2$

Rectangle: $length \times width$

Cylinder: $\pi \times radius^2 \times height$

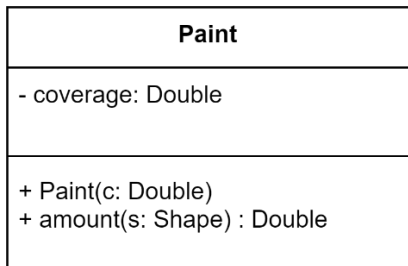
Method toString() digunakan untuk mengembalikan nilai String dari nama bangun

Berikut adalah ilustrasi dari kelas Sphere.java. Implementasikan kelas lainnya untuk Shape, Rectangle dan Cyclinder

Contoh Ilustrasi Sphere.java

```
public class Sphere extends Shape
{
private double radius; //radius in feet
// -----
// Constructor: Sets up the sphere.
// -----
public Sphere(double r)
{
super("Sphere"); radius = r;
}
// -----
// Returns the surface area of the sphere.
// -----
public double area()
{
return 4*Math.PI*(radius*radius);
}
// -----
// Returns the sphere as a String.
// -----
public String toString()
{
return super.toString() + " of radius " + radius;
}
}
```


Selanjutnya, Buatlah kelas Paint.java seperti ditunjukkan diagram kelas berikut.



Method amount digunakan untuk menghitung banyaknya liter cat yang digunakan dengan persamaan berikut:

$$\text{amount of paint} = \frac{\text{area of shape}}{\text{coverage}}$$

Lengkapi kode dibawah supaya menghasilkan keluaran yang diinginkan

Paint.java
<pre>public class Paint { private double coverage; //number of square feet per gallon // ----- // Constructor: Sets up the paint object. // ----- public Paint(double c) { coverage = c; } // ----- // Returns the amount of paint (number of gallons) // needed to paint the shape given as the parameter. // ----- public double amount(Shape s) { System.out.println ("Computing amount for " + s); return 0; } }</pre>

Terakhir, Buatlah kelas main bernama PaintThings.java. Tambahkan beberapa hal berikut agar program berjalan sesuai yang diinginkan.

1. Instansiasi 3 bentuk objek:
 - a. objek bernama deck berbentuk persegi panjang dengan ukuran Panjang 20cm dan lebar 30cm.
 - b. objek bernama bigBall berbentuk bola dengan ukuran radius 15cm.
 - c. objek bernama tank berbentuk silinder dengan ukuran radius 10cm dan tinggi 30cm.
2. Panggil fungsi yang tepat agar dapat menghitung jumlah cat yang diperlukan.

Petunjuk untuk kelas main PaintThings.java

```
import      java.text.DecimalFormat;
public class PaintThings
{
    -----
    //
    // Creates some shapes and a Paint object
    // and prints the amount of paint needed
    // to paint each shape. -----
    //
    public static void main (String[] args)
    {
        final double COVERAGE = 350;
        Paint paint = new Paint(COVERAGE);
        Rectangle deck;

        Sphere      bigBall;
        Cylinder tank;

        double deckAmt, ballAmt, tankAmt;

        // Instantiate the three shapes to paint
```

```

System.out.println ("Big Ball " + fmt.format(ballAmt)); System.out.println
("Tank " + fmt.format(tankAmt));

}

}

```

1. Jalankan program dan perhatikan hasil untuk ketiga bentuk yang berbeda, screenshot hasil yang didapatkan dan lampirkan di dalam source code.
2. Simpan coding anda dengan nama package: **soal1**
3. Pastikan terdapat screenshoot pada repositori github

Tabel 47.1 Soal 1

A. Source Code

1	package soal1;
2	
3	import java.lang.Math;
4	public class sphere extends shape {
5	private double radius;
6	
7	public sphere(Double r){
8	super("Sphere");
9	radius = r;
10	}
11	
12	public Double area(){return
13	4*Math.PI*(radius*radius);}
14	
15	public String toString(){return super.toString()
16	+ " of radius "+radius;}
17	}

