

LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBYEK

MODUL 9

ABSTRACT CLASS



Disusun Oleh :

Nama : Yoga Pramudita

NIM : L200200182

Kelas : D

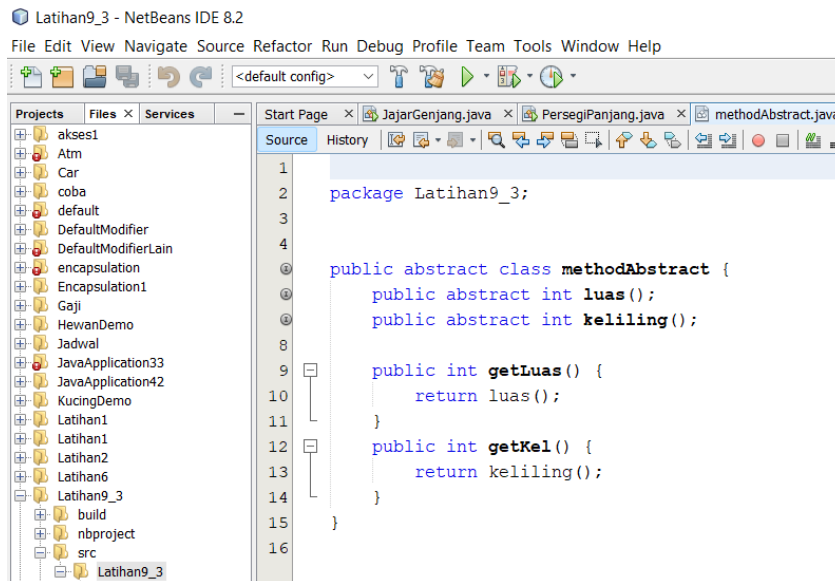
Matkul : Praktikum PBO

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA
TAHUN 2021/2022**

9.3 LATIHAN

Dengan menggunakan class MethodAbstract pada program 5 diatas, buatlah class PersegiPanjang, JajarGenjang, Lingkaran, dan Segitiga! Selanjutnya implementasikan method luas() dan keliling() yang sesuai dengan perhitungan masing – masing.

