

LEMBAR KERJA MAHASISWA  
PENROGRAMAN BERORIENTASI OBJEK



NAMA : ADMIRA SABTATIL YADAINI

NIM : 22241073

KL : PTI-B

PROGRAM STUDI S1 PENDIDIKAN TEKNOLOGI INFORMASI  
FAKULTAS SAINS TEKNIK DAN TERAPAN  
UNIVERSITAS PENDIDIKAN MANDALIKA

2023 / 2024

Main.java X

src > com > tutorial > Main.java > Segitiga > display()

```
1  package com.tutorial;
2
3  // Kelas Induk atau Superclass
4  class BangunDatar {
5      public float hitungLuas() {
6          return 0.0f;
7      }
8
9      public float hitungKeliling() {
10         return 0.0f;
11     }
12 }
13
14 // Subclass Persegi
15 class Persegi extends BangunDatar {
16     private float sisi;
17
18     public Persegi(float sisi) {
19         this.sisi = sisi;
20     }
21
22     @Override
23     public float hitungLuas() {
24         return sisi * sisi;
25     }
26
27     @Override
28     public float hitungKeliling() {
29         return 4 * sisi;
30     }
31
32     public void display() {
33         System.out.println("Persegi dengan sisi: " + sisi);
34     }
35 }
36
37 // Subclass Lingkaran
38 class Lingkaran extends BangunDatar {
39     private float jariJari;
40     private final float PI = 3.14f;
41
42     public Lingkaran(float jariJari) {
43         this.jariJari = jariJari;
44     }
45
46     @Override
47     public float hitungLuas() {
48         return PI * jariJari * jariJari;
```



```

49     }
50
51     @Override
52     public float hitungKeliling() {
53         return 2 * PI * jariJari;
54     }
55
56     public void display() {
57         System.out.println("Jari-jari lingkaran: " + jariJari);
58     }
59 }
60
61 // Subclass Persegi Panjang
62 class PersegiPanjang extends BangunDatar {
63     private float panjang;
64     private float lebar;
65
66     public PersegiPanjang(float panjang, float lebar) {
67         this.panjang = panjang;
68         this.lebar = lebar;
69     }
70
71     @Override
72     public float hitungLuas() {

```

```

73         return panjang * lebar;
74     }
75
76     @Override
77     public float hitungKeliling() {
78         return 2 * (panjang + lebar);
79     }
80
81     public void display() {
82         System.out.println("Panjang persegi panjang: " + panjang + " dan lebar: " + lebar);
83     }
84 }
85
86 // Subclass Segitiga
87 class Segitiga extends BangunDatar {
88     private float alas;
89     private float tinggi;
90     private float sisi1;
91     private float sisi2;
92     private float sisi3;
93
94     public Segitiga(float alas, float tinggi, float sisi1, float sisi2, float sisi3) {
95         this.alas = alas;
96         this.tinggi = tinggi;

```

```

97         this.sisi1 = sisi1;
98         this.sisi2 = sisi2;
99         this.sisi3 = sisi3;
100     }
101
102     @Override
103     public float hitungLuas() {
104         return 0.5f * alas * tinggi;
105     }
106
107     @Override
108     public float hitungKeliling() {
109         return sisi1 + sisi2 + sisi3;
110     }
111
112     public void display() {
113         System.out.println("Alas segi tiga: " + alas + ", tinggi segi tiga: " + tinggi +
114             ", dan panjang sisi-sisi: " + sisi1 + ", " + sisi2 + ", " + sisi3);
115     }
116 }
117
118 public class Main {
119     Run | Debug
120     public static void main(String[] args) {

```

```

118     public class Main {
119         Run | Debug
120         public static void main(String[] args) {
121             // Menggunakan kelas Persegi
122             Persegi persegi = new Persegi(sisi: 5.0f);
123             persegi.display();
124             System.out.println("Luas Persegi: " + persegi.hitungLuas());
125             System.out.println("Keliling Persegi: " + persegi.hitungKeliling());
126             System.out.println();
127
128             // Menggunakan kelas Lingkaran
129             Lingkaran lingkaran = new Lingkaran(jariJari: 25.0f);
130             lingkaran.display();
131             System.out.println("Luas Lingkaran: " + lingkaran.hitungLuas());
132             System.out.println("Keliling Lingkaran: " + lingkaran.hitungKeliling());
133             System.out.println();
134
135             // Menggunakan kelas Persegi Panjang
136             PersegiPanjang persegiPanjang = new PersegiPanjang(panjang: 12.0f, lebar: 14.0f);
137             persegiPanjang.display();
138             System.out.println("Luas Persegi Panjang: " + persegiPanjang.hitungLuas());
139             System.out.println("Keliling Persegi Panjang: " + persegiPanjang.hitungKeliling());
140             System.out.println();

