

**PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK**

**Bab 9**

**ABSTRACT CLASS**



**DISUSUN OLEH:**

**ILHAM RIAN NOVANTO**

**L200200247**

**INFORMATIKA**

**FAKULTAS KOMUNIKASI DAN INFORMATIKA**

**UNIVERSITAS MUHAMMADIYAH SURAKARTA**

**2021**

Kegiatan praktikum modul 9 yaitu:

### 9.3. LATIHAN

Dengan menggunakan class MethodAbstrak pada Program 5 di atas, buatlah class PersegiPanjang, JajarGenjang, Lingkaran, dan Segitiga! Selanjutnya implementasikan method luas() dan keliling() yang sesuai dengan perhitungan masing-masing.

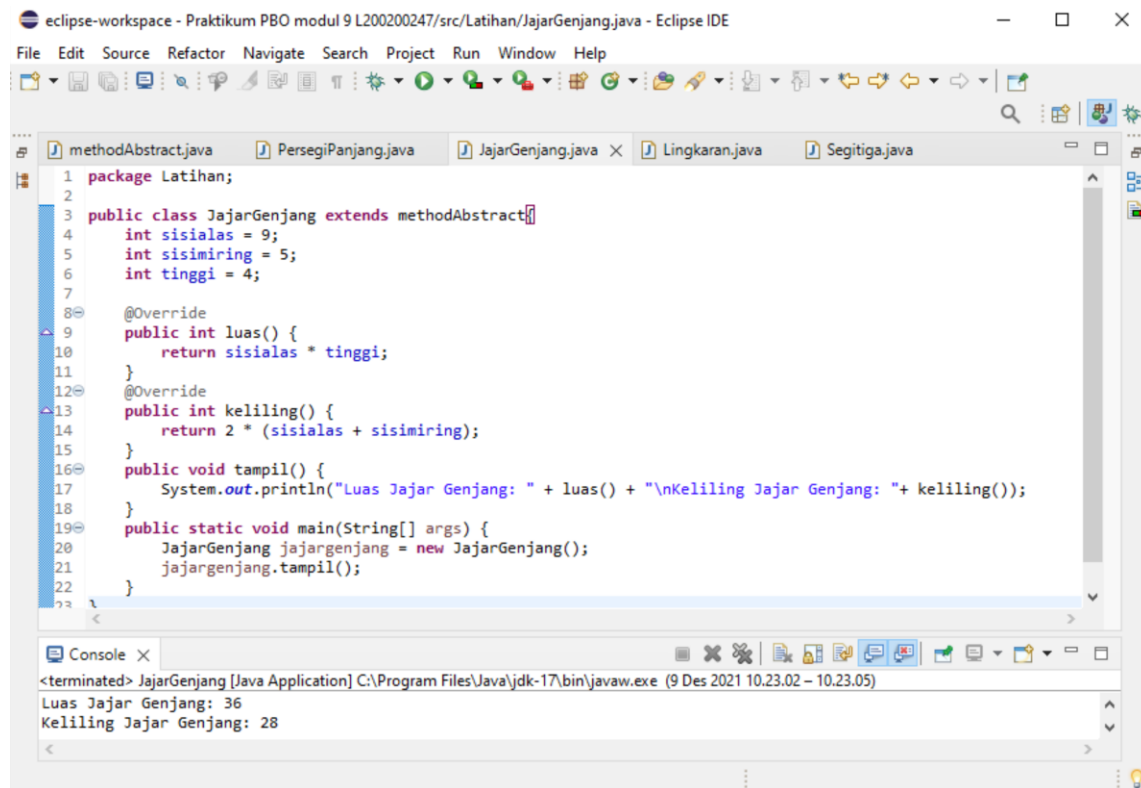
#### Class MethodAbstract

```
eclipse-workspace - Praktikum PBO modul 9 L200200247/src/Latihan/methodAbstract.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
methodAbstract.java x PersegiPanjang.j... JajarGenjang.java Lingkaran.java Segitiga.java
1 package Latihan;
2
3 public abstract class methodAbstract {
4     public abstract int luas();
5     public abstract int keliling();
6
7     public int getLuas() {
8         return luas();
9     }
10    public int getKell() {
11        return keliling();
12    }
13
14 }
15
```