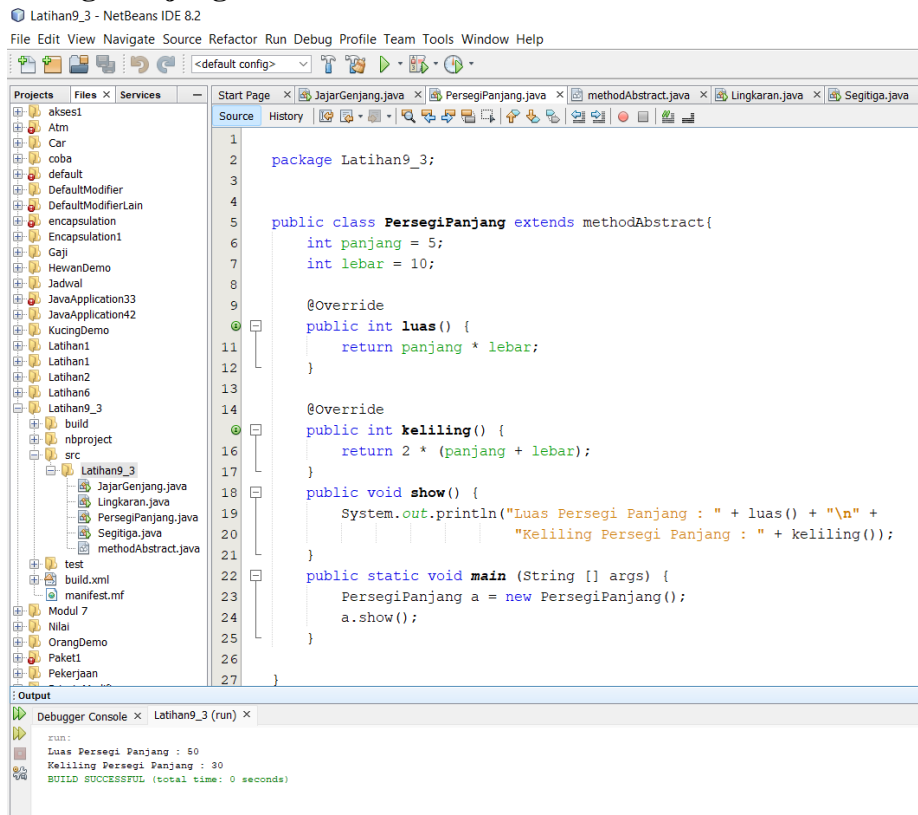
- **MethodAbstract**



The screenshot shows the NetBeans IDE 8.2 interface. The 'Projects' window on the left displays a project named 'Latihan9_3' with a source folder 'src'. The 'Source' window on the right shows the code for 'methodAbstract.java'. The code defines an abstract class 'methodAbstract' with two abstract methods: 'luas()' and 'keliling()'. It also includes two concrete methods: 'getLuas()' which returns the result of 'luas()', and 'getKel()' which returns the result of 'keliling()'. The package is 'Latihan9_3'.

```
1 package Latihan9_3;
2
3
4 public abstract class methodAbstract {
5     public abstract int luas();
6     public abstract int keliling();
7
8
9     public int getLuas() {
10         return luas();
11     }
12     public int getKel() {
13         return keliling();
14     }
15 }
16
```

- **Persegi Panjang**



The screenshot shows the NetBeans IDE 8.2 interface. The 'Projects' window on the left shows the 'Latihan9_3' project with a 'src' folder containing 'JajarGenjang.java', 'Lingkaran.java', 'PersegiPanjang.java', 'Segitiga.java', and 'methodAbstract.java'. The 'Source' window on the right shows the code for 'PersegiPanjang.java'. The code defines a class 'PersegiPanjang' that extends 'methodAbstract'. It has two instance variables: 'panjang' (5) and 'lebar' (10). It overrides the 'luas()' method to return 'panjang * lebar' and the 'keliling()' method to return '2 * (panjang + lebar)'. It also has a 'show()' method that prints the area and perimeter, and a 'main()' method that creates a 'PersegiPanjang' object and calls 'show()'. The package is 'Latihan9_3'.

```
1 package Latihan9_3;
2
3
4 public class PersegiPanjang extends methodAbstract{
5     int panjang = 5;
6     int lebar = 10;
7
8     @Override
9     public int luas() {
10         return panjang * lebar;
11     }
12
13     @Override
14     public int keliling() {
15         return 2 * (panjang + lebar);
16     }
17
18     public void show() {
19         System.out.println("Luas Persegi Panjang : " + luas() + "\n" +
20                             "Keliling Persegi Panjang : " + keliling());
21     }
22
23     public static void main (String [] args) {
24         PersegiPanjang a = new PersegiPanjang();
25         a.show();
26     }
27 }

```

The 'Output' window at the bottom shows the results of running the program:

```
DEBUGGER CONSOLE x Latihan9_3 (run) x
RUN:
Luas Persegi Panjang : 50
Keliling Persegi Panjang : 30
BUILD SUCCESSFUL (total time: 0 seconds)
```

- **Jajar Genjang**

Latihan9_3 - NetBeans IDE 8.2

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>

Projects Files Services
  akse1
  Atm
  Car
  coba
  default
  DefaultModifier
  DefaultModifierLain
  encapsulation
  Encapsulation1
  Gaji
  HewanDemo
  Jadwal
  JavaApplication33
  JavaApplication42
  KucingDemo
  Latihan1
  Latihan1
  Latihan2
  Latihan6
  Latihan9_3
    build
    nbproject
    src
      JajarGenjang.java
      Lingkaran.java
      PersegiPanjang.java
      Segitiga.java
      methodAbstract.java
    test
    build.xml
    manifest.mf
  Modul 7
  Nilai
  OrangDemo
  Paket1
  Pekerjaan