```



```

141 // Menggunakan kelas Segitiga
142 Segitiga segitiga = new Segitiga(alas: 4.0f, tinggi: 5.0f, sisi1: 6.0f, sisi2: 5.0f, sisi3: 4.0f);
143 segitiga.display();
144 System.out.println("Luas Segitiga: " + segitiga.hitungluas());
145 System.out.println("Keliling Segitiga: " + segitiga.hitungkeliling());
146 }
147 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Jari-jari lingkaran: 25.0  
 Luas lingkaran: 1962.5  
 Keliling lingkaran: 157.0

Panjang persegi panjang: 12.0 dan lebar: 14.0  
 Luas Persegi Panjang: 168.0  
 Keliling Persegi Panjang: 52.0

Alas segi tiga: 4.0, tinggi segi tiga: 5.0, dan panjang sisi-sisi: 6.0, 5.0, 4.0  
 Luas Segitiga: 10.0  
 Keliling Segitiga: 15.0

PS D:\Admira.22241073\_PBO\K04 4-Bangun Datar>

① Opening Java

## PENJELASANNYA

- \* Baris ke-1 : Merupakan nama package
- \* Baris ke-4 : Merupakan class induk atau superclass
- \* Baris ke-5 & 6 : Merupakan method tanpa parameter dengan return
- \* Baris ke-9 & 10 : Merupakan method tanpa parameter dengan return
- \* Baris ke-15 : Merupakan subclass dari Bangun Datar
- \* Baris ke-16 : Merupakan atribut dari sub class Bangun Datar
- \* Baris ke-18 : Merupakan konstruktor
- \* Baris ke-19 : Merupakan this (menginisialisasi nilai diri)
- \* Baris ke-20 : Merupakan override yg digunakan untuk mengisi (override) implementasi metode dari superclass.
- \* Baris ke-23 & 24 : Merupakan method tanpa parameter dengan return
- \* Baris ke-27 : Merupakan override yg digunakan untuk mengisi (override) implementasi metode dari superclass.
- \* Baris ke-28 : Merupakan method tanpa parameter dengan return
- \* Baris ke-32 & 33 : Merupakan Method display() yang digunakan untuk menampilkan informasi tentang persegi yg dibentuk



- \* Baris ke-38 : Merupakan subclass dari Bangun Datar
- \* Baris ke-39 & 40 : Merupakan atribut dari sub class Bangun Datar.
- \* Baris ke-42 : Merupakan Konstruktork
- \* Baris ke-43 : Merupakan this untuk menginisialisasi nilai PI
- \* Baris ke-46 : Merupakan override yg digunakan untuk mengisi (override) implementasi metode dari superclass.
- \* Baris ke-47 : Merupakan method tanpa parameter dengan return
- \* Baris ke-51 : Merupakan override yg digunakan untuk mengisi (override) implementasi metode dari superclass.
- \* Baris ke-52 & 53 : Merupakan Method tanpa parameter dengan return.
- \* Baris ke-56 : Merupakan method 'display()' yg digunakan untuk menampilkan informasi tentang lingkaran yg dibentuk.
- \* Baris ke-62 : Merupakan subclass dari Bangun Datar
- \* Baris ke-63 & 64 : Merupakan atribut dari sub class Bangun Datar.
- \* Baris ke-66 : Merupakan konstruktork
- \* Baris ke-67 & 68 : Merupakan this untuk menginisialisasi nilai panjang & lebar

- \* Baris ke-71 : Merupakan override yg digunakan untuk menerima (override) implementasi metode dari superclass.
- \* Baris ke-72 & 73 : Merupakan method tanpa parameter dengan return
- \* Baris ke-76 : Merupakan override yg digunakan untuk menerima (override) implementasi metode dari superclass.
- \* Baris ke-77 & 78 : Merupakan method tanpa parameter dengan return.
- \* Baris ke-81 & 82 : Merupakan method 'display()' yg digunakan untuk menampilkan informasi tentang persegi panjang
- \* Baris ke-87 : Merupakan subclass dari Bangun Datar
- \* Baris ke-88-92 : Merupakan atribut dari sub class Bangun Datar
- \* Baris ke-94 : Merupakan konstruktor
- \* Baris ke-95-99 : Merupakan this untuk menginisialisasi nilai alas, tinggi, sisi1, sisi2, sisi3.
- \* Baris ke-102 : Merupakan override yg digunakan untuk menerima (override) implementasi metode dari superclass.
- \* Baris ke-103-104 : Merupakan method tanpa parameter dengan return.
- \* Baris ke-107 : Merupakan override yg digunakan untuk menerima (override) implementasi metode dari superclass.

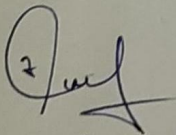


- \* Baris ke-108 & 109 : Merupakan method tanpa parameter dengan ada return.
- \* Baris ke-112 -114 : Merupakan ~~exit~~ 'display()' yg digunakan untuk menampilkan informasi tentang segi tiga yg dibentuk.
- \* Baris ke-118 : Merupakan class Main
- \* Baris ke-119-122 : Merupakan method tanpa parameter dan tanpa return
- \* Baris ke-123-125 : Merupakan output untuk menampilkan informasi ke layar selama eksekusi program.
- \* Baris ke-128-129 : Merupakan method tanpa parameter dan tanpa return.
- \* Baris ke-130-132 : Merupakan output untuk menampilkan informasi ke layar selama eksekusi program.
- \* Baris ke-135-136 : Merupakan method tanpa parameter dan tanpa return.
- \* Baris ke-137 - 139 : Merupakan output untuk menampilkan informasi ke layar.
- \* Baris ke-142-143 : Merupakan method tanpa parameter dan tanpa return.
- \* Baris ke-144 & 145 : Merupakan output untuk menampilkan informasi ke layar.

## LEMBAR PENGESAHAN LAPORAN

Judul Laporan : Lembar Kerja Mahasiswa  
Nama Lengkap : ADMIRA SABTATIL YADAINI  
NIM : 22241073  
Kelas : PTI-B  
Mata Kuliah : Pemrograman Berorientasi Objek

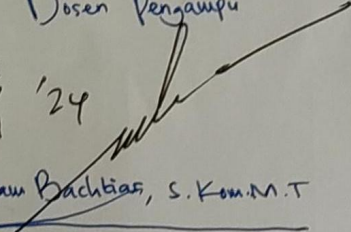
Disusun oleh



Admira Sabtatil Yadaini

Mengetahui  
Dosen Pengampu

Aec  
2/1 '24

  
Adam Bachtiar, S.Kom.M.T