

PRAKTIKUM PEMROGAMAN BERORIENTASI OBJEK

MODUL 9 : ABSTRACT CLASS



Disusun oleh:

AFIFAH GH AISANI IMANA

L200190198

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS KOMUNIKASI DAN INFORMATIKA

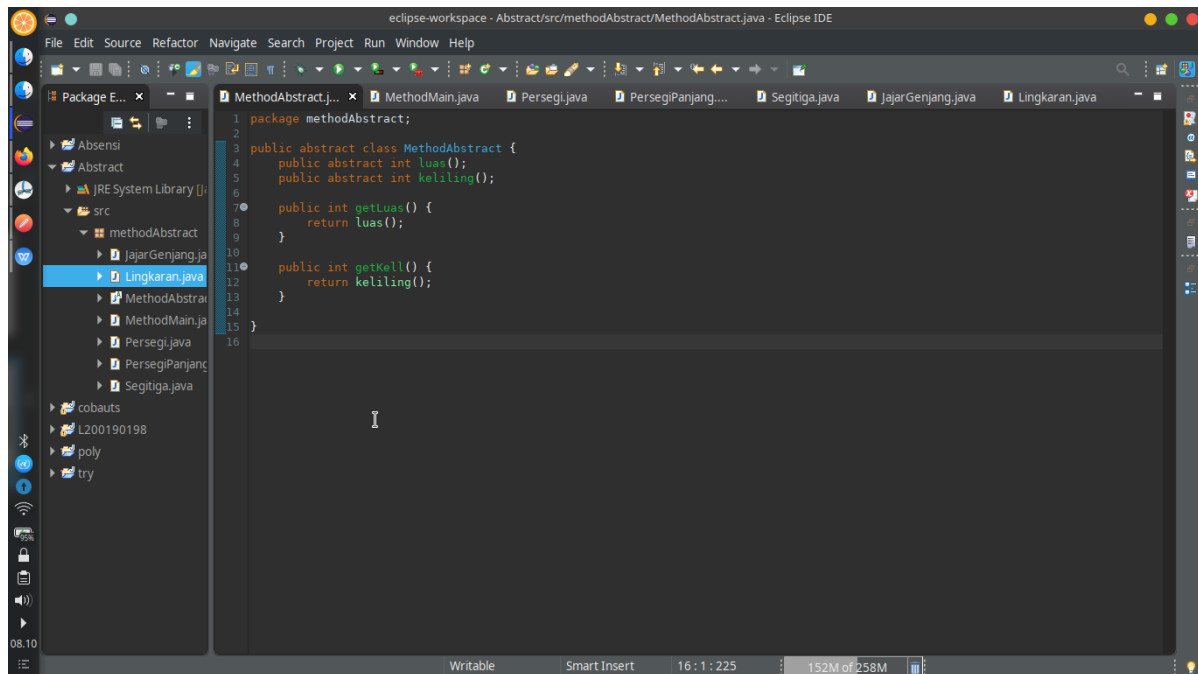
UNIVERSITAS MUHAMMADIYAH SURAKARTA

TAHUN 2019/2020

Latihan.