Start Page JajarGenjang.java PersegiPanjang.java methodAbstract.java Lingkaran.java Segitiga.java
Source History
1 package Latihan9_3;
2
3 public class JajarGenjang extends methodAbstract {
4     int sisialas = 5;
5     int sisimiring = 4;
6     int tinggi = 9;
7
8
9     @Override
10    public int luas() {
11        return sisialas * tinggi;
12    }
13
14    @Override
15    public int keliling() {
16        return 2 * (sisialas + sisimiring);
17    }
18
19    public void show() {
20        System.out.println("Luas Jajar Genjang : " + luas() + "\n" +
21                            "Keliling Jajaar Genjang : " + keliling());
22    }
23
24    public static void main(String[] args) {
25        JajarGenjang b = new JajarGenjang();
26        b.show();
27    }
28
Output
Debugger Console x Latihan9_3 (run) x
run:
Luas Jajar Genjang : 45
Keliling Jajaar Genjang : 18
BUILD SUCCESSFUL (total time: 0 seconds)
```

- **Lingkaran**

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>

Projects Files Services
  akse1
  Atm
  Car
  coba
  default
  DefaultModifier
  DefaultModifierLain
  encapsulation
  Encapsulation1
  Gaji
  HewanDemo
  Jadwal
  JavaApplication33
  JavaApplication42
  KucingDemo
  Latihan1
  Latihan1
  Latihan2
  Latihan6
  Latihan9_3
    build
    nbproject
    src
      JajarGenjang.java
      Lingkaran.java
      PersegiPanjang.java
      Segitiga.java
      methodAbstract.java
    test
    build.xml
    manifest.mf
  Modul 7
  Nilai
  OrangDemo
  Paket1
  Pekerjaan

Start Page JajarGenjang.java PersegiPanjang.java methodAbstract.java Lingkaran.java Segitiga.java
Source History
1 package Latihan9_3;
2
3 public class Lingkaran extends methodAbstract {
4     double pi = 3.14;
5     int diameter = 10;
6     int jarijari = 5;
7
8
9     @Override
10    public int luas() {
11        return (int) (pi * jarijari*jarijari);
12    }
13
14    @Override
15    public int keliling() {
16        return (int) (pi*diameter);
17    }
18
19    public void show() {
20        System.out.println("Luas Lingkaran : " + luas() + "\n" +
21                            "Keliling Lingkaran : " + keliling());
22    }
23
24    public static void main (String[] args) {
25        Lingkaran c = new Lingkaran();
26        c.show();
27    }
28
Output
Debugger Console x Latihan9_3 (run) x
run:
Luas Lingkaran : 78
Keliling Lingkaran : 31
BUILD SUCCESSFUL (total time: 0 seconds)
```

- Segitiga

Latihan9_3 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

The screenshot displays the NetBeans IDE interface. On the left, the 'Projects' pane shows a tree structure with 'Latihan9_3' expanded, revealing sub-projects like 'src' and 'test'. The 'src' folder contains 'Latihan9_3', which includes 'JajarGenjang.java', 'Lingkaran.java', 'PersegiPanjang.java', 'Segitiga.java', and 'methodAbstract.java'. The main editor window shows the source code of 'Segitiga.java'. The code defines a package 'Latihan9_3' and a class 'Segitiga' that extends 'methodAbstract'. It includes attributes 'alas', 'tinggi', and 'sisimiring', and implements methods 'luas()', 'keliling()', 'show()', and a 'main' method. The 'show()' method prints the calculated area and perimeter. The bottom pane shows the 'Output' window with the results of running the program: 'Luas Segitiga : 56' and 'Keliling Segitiga : 36', followed by a 'BUILD SUCCESSFUL' message.

```
1 package Latihan9_3;
2
3 public class Segitiga extends methodAbstract{
4     int alas = 8;
5     int tinggi = 14;
6     int sisimiring = 14;
7
8     @Override
9     public int luas() {
10         return (int) (0.5 * alas * tinggi);
11     }
12
13     @Override
14     public int keliling() {
15         return alas + sisimiring * 2;
16     }
17
18     public void show() {
19         System.out.println("Luas Segitiga : " + luas() + "\n" +
20                             "Keliling Segitiga : " + keliling());
21     }
22
23     public static void main (String[] args) {
24         Segitiga d = new Segitiga();
25         d.show();
26     }
27 }
```

Output

run:

Luas Segitiga : 56
Keliling Segitiga : 36
BUILD SUCCESSFUL (total time: 0 seconds)