#### Class Persegi Panjang

```
eclipse-workspace - Praktikum PBO modul 9 L200200247/src/Latihan/PersegiPanjang.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
methodAbstract.java x PersegiPanjang.j... JajarGenjang.java Lingkaran.java Segitiga.java
1 package Latihan;
2
3 public class PersegiPanjang extends methodAbstract{
4     int panjang = 12;
5     int lebar = 5;
6
7     @Override
8     public int luas() {
9         return panjang * lebar;
10    }
11    @Override
12    public int keliling() {
13        return 2 * (panjang + lebar);
14    }
15    public void tampil() {
16        System.out.println("Luas Persegi Panjang: " + luas() + "\nKeliling Persegi Panjang: " + ke
17    }
18    public static void main(String [] args) {
19        PersegiPanjang persegiPanjang = new PersegiPanjang();
20        persegiPanjang.tampil();
21    }
22 }
23
Console x
<terminated> PersegiPanjang [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 10.21.41 - 10.21.46)
Luas Persegi Panjang: 60
Keliling Persegi Panjang: 34
```

## Class JajarGenjang

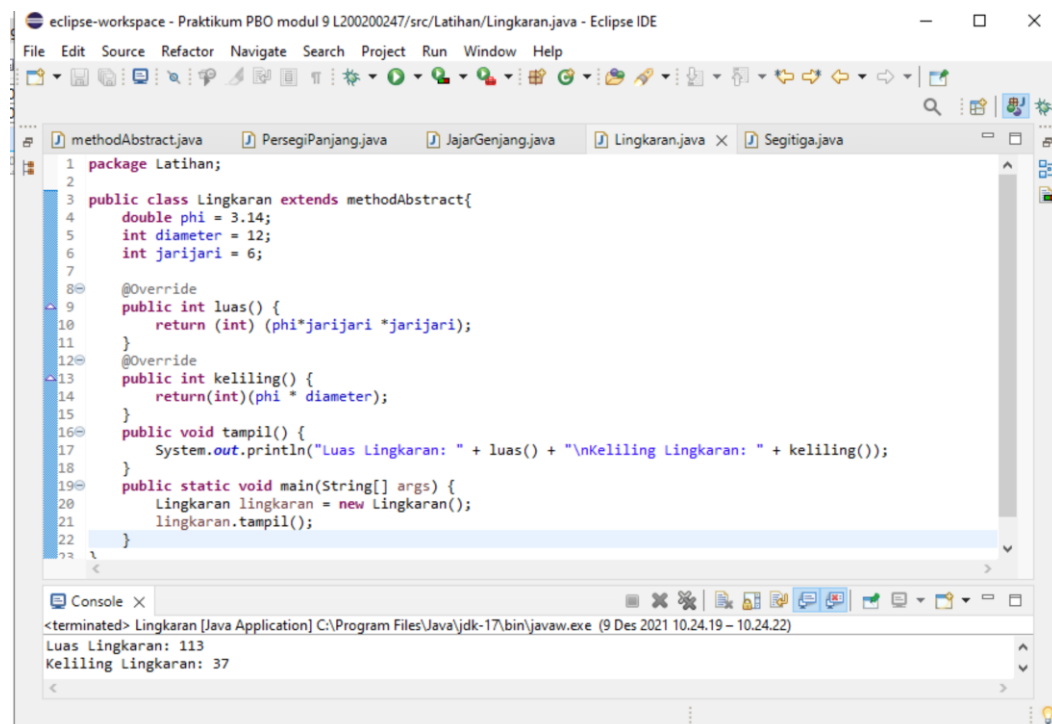


```
1 package Latihan;
2
3 public class JajarGenjang extends methodAbstract{
4     int sisialas = 9;
5     int sisimiring = 5;
6     int tinggi = 4;
7
8     @Override
9     public int luas() {
10         return sisialas * tinggi;
11     }
12     @Override
13     public int keliling() {
14         return 2 * (sisialas + sisimiring);
15     }
16     public void tampil() {
17         System.out.println("Luas Jajar Genjang: " + luas() + "\nKeliling Jajar Genjang: " + keliling());
18     }
19     public static void main(String[] args) {
20         JajarGenjang jajargenjang = new JajarGenjang();
21         jajargenjang.tampil();
22     }
23 }
```

Console Output:

```
<terminated> JajarGenjang [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 10.23.02 - 10.23.05)
Luas Jajar Genjang: 36
Keliling Jajar Genjang: 28
```