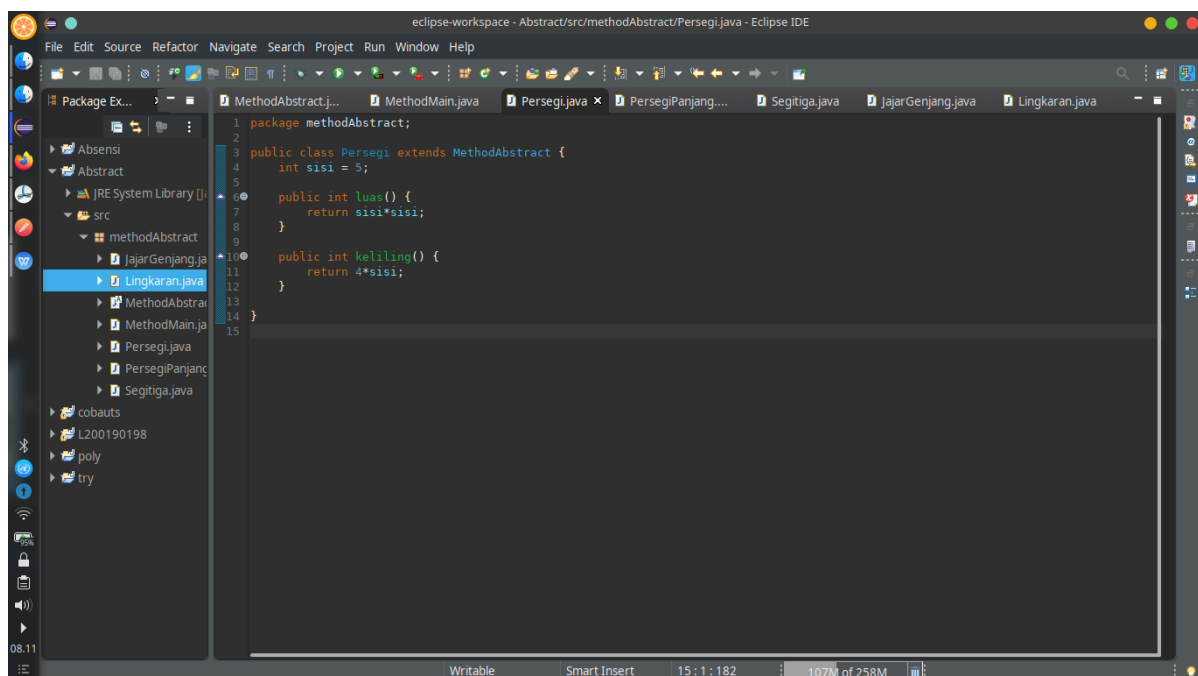
Dengan menggunakan class MethodAbstrak pada Program5 di atas, buatlah class PersegiPanjang, JajarGenjang, Lingkaran, dan Segitiga! Selanjutnya implementasikan method luas() dan keliling() yang sesuai dengan perhitungan masing-masing class.

Class MethodAbstract



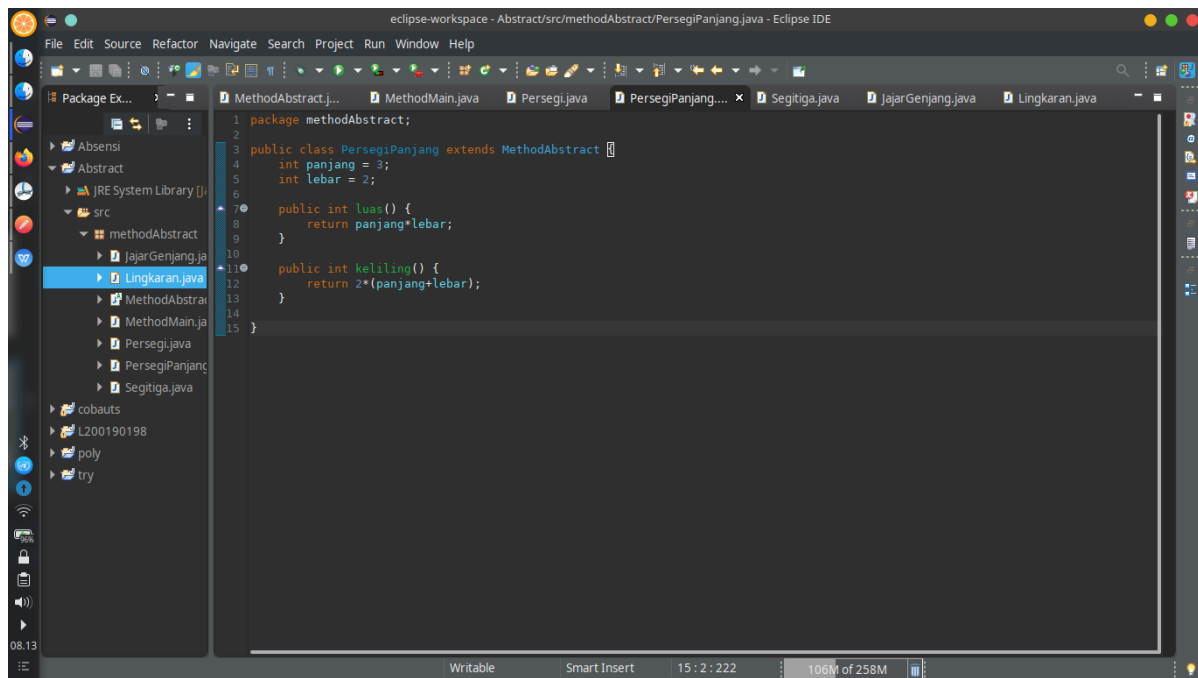
```
1 package methodAbstract;
2
3 public abstract class MethodAbstract {
4     public abstract int luas();
5     public abstract int keliling();
6
7     public int getluas() {
8         return luas();
9     }
10
11     public int getkell() {
12         return keliling();
13     }
14 }
15
16
```