Tabel 48.1 Source Code Soal 1 Spehere

1	
2	package soal1;
3	
4	public abstract class shape {
5	protected String shapeName;
6	
7	public shape(String name) {
8	shapeName = name;
9	}
10	
11	protected abstract Double area();
12	
13	public String toString() {
14	return shapeName;
15	}
16	}

Tabel 49.1 Source Code Soal 1 Shape

1	package soal1;
2	
3	public class rectangle extends shape{
4	private Double length, width;
5	
6	public rectangle(Double l, Double w){
7	super("Rectangle");
8	length = l;
9	width = w;
10	}
11	
12	public Double area(){return length*width;}
13	
14	public String toString(){return super.toString()
15	+ " of length " + length + " and width "+width;}
16	}

Tabel 50.1 Source Code Soal 1 Rectangle

1	package soal1;
2	
3	public class cylinder extends shape{
4	private Double radius, height;
5	
6	public cylinder(Double r, Double h){
7	super("Cylinder");
8	radius = r;
9	height = h;

10	}
11	
12	public Double area(){return
13	Math.PI*(radius*radius)*height;}
14	
15	public String toString(){return super.toString()
16	+ " of radius " + radius + " and height "+height;}
17	}

Tabel 51.1 Source Code Soal 1 Cylinder

1	package soall1;
2	
3	public class paint {
4	private double coverage;
5	
6	public paint(double c){coverage = c;}
7	
8	public double amount(shape s){
9	System.out.println("Computing amount for "+
10	s);
11	return s.area()/coverage;
12	}
13	}