## Class Lingkaran

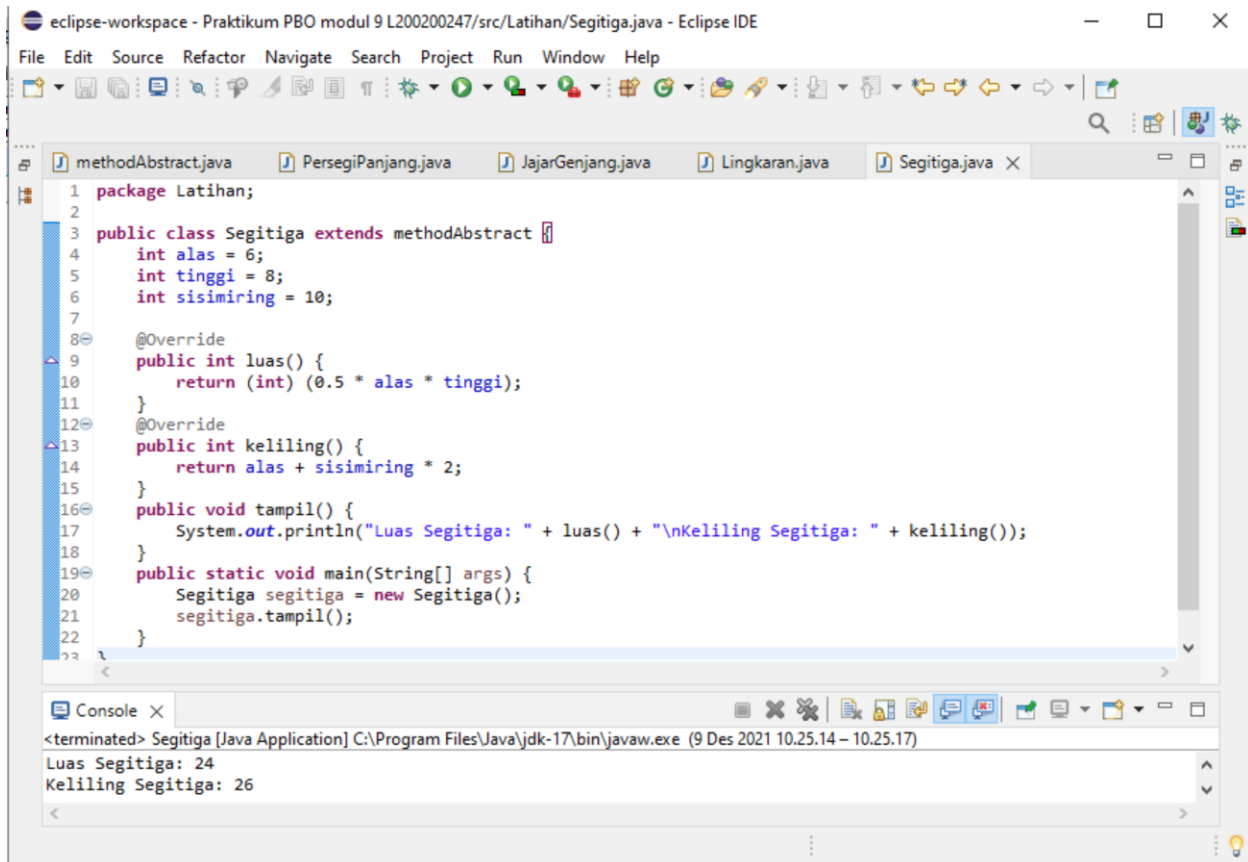


```
1 package Latihan;
2
3 public class Lingkaran extends methodAbstract{
4     double phi = 3.14;
5     int diameter = 12;
6     int jarijari = 6;
7
8     @Override
9     public int luas() {
10         return (int) (phi*jarijari *jarijari);
11     }
12     @Override
13     public int keliling() {
14         return(int)(phi * diameter);
15     }
16     public void tampil() {
17         System.out.println("Luas Lingkaran: " + luas() + "\nKeliling Lingkaran: " + keliling());
18     }
19     public static void main(String[] args) {
20         Lingkaran lingkaran = new Lingkaran();
21         lingkaran.tampil();
22     }
23 }
```

Console Output:

```
<terminated> Lingkaran [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 10.24.19 - 10.24.22)
Luas Lingkaran: 113
Keliling Lingkaran: 37
```

## Class Segitiga



The screenshot shows the Eclipse IDE interface. The top menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations and development tools. The package explorer on the left shows a project named 'Latihan' with several Java files: methodAbstract.java, PersegiPanjang.java, JajarGenjang.java, Lingkaran.java, and Segitiga.java. The editor window displays the code for Segitiga.java, which is a public class extending methodAbstract. The code defines three attributes: alas (6), tinggi (8), and sisimiring (10). It includes two overridden methods: luas() which calculates the area using the formula  $0.5 \times \text{alas} \times \text{tinggi}$ , and keliling() which calculates the perimeter as  $\text{alas} + \text{sisimiring} \times 2$ . A tampil() method prints the results of these calculations. A static main method creates an instance of Segitiga and calls the tampil() method. The console at the bottom shows the output of the program: 'Luas Segitiga: 24' and 'Keliling Segitiga: 26'.