Class Persegi



```
1 package methodAbstract;
2
3 public class Persegi extends MethodAbstract {
4     int sisi = 5;
5
6     public int luas() {
7         return sisi*sisi;
8     }
9
10    public int keliling() {
11        return 4*sisi;
12    }
13 }
14
15
```

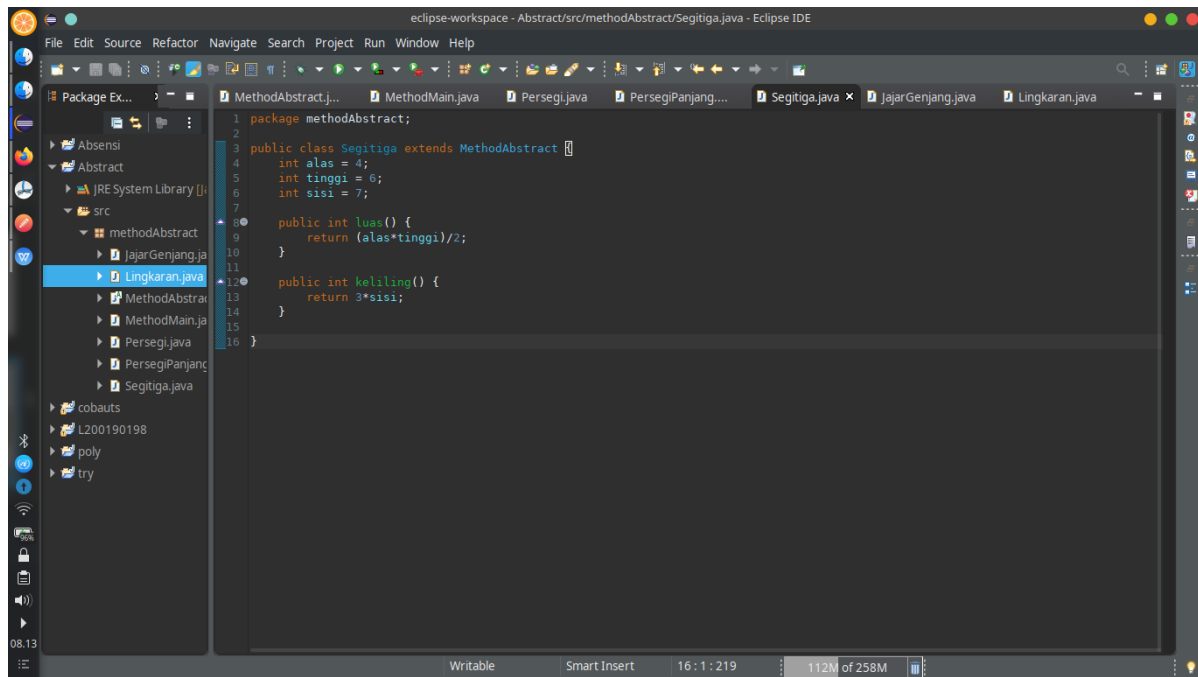
Class Persegi Panjang



The screenshot shows the Eclipse IDE with the file `PersegiPanjang.java` open. The package is `methodAbstract`. The class `PersegiPanjang` extends `MethodAbstract`. It has two attributes: `panjang` (3) and `lebar` (2). It has two methods: `luas()` which returns `panjang*lebar`, and `keliling()` which returns `2*(panjang+lebar)`.

```
1 package methodAbstract;
2
3 public class PersegiPanjang extends MethodAbstract {
4     int panjang = 3;
5     int lebar = 2;
6
7     public int luas() {
8         return panjang*lebar;
9     }
10
11    public int keliling() {
12        return 2*(panjang+lebar);
13    }
14 }
15
```