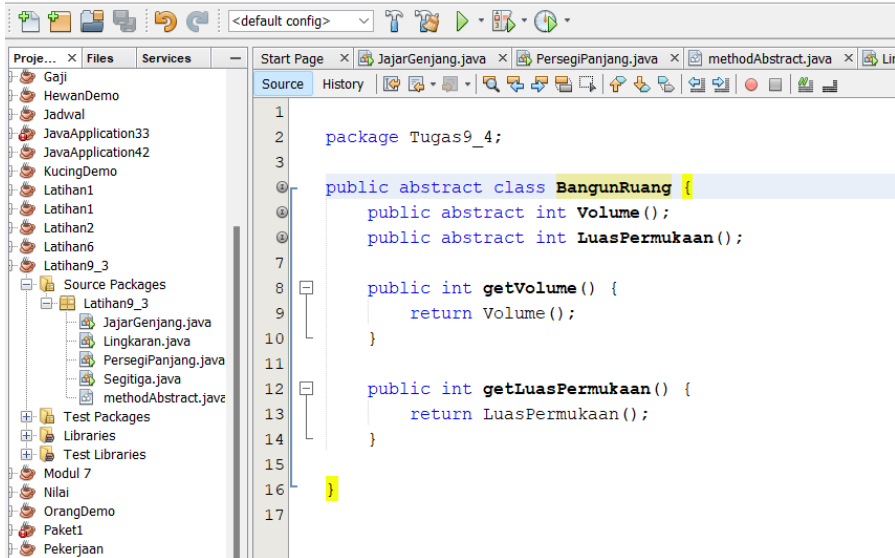
9.4 TUGAS

Buatlah class abstract untuk bangun ruang, dengan ketentuan memiliki method abstract untuk menghitung volume dan luas selimut/luas permukaan. Selanjutnya buatlah class Balok, Kubus, Bola, Kerucut, dan PrismaSegitiga untuk mengimplementasikan method abstract tersebut!

- **Bangun Ruang**

Tugas 9.4 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

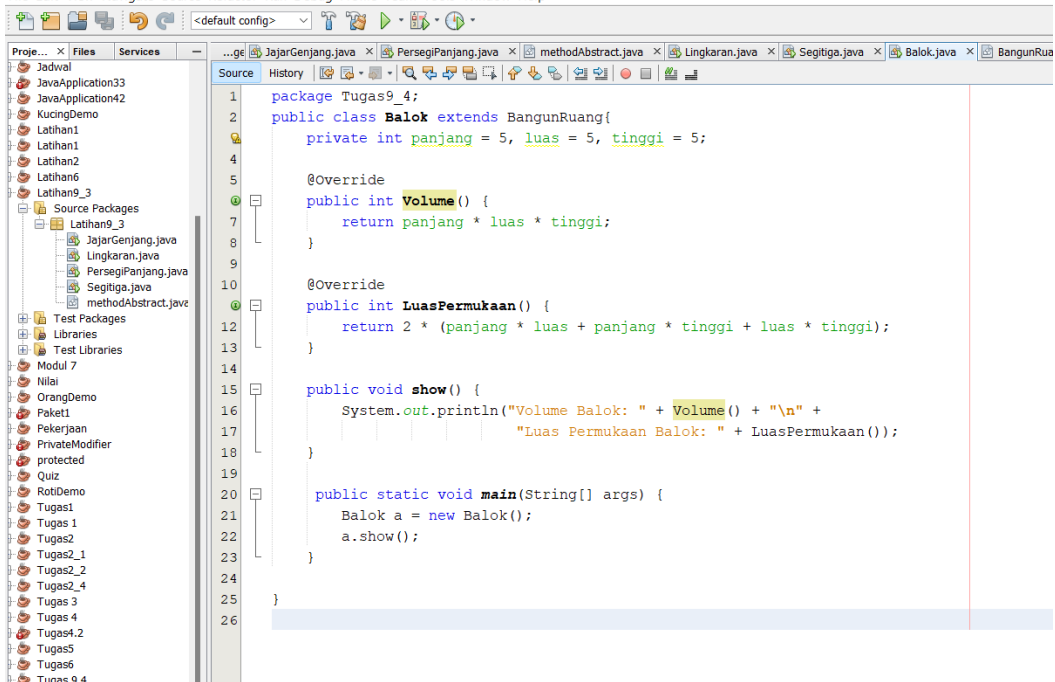


```
1
2 package Tugas9_4;
3
4 public abstract class BangunRuang {
5     public abstract int Volume();
6     public abstract int LuasPermukaan();
7
8     public int getVolume() {
9         return Volume();
10    }
11
12    public int getLuasPermukaan() {
13        return LuasPermukaan();
14    }
15
16 }
17
```