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

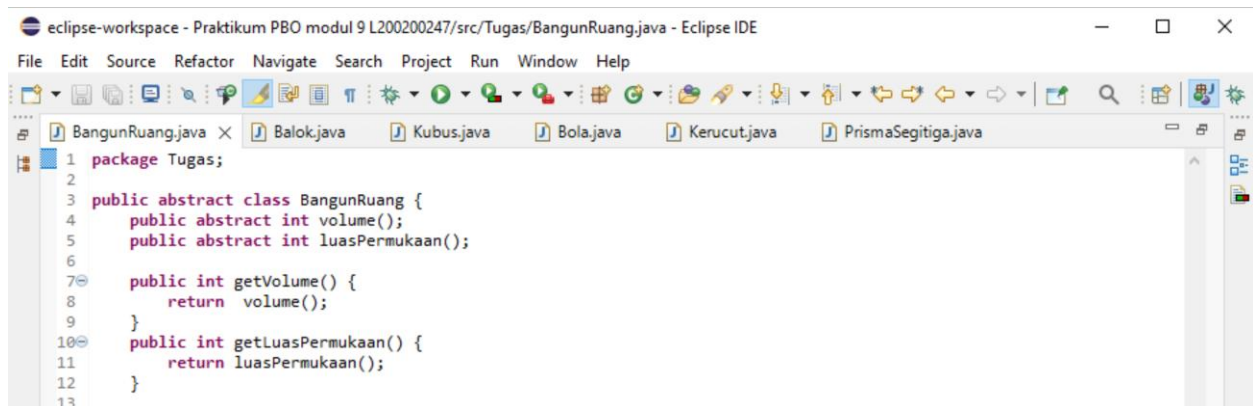
Console Output:

```
<terminated> Segitiga [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 10.25.14 - 10.25.17)
Luas Segitiga: 24
Keliling Segitiga: 26
```

## 9.4.TUGAS

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

### Class Abstract BangunRuang

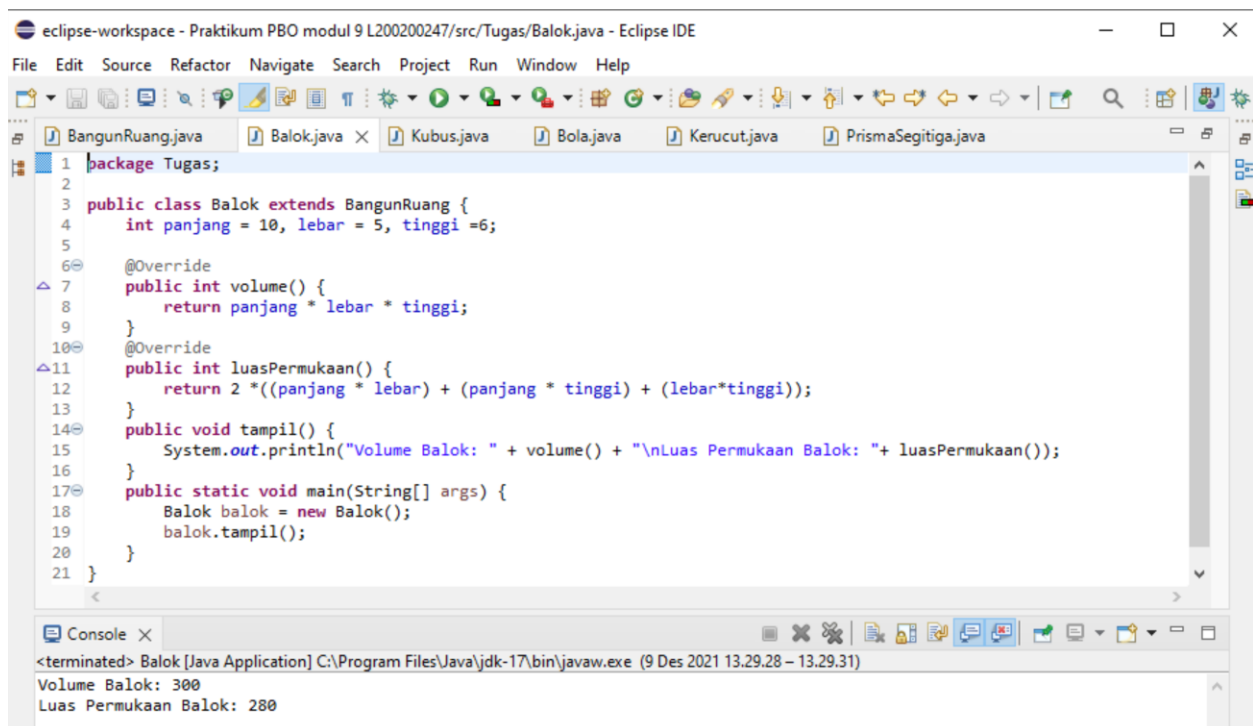


```
eclipse-workspace - Praktikum PBO modul 9 L200200247/src/Tugas/BangunRuang.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

BangunRuang.java X Balok.java Kubus.java Bola.java Kerucut.java PrismaSegitiga.java

1 package Tugas;
2
3 public abstract class BangunRuang {
4     public abstract int volume();
5     public abstract int luasPermukaan();
6
7     public int getVolume() {
8         return volume();
9     }
10    public int getLuasPermukaan() {
11        return luasPermukaan();
12    }
13 }
```