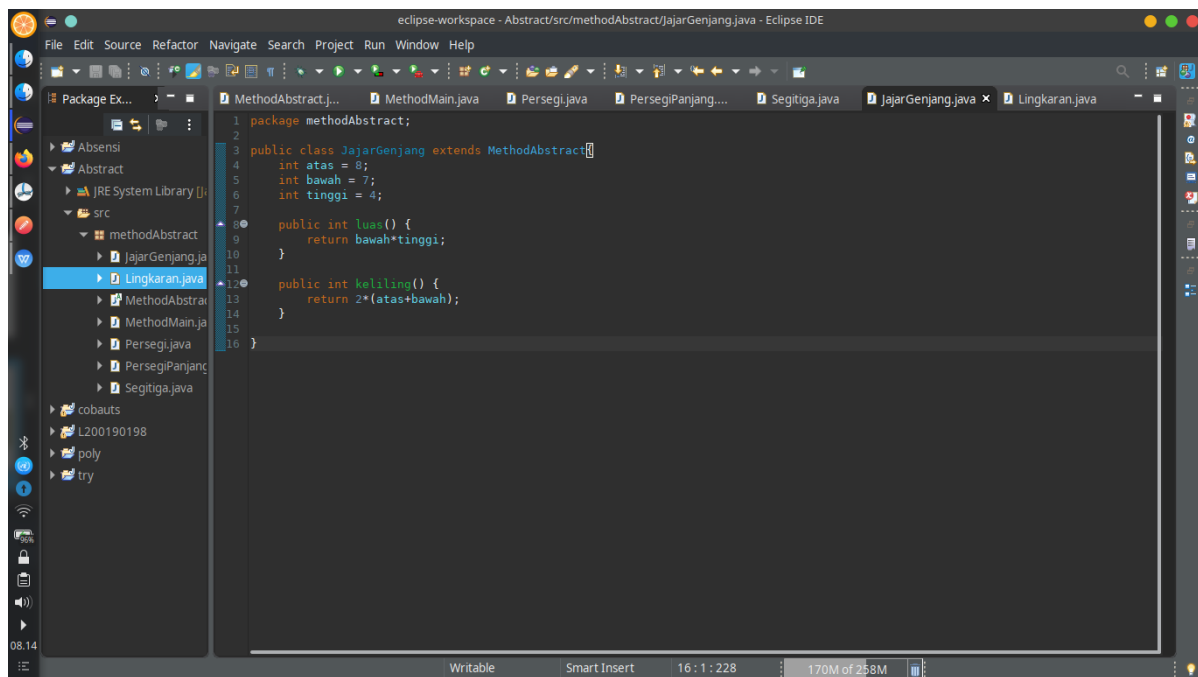
Class Segitiga



The screenshot shows the Eclipse IDE with the file `Segitiga.java` open. The package is `methodAbstract`. The class `Segitiga` extends `MethodAbstract`. It has three attributes: `alas` (4), `tinggi` (6), and `sisi` (7). It has two methods: `luas()` which returns `(alas*tinggi)/2`, and `keliling()` which returns `3*sisi`.

```
1 package methodAbstract;
2
3 public class Segitiga extends MethodAbstract {
4     int alas = 4;
5     int tinggi = 6;
6     int sisi = 7;
7
8     public int luas() {
9         return (alas*tinggi)/2;
10    }
11
12    public int keliling() {
13        return 3*sisi;
14    }
15 }
16
```