- **Balok**

Tugas 9.4 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help



```
1 package Tugas9_4;
2 public class Balok extends BangunRuang{
3     private int panjang = 5, luas = 5, tinggi = 5;
4
5     @Override
6     public int Volume() {
7         return panjang * luas * tinggi;
8     }
9
10    @Override
11    public int LuasPermukaan() {
12        return 2 * (panjang * luas + panjang * tinggi + luas * tinggi);
13    }
14
15    public void show() {
16        System.out.println("Volume Balok: " + Volume() + "\n" +
17                            "Luas Permukaan Balok: " + LuasPermukaan());
18    }
19
20    public static void main(String[] args) {
21        Balok a = new Balok();
22        a.show();
23    }
24
25 }
26
```

- Kubus

Tugas 9.4 - NetBeans IDE 8.2

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
JajarGenjang.java x PersegiPanjang.java x methodAbstract.java x Lingkaran.java x Segitiga.java x Balok.java
Source History
1 package Tugas9_4;
2 public class Kubus extends BangunRuang{
3
4     public int sisi;
5
6     @Override
7     public int Volume() {
8         return sisi * sisi * sisi;
9     }
10
11    @Override
12    public int LuasPermukaan() {
13        return 6 * sisi * sisi;
14    }
15
16    public void show() {
17        System.out.println("Volume Kubus: " + Volume() + "\n" +
18                            "Luas Permukaan Kubus: " + LuasPermukaan());
19    }
20
21    public static void main(String[] args) {
22        Kubus b = new Kubus();
23        b.show();
24    }
25 }
26
```

- Bola

Tugas 9.4 - NetBeans IDE 8.2

```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
JajarGenjang.java x PersegiPanjang.java x methodAbstract.java x Lingkaran.java x Segitiga.java x Balok.java
Source History
1 package Tugas9_4;
2
3 public class Bola extends BangunRuang{
4     final double pi = 3.14;
5     int jarijari = 16;
6
7     @Override
8     public int Volume() {
9         return (int) (4/3 * pi * jarijari * jarijari * jarijari);
10    }
11
12    @Override
13    public int LuasPermukaan() {
14        return (int) (4 * jarijari * jarijari);
15    }
16
17    public void show() {
18        System.out.println("Volume Bola: " + Volume() + "\n" +
19                            "Luas Permukaan Bola: " + LuasPermukaan());
20    }
21
22    public static void main(String[] args) {
23        Bola c = new Bola();
24        c.show();
25    }
26 }
27
```

- Kerucut

Tugas 9.4 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

```
<default config>
package Tugas9_4;
public class Kerucut extends BangunRuang{
    double pi = 3.14;
    int jarijari = 8;
    int tinggi = 14;
    int s = 22;

    @Override
    public int Volume() {
        return (int) (0.33 * pi * jarijari * jarijari * tinggi);
    }

    @Override
    public int LuasPermukaan() {
        int LuasAlas = (int) pi * jarijari * jarijari;
        int LuasSelimut = (int) pi * jarijari * s;
        return LuasAlas + LuasSelimut;
    }

    public void show() {
        System.out.println("Volume Kerucut: " + Volume() + "\n" +
            "Luas Permukaan Kerucut: " + LuasPermukaan());
    }

    public static void main(String[] args) {
        Kerucut d = new Kerucut();
        d.show();
    }
}
```

- Prisma Segitiga

Tugas 9.4 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

```
<default config>
package Tugas9_4;
public class PrismaSegitiga extends BangunRuang{
    int alas = 3;
    int tinggisegitiga = 4;
    int tinggiprisma = 7;

    @Override
    public int Volume() {
        return (int) ((alas * tinggisegitiga * 0.5) * tinggiprisma);
    }

    @Override
    public int LuasPermukaan() {
        int luasalas = (int) (0.5) * alas * tinggisegitiga;
        int kelilingalas = (int) Math.sqrt(Math.pow(alas, 2) + Math.pow(tinggisegitiga, 2));
        int kelilingsegitiga = (int) alas + tinggisegitiga + kelilingalas;
        return (2 * luasalas) + (kelilingsegitiga * tinggiprisma);
    }

    public void show() {
        System.out.println("Volume Prisma Segitiga: " + Volume() + "\n" +
            "Luas Permukaan Prisma Segitiga: " + LuasPermukaan());
    }

    public static void main(String[] args) {
        PrismaSegitiga e = new PrismaSegitiga();
        e.show();
    }
}
```