### Class Balok



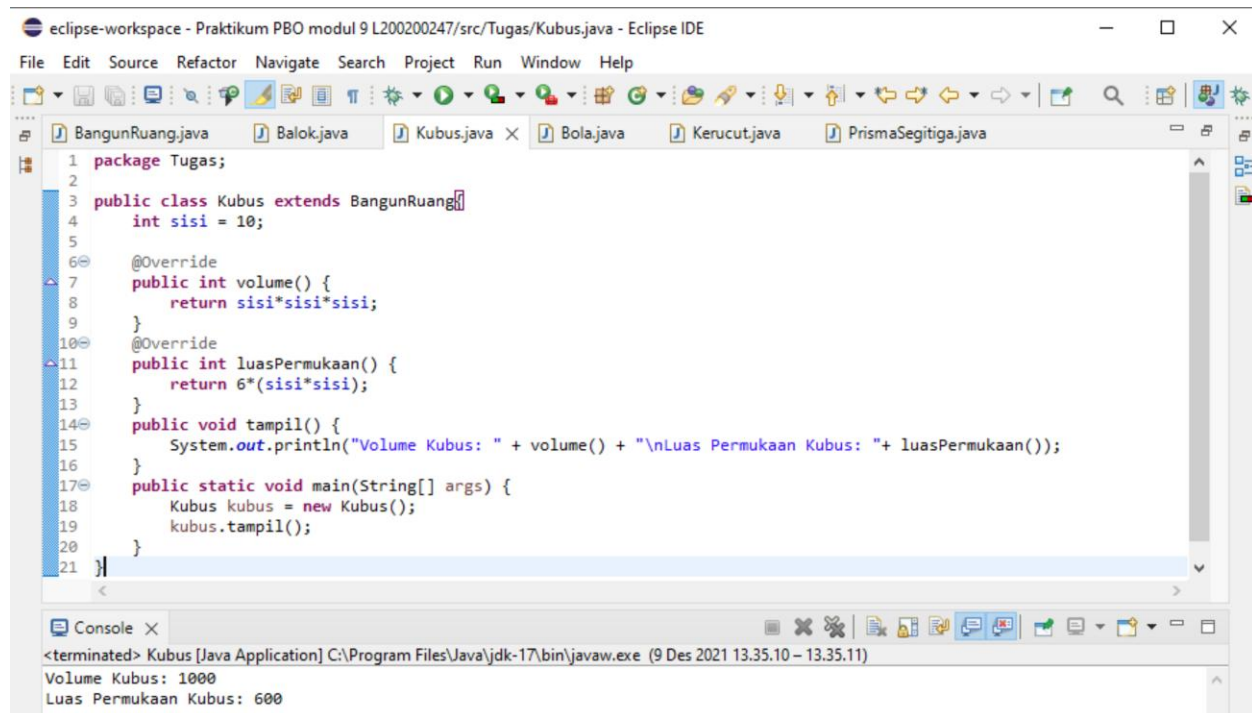
```
eclipse-workspace - Praktikum PBO modul 9 L200200247/src/Tugas/Balok.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

BangunRuang.java Balok.java X Kubus.java Bola.java Kerucut.java PrismaSegitiga.java

1 package Tugas;
2
3 public class Balok extends BangunRuang {
4     int panjang = 10, lebar = 5, tinggi = 6;
5
6     @Override
7     public int volume() {
8         return panjang * lebar * tinggi;
9     }
10    @Override
11    public int luasPermukaan() {
12        return 2 * ((panjang * lebar) + (panjang * tinggi) + (lebar * tinggi));
13    }
14    public void tampil() {
15        System.out.println("Volume Balok: " + volume() + "\nLuas Permukaan Balok: " + luasPermukaan());
16    }
17    public static void main(String[] args) {
18        Balok balok = new Balok();
19        balok.tampil();
20    }
21 }
```

```
Console X
<terminated> Balok [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 13.29.28 - 13.29.31)
Volume Balok: 300
Luas Permukaan Balok: 280
```