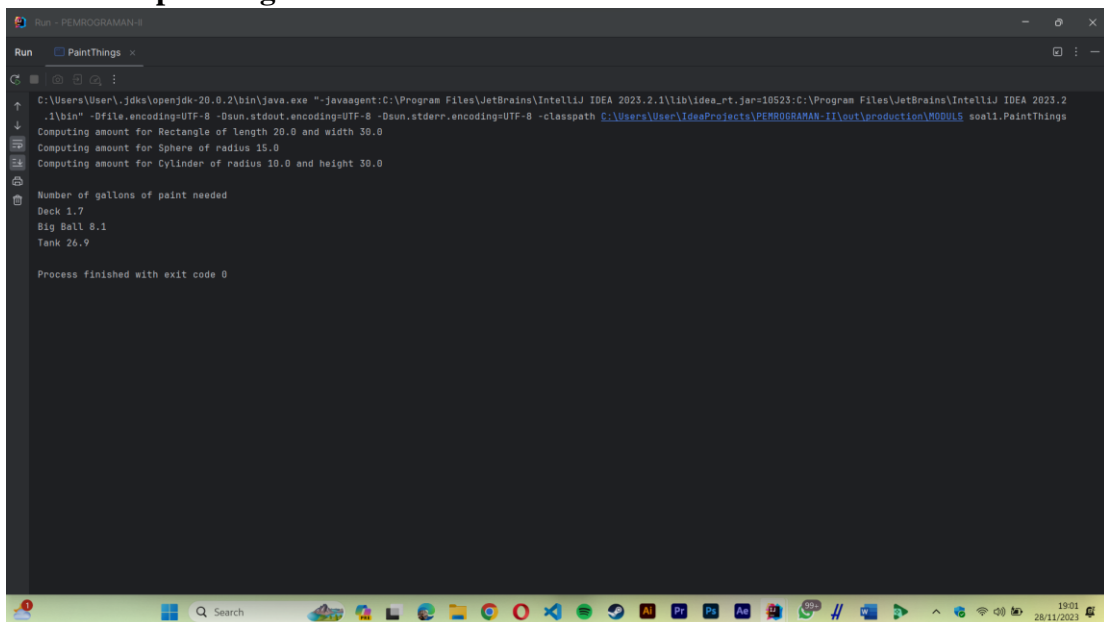
Tabel 52.1 Source Code Soal 1 Paint

1	package soall1;
2	
3	import java.text.DecimalFormat;
4	
5	public class PaintThings {
6	public static void main(String[] args) {
7	final double COVERAGE = 350;
8	paint paint = new paint(COVERAGE);
9	rectangle deck;
10	sphere bigBall;
11	cylinder tank;
12	double deckAmt, ballAmt, tankAmt;
13	
14	deck = new rectangle(20.0,30.0);
15	bigBall = new sphere(15.0);
16	tank = new cylinder(10.0,30.0);
17	
18	deckAmt = paint.amount(deck);
20	ballAmt = paint.amount(bigBall);

21	tankAmt = paint.amount(tank);
22	
23	DecimalFormat fmt = new
24	DecimalFormat("0.##");
25	System.out.println("\nNumber of gallons of
26	paint needed");
27	System.out.println("Deck " +
28	fmt.format(deckAmt));
29	System.out.println("Big Ball " +
30	fmt.format(ballAmt));
31	System.out.println("Tank " +
32	fmt.format(tankAmt));
33	}
34	}

Tabel 53.1 Source Code Soal 1 PaintThings

B. Output Program



Gambar 19.1 Screenshoot Output Soal 1

C. Pembahasan

a. Sphere

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 4, 7, 12, dan 15: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 4 dan 17 : { }, penanda awal dan akhir *block code*..

b. Shape

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3, 6, dan 12 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 15 : { }, penanda awal dan akhir *block code*.

Line 10 : *Abstract*, sebagai ekspresi konsep umum tempat kelas yang lebih spesifik dapat diturunkan.

Line 13 : *Return*, mengembalikan nilai dari sebuah function.

c. Rectangle

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3, 6, 12, dan 14: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 : *Extends* , kata kunci yang ditulis dengan kelas anak selama deklarasi kelas diikuti dengan nama kelas induk.

Line 3 dan 16: { }, penanda awal dan akhir *block code*.

d. Cylinder

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3 : *Extends* , kata kunci yang ditulis dengan kelas anak selama deklarasi kelas diikuti dengan nama kelas induk.

Line 3, 6, 12, dan 15: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 17 : { }, penanda awal dan akhir *block code*.

Line 15 : *Return*, mengembalikan nilai dari sebuah function.

e. Paint

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3, 6, dan 8: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 3 dan 13 : { }, penanda awal dan akhir *block code*.

Line 9 : *System.out.println* mencetak sesuatu dari program sebagai output.

Line 11 : *Return*, mengembalikan nilai dari sebuah function.

f. PaintThings

Line 1 : Package, untuk pengelompokan class atau interface menjadi satu unit.

Line 2: Import, suatu perintah untuk memasukkan suatu method atau perintah dalam Bahasa Java.

Line 4 dan 11: Public, kunci pada java yang dapat diakses kelas class lain.

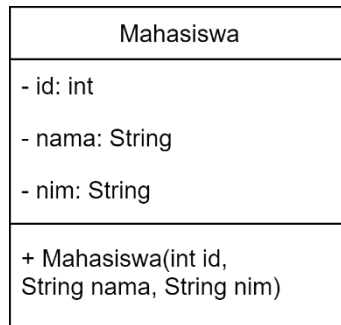
Line 4 dan 34 : {}, penanda awal dan akhir block code.

Line 24-31 : System.out.println mencetak sesuatu dari program sebagai output.

MODUL 6: GRAPHICAL USER INTEFACE

SOAL

Diberikan class diagram seperti berikut: (isi program harus sesuai dengan class diagram)



Implementasikan class diagram diatas menjadi class pada Bahasa pemrograman java. Class mahasiswa harus menerapkan setter dan getter.

Program harus menampilkan list data dalam bentuk tabel.