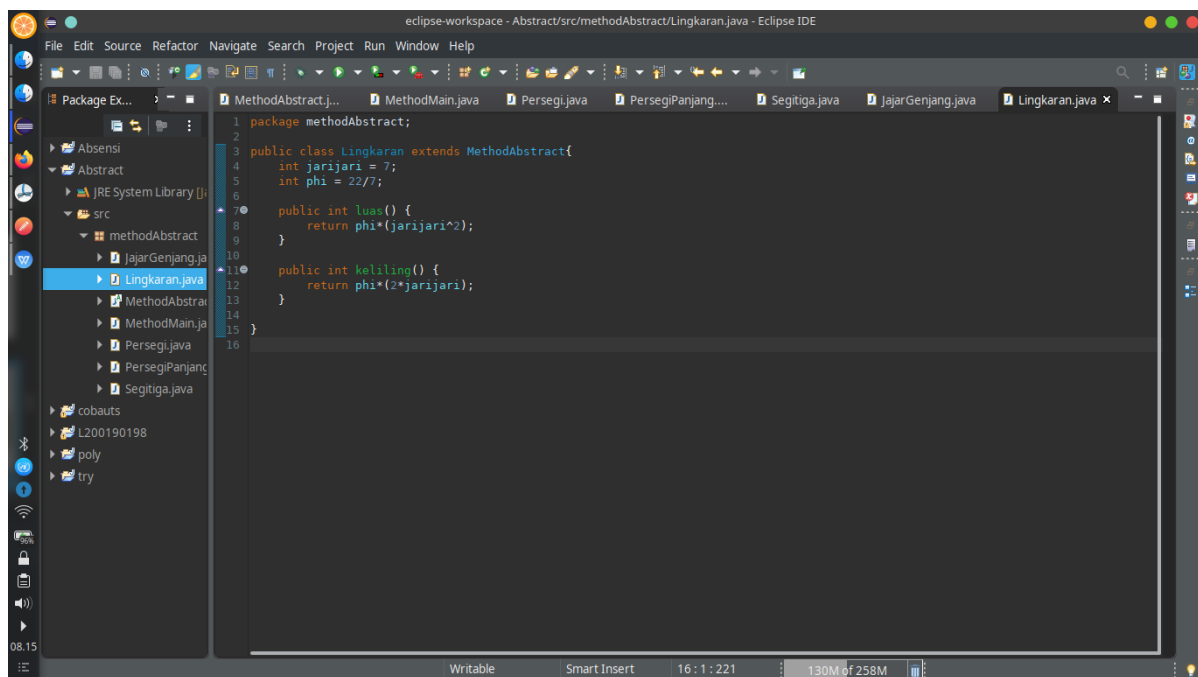
Class JajarGenjang



The screenshot shows the Eclipse IDE interface with the 'JajarGenjang.java' file open. The file is part of the 'methodAbstract' package and extends the 'MethodAbstract' class. The code defines two methods: 'luas()' and 'keliling()'. The 'luas()' method calculates the area based on 'atas' and 'bawah' values, while the 'keliling()' method calculates the perimeter based on 'atas' and 'bawah' values. The IDE's Package Explorer on the left shows the project structure, including the 'methodAbstract' package and its sub-packages.

```
1 package methodAbstract;
2
3 public class JajarGenjang extends MethodAbstract{
4     int atas = 8;
5     int bawah = 7;
6     int tinggi = 4;
7
8     public int luas() {
9         return bawah*tinggi;
10    }
11
12    public int keliling() {
13        return 2*(atas+bawah);
14    }
15 }
16 }
```

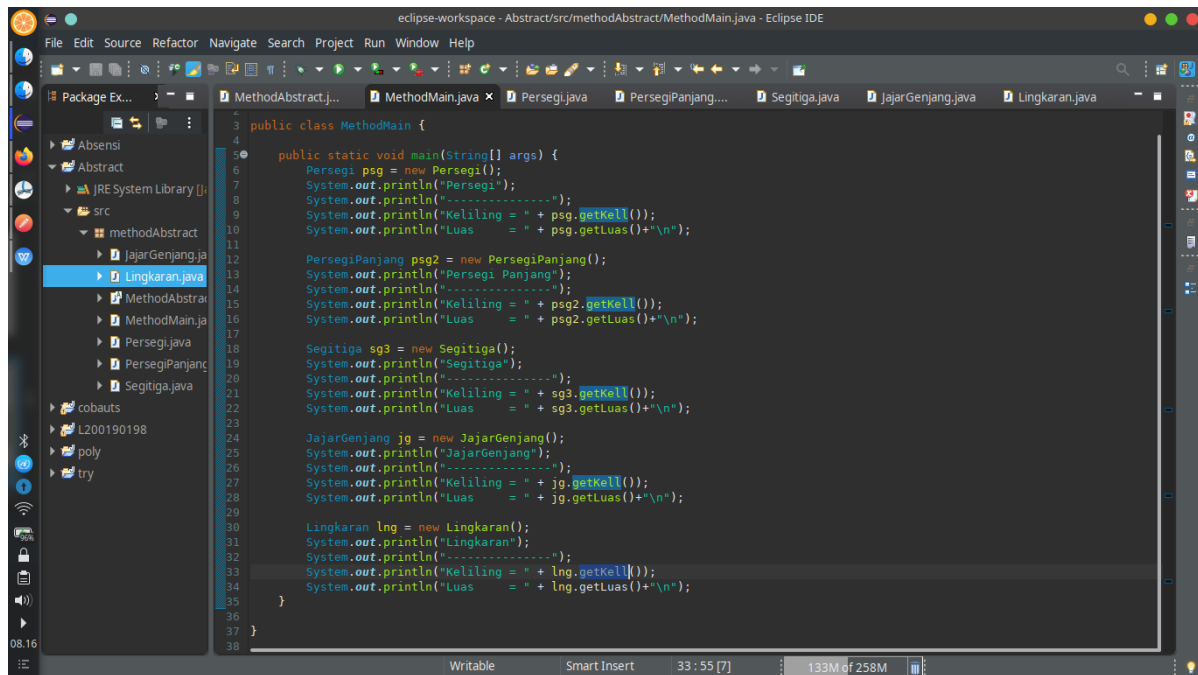
Class Lingkaran



The screenshot shows the Eclipse IDE interface with the 'Lingkaran.java' file open. The file is part of the 'methodAbstract' package and extends the 'MethodAbstract' class. The code defines two methods: 'luas()' and 'keliling()'. The 'luas()' method calculates the area based on 'jarijari' (radius) and 'phi' values, while the 'keliling()' method calculates the perimeter based on 'jarijari' and 'phi' values. The IDE's Package Explorer on the left shows the project structure, including the 'methodAbstract' package and its sub-packages.

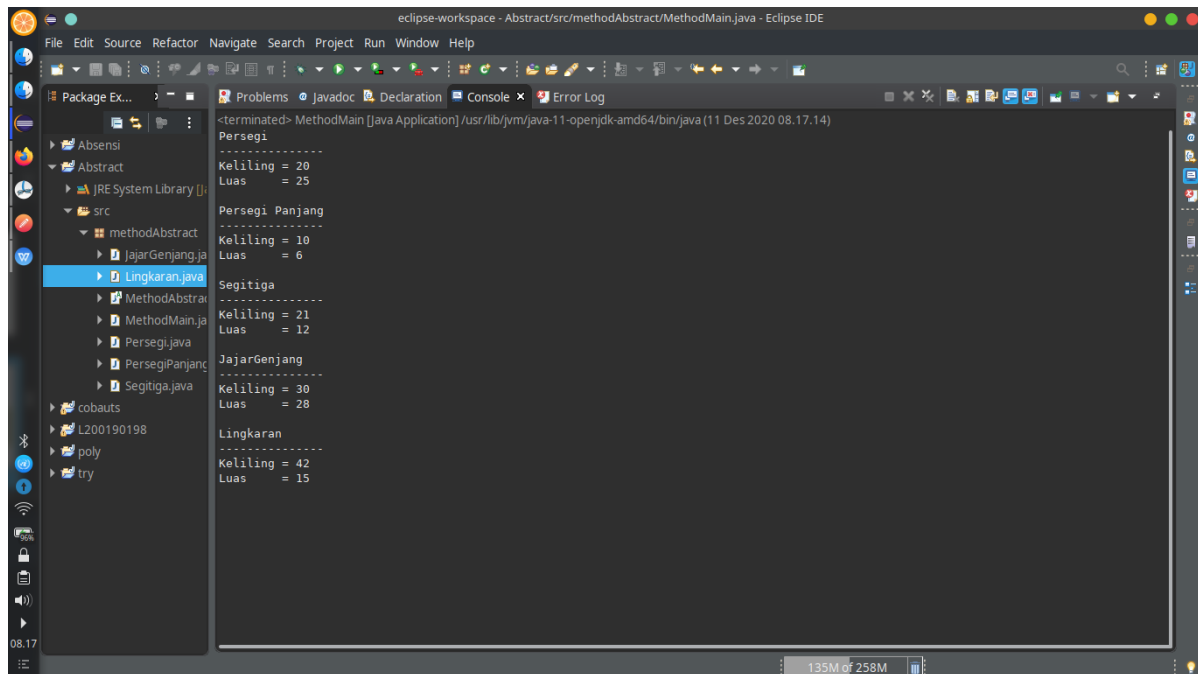
```
1 package methodAbstract;
2
3 public class Lingkaran extends MethodAbstract{
4     int jarijari = 7;
5     int phi = 22/7;
6
7     public int luas() {
8         return phi*(jarijari^2);
9     }
10
11    public int keliling() {
12        return phi*(2*jarijari);
13    }
14 }
15 }
16 }
```

Main Class MethodMain



```
1 public class MethodMain {
2
3     public static void main(String[] args) {
4
5         Persegi psg = new Persegi();
6         System.out.println("Persegi");
7         System.out.println("-----");
8         System.out.println("Keliling = " + psg.getKel();
9         System.out.println("Luas    = " + psg.getLuas()+"\n");
10
11         PersegiPanjang psg2 = new PersegiPanjang();
12         System.out.println("Persegi Panjang");
13         System.out.println("-----");
14         System.out.println("Keliling = " + psg2.getKel();
15         System.out.println("Luas    = " + psg2.getLuas()+"\n");
16
17         Segitiga sg3 = new Segitiga();
18         System.out.println("Segitiga");
19         System.out.println("-----");
20         System.out.println("Keliling = " + sg3.getKel();
21         System.out.println("Luas    = " + sg3.getLuas()+"\n");
22
23         JajarGenjang jg = new JajarGenjang();
24         System.out.println("Jajar Genjang");
25         System.out.println("-----");
26         System.out.println("Keliling = " + jg.getKel();
27         System.out.println("Luas    = " + jg.getLuas()+"\n");
28
29         Lingkaran lng = new Lingkaran();
30         System.out.println("Lingkaran");
31         System.out.println("-----");
32         System.out.println("Keliling = " + lng.getKel();
33         System.out.println("Luas    = " + lng.getLuas()+"\n");
34     }
35 }
36
37 }
38 }
```

Output



```
<terminated>- MethodMain [Java Application] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (11 Des 2020 08.17.14)
Persegi
-----
Keliling = 20
Luas    = 25

Persegi Panjang
-----
Keliling = 10
Luas    = 6

Segitiga
-----
Keliling = 21
Luas    = 12

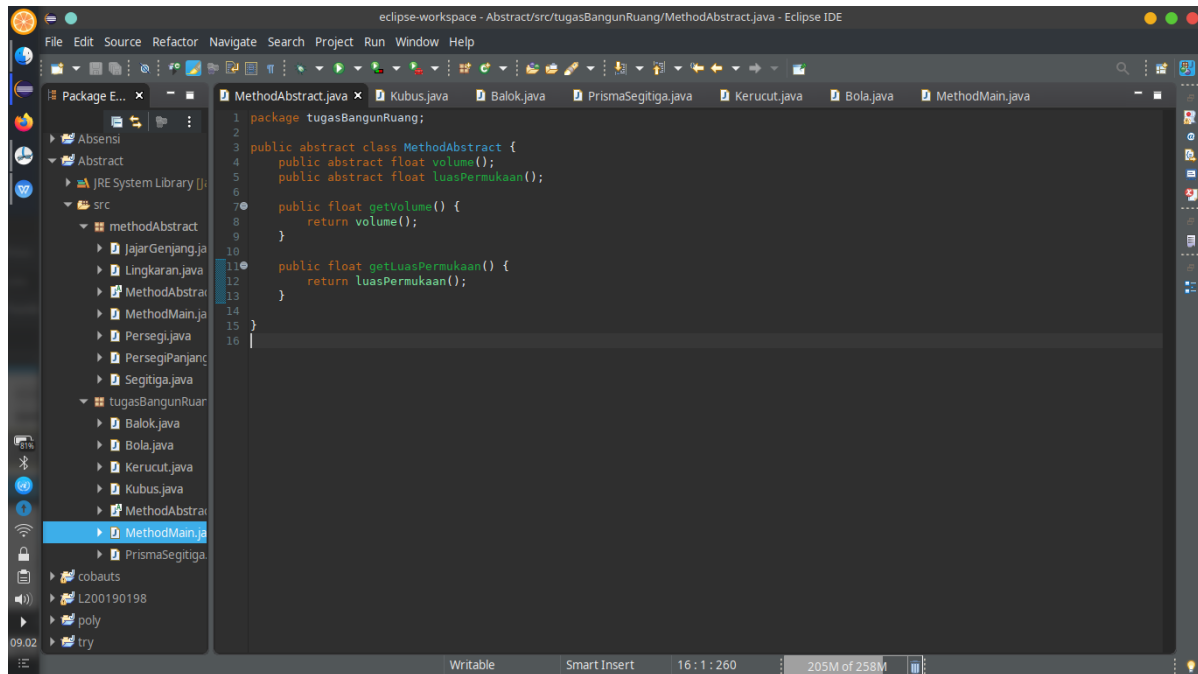
Jajar Genjang
-----
Keliling = 30
Luas    = 28

Lingkaran
-----
Keliling = 42
Luas    = 15
```

Tugas.