## Class Kubus



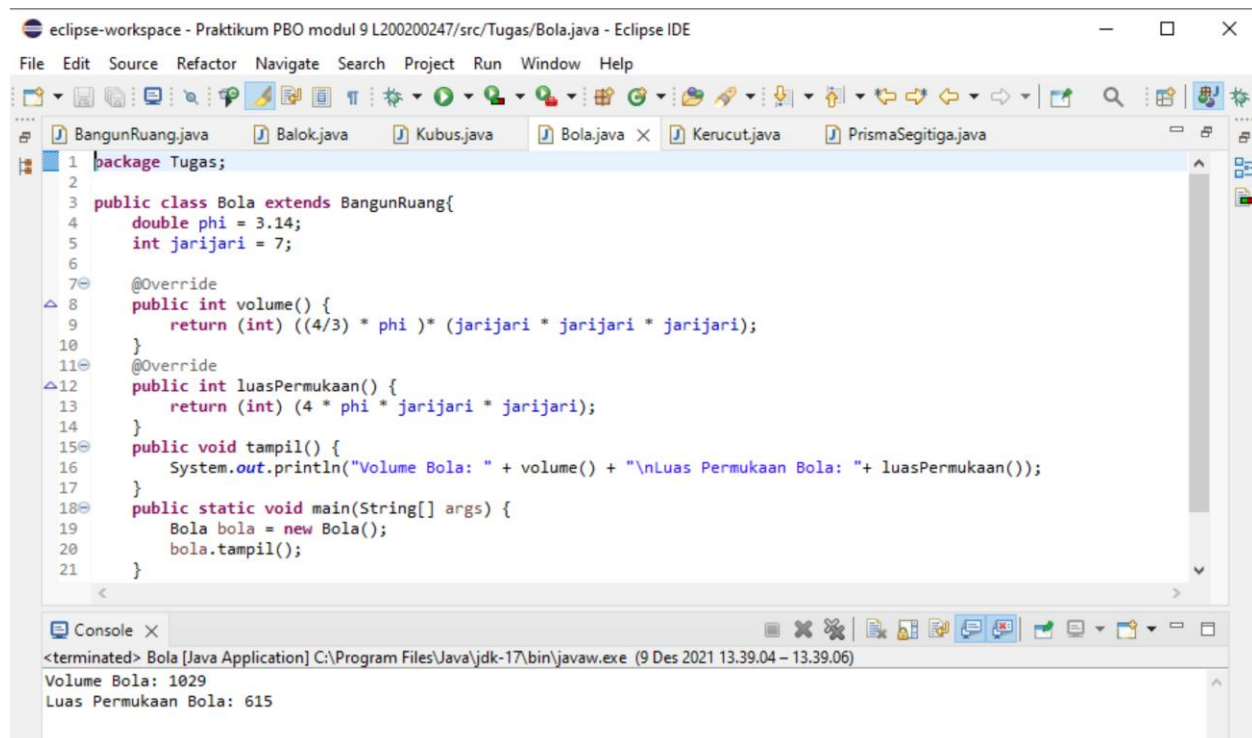
The screenshot shows the Eclipse IDE with the file `Kubus.java` open. The code defines a `Kubus` class that extends `BangunRuang`. It includes methods for calculating volume, surface area, and a display method. The console output shows the results of running the program.

```
1 package Tugas;
2
3 public class Kubus extends BangunRuang{
4     int sisi = 10;
5
6     @Override
7     public int volume() {
8         return sisi*sisi*sisi;
9     }
10    @Override
11    public int luasPermukaan() {
12        return 6*(sisi*sisi);
13    }
14    public void tampil() {
15        System.out.println("Volume Kubus: " + volume() + "\nLuas Permukaan Kubus: " + luasPermukaan());
16    }
17    public static void main(String[] args) {
18        Kubus kubus = new Kubus();
19        kubus.tampil();
20    }
21 }
```

Console Output:

```
<terminated> Kubus [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 13.35.10 - 13.35.11)
Volume Kubus: 1000
Luas Permukaan Kubus: 600
```

## Class Bola



The screenshot shows the Eclipse IDE with the file `Bola.java` open. The code defines a `Bola` class that extends `BangunRuang`. It includes methods for calculating volume, surface area, and a display method. The console output shows the results of running the program.