Kolom dari tabel adalah :

- NIM
- Nama

Kemudian buatlah 10 data secara *hardcode* untuk ditampilkan pada tabel Contoh program dapat dilihat sebagai berikut:

19	return id;
20	}
21	
22	public void setId(int id) {
23	this.id = id;
25	}
25	
26	public String getName() {
27	return name;
28	}
29	
30	public void setName(String name) {
31	this.name = name;
32	}
33	
34	public String getNim() {
35	return nim;
36	}
37	
38	public void setNim(String nim) {
39	this.nim=nim;
40	}
41	
42	}

Tabel 54.1 Source Code Soal 1 Mahasiswa

1	package com.example.modul_6;
2	
3	import javafx.application.Application;
4	import javafx.scene.Scene;
5	import javafx.scene.control.TableColumn;
6	import javafx.scene.control.TableView;
7	import
8	javafx.scene.control.cell.PropertyValueFactory;
9	import javafx.scene.layout.VBox;
10	import javafx.stage.Stage;
11	
12	public class Main extends Application {
13	
14	public static void main(String[] args) {
15	launch(args);
16	}
17	
18	@Override
19	public void start(Stage primaryStage) {
20	

```

21 TableView tableView = new TableView();
22
23 TableColumn<Mahasiswa, String> column1 =
24     new TableColumn<>("NIM");
25
26 column1.setCellValueFactory(
27     new PropertyValueFactory<>("nim"));
28
29
30 TableColumn<Mahasiswa, String> column2 =
31     new TableColumn<>("Nama");
32
33 column2.setCellValueFactory(
34     new PropertyValueFactory<>("name"));
35
36
37 tableView.getColumns().add(column1);
38 tableView.getColumns().add(column2);
39
40 tableView.getItems().add(
41     new Mahasiswa(1, "Audrey", "7000"));
42 tableView.getItems().add(
43     new Mahasiswa(2, "Dandellion",
44 "7001"));
45 tableView.getItems().add(
46     new Mahasiswa(3, "Deyya", "7002"));
47 tableView.getItems().add(
48     new Mahasiswa(4, "Aina", "7003"));
49 tableView.getItems().add(
50     new Mahasiswa(5, "Dawan", "7004"));
51 tableView.getItems().add(
52     new Mahasiswa(6, "Helenah",
53 "7005"));
54 tableView.getItems().add(
55     new Mahasiswa(7, "Lidama", "7006"));
56 tableView.getItems().add(
57     new Mahasiswa(8, "Aninda", "7007"));
58 tableView.getItems().add(
59     new Mahasiswa(9, "Isanhabang",
60 "7008"));
61 tableView.getItems().add(
62     new Mahasiswa(10, "IrfanStis",
63 "7009"));
64
65 VBox vbox = new VBox(tableView);

```

```

66
67         Scene scene = new Scene(vbox);
68
69         primaryStage.setScene(scene);
70
71         primaryStage.show();
72     }
73
74 }

```

Tabel 55.1 Source Code Soal 1 Main

B. Output Program



NIM	Nama
7000	Audrey
7001	Dandellion
7002	Deyya
7003	Aina
7004	Dawan
7005	Helenah
7006	Lidama
7007	Aninda
7008	Isanhabang
7009	IrfanStis

Gambar 20.1 Screenshot Output Soal 1

C. Pembahasan

a. Mahasiswa

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3 dan 4: *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 7 : *int*, tipe data *integer* atau menerima bilangan bulat.

Line 8 : *String*, tipe data teks yang berisikan karakter lebih.

Line 9 : *String*, tipe data teks yang berisikan karakter lebih.

Line 18, 22, 26,30, 35, dan 38 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 42: {}, penanda awal dan akhir *block code*..

b. Main

Line 1 : *Package*, untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3 – 10 : *Import*, suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 12 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 12 : *Extends* , kata kunci yang ditulis dengan kelas anak selama deklarasi kelas diikuti dengan nama kelas induk.

Line 12 dan 72 : {}, penanda awal dan akhir *block code*.

LINK GITHUB

<https://github.com/RylenRawr/PEMROGRAMAN-II>