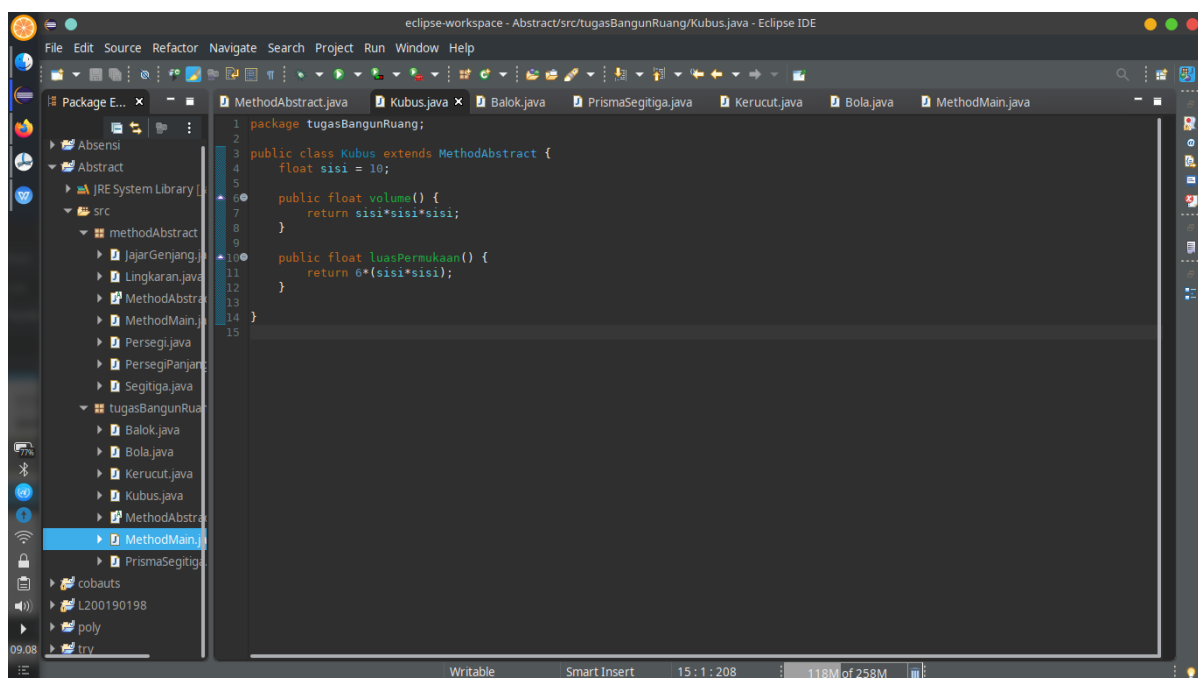
Buatlah class abstract untuk bangun ruang, dengan ketentuan memiliki method abstract untuk menghitung volume, dan luasSelimut/luasPermukaan. Selanjutnya buatlah class Balok, Kubus, Bola, Kerucut, dan PrismaSegitiga untuk mengimplementasikan method abstract tersebut!

Class MethodAbstract



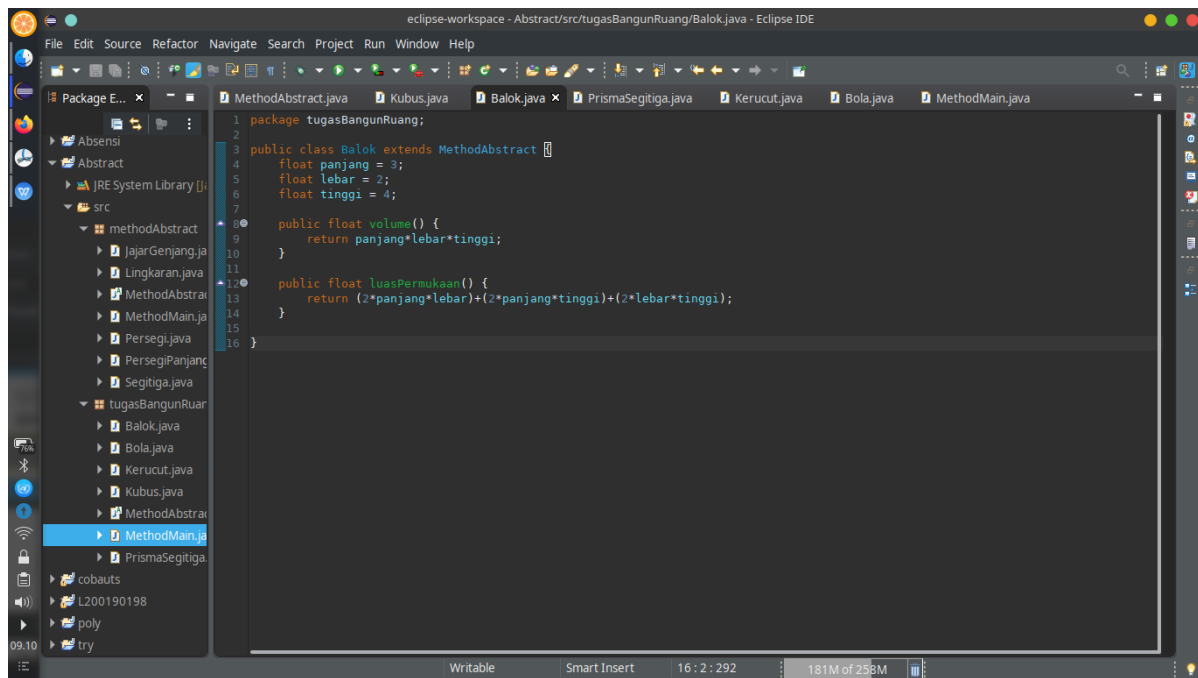
```
1 package tugasBangunRuang;
2
3 public abstract class MethodAbstract {
4     public abstract float volume();
5     public abstract float luasPermukaan();
6
7     public float getVolume() {