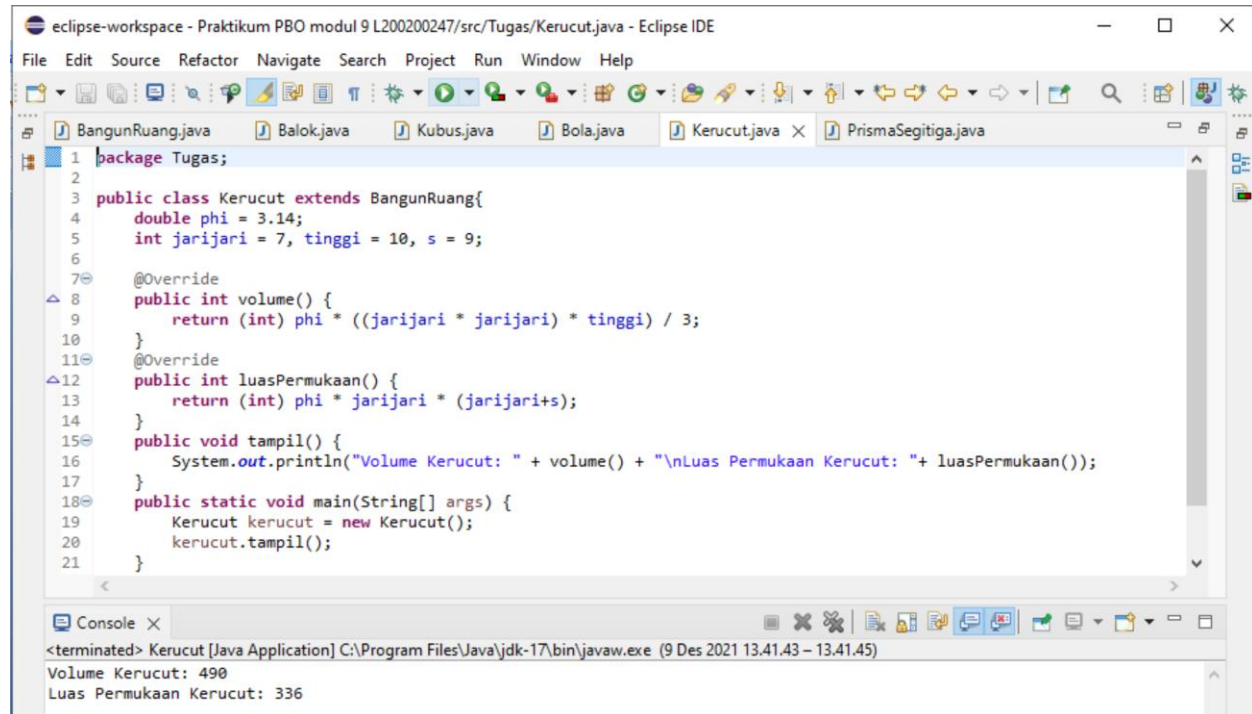
```
1 package Tugas;
2
3 public class Bola extends BangunRuang{
4     double phi = 3.14;
5     int jarijari = 7;
6
7     @Override
8     public int volume() {
9         return (int) ((4/3) * phi ) * (jarijari * jarijari * jarijari);
10    }
11    @Override
12    public int luasPermukaan() {
13        return (int) (4 * phi * jarijari * jarijari);
14    }
15    public void tampil() {
16        System.out.println("Volume Bola: " + volume() + "\nLuas Permukaan Bola: " + luasPermukaan());
17    }
18    public static void main(String[] args) {
19        Bola bola = new Bola();
20        bola.tampil();
21    }
22 }
```

Console Output:

```
<terminated> Bola [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 13.39.04 - 13.39.06)
Volume Bola: 1029
Luas Permukaan Bola: 615
```



## Class Kerucut



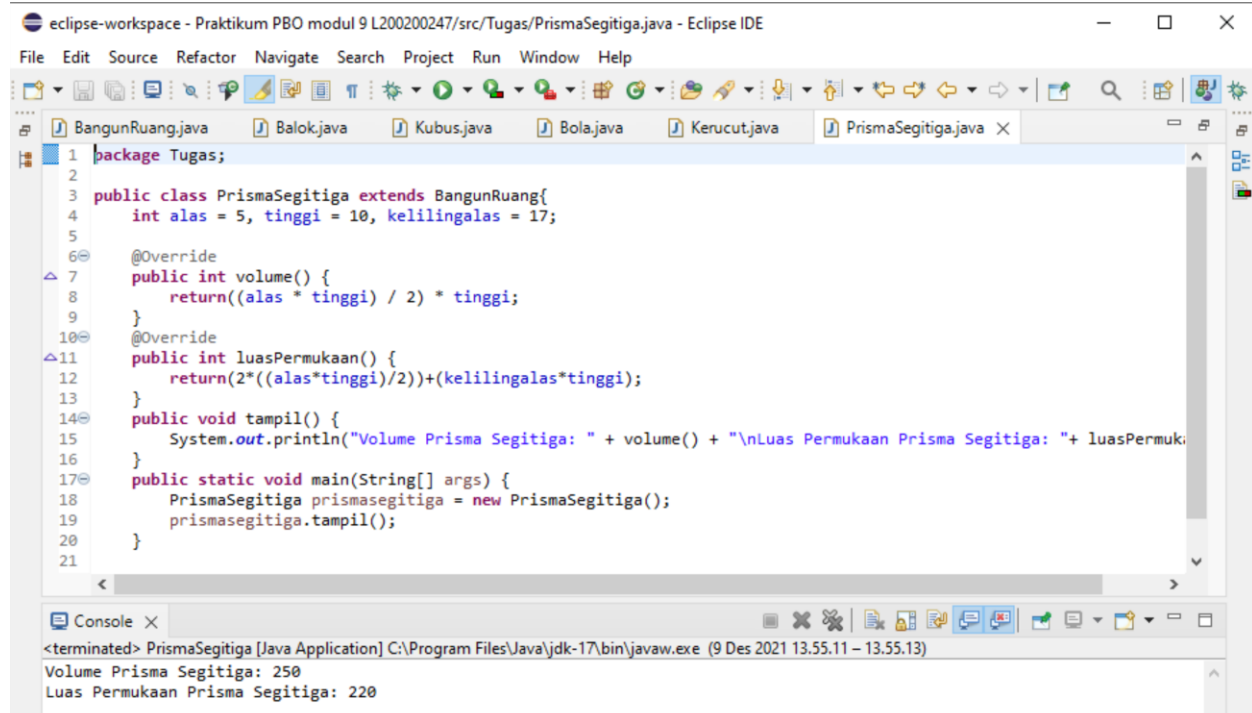
The screenshot shows the Eclipse IDE with the file `Kerucut.java` open. The code defines a class `Kerucut` that extends `BangunRuang`. It includes constants for `phi`, `jarijari`, `tinggi`, and `s`. The `volume()` method calculates the volume using the formula  $\frac{\pi \times \text{jarijari}^2 \times \text{tinggi}}{3}$ . The `luasPermukaan()` method calculates the surface area using the formula  $\pi \times \text{jarijari} \times (\text{jarijari} + s)$ . The `tampil()` method prints both values. The `main` method creates an instance of `Kerucut` and calls `tampil()`.

```
1 package Tugas;
2
3 public class Kerucut extends BangunRuang{
4     double phi = 3.14;
5     int jarijari = 7, tinggi = 10, s = 9;
6
7     @Override
8     public int volume() {
9         return (int) phi * ((jarijari * jarijari) * tinggi) / 3;
10    }
11    @Override
12    public int luasPermukaan() {
13        return (int) phi * jarijari * (jarijari+s);
14    }
15    public void tampil() {
16        System.out.println("Volume Kerucut: " + volume() + "\nLuas Permukaan Kerucut: " + luasPermukaan());
17    }
18    public static void main(String[] args) {
19        Kerucut kerucut = new Kerucut();
20        kerucut.tampil();
21    }
}
```

The console output shows the results of the program execution:

```
<terminated> Kerucut [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 13.41.43 - 13.41.45)
Volume Kerucut: 490
Luas Permukaan Kerucut: 336
```

## Class Prisma Segitiga



The screenshot shows the Eclipse IDE with the file `PrismaSegitiga.java` open. The code defines a class `PrismaSegitiga` that extends `BangunRuang`. It includes constants for `alas`, `tinggi`, and `kelilingalas`. The `volume()` method calculates the volume using the formula  $\frac{\text{alas} \times \text{tinggi}}{2}$ . The `luasPermukaan()` method calculates the surface area using the formula  $2 \times \left( \frac{\text{alas} \times \text{tinggi}}{2} \right) + (\text{kelilingalas} \times \text{tinggi})$ . The `tampil()` method prints both values. The `main` method creates an instance of `PrismaSegitiga` and calls `tampil()`.

```
1 package Tugas;
2
3 public class PrismaSegitiga extends BangunRuang{
4     int alas = 5, tinggi = 10, kelilingalas = 17;
5
6     @Override
7     public int volume() {
8         return (alas * tinggi) / 2 * tinggi;
9     }
10    @Override
11    public int luasPermukaan() {
12        return 2*((alas*tinggi)/2)+(kelilingalas*tinggi);
13    }
14    public void tampil() {
15        System.out.println("Volume Prisma Segitiga: " + volume() + "\nLuas Permukaan Prisma Segitiga: " + luasPermukaan());
16    }
17    public static void main(String[] args) {
18        PrismaSegitiga prismaSegitiga = new PrismaSegitiga();
19        prismaSegitiga.tampil();
20    }
21 }
```

The console output shows the results of the program execution:

```
<terminated> PrismaSegitiga [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (9 Des 2021 13.55.11 - 13.55.13)
Volume Prisma Segitiga: 250
Luas Permukaan Prisma Segitiga: 220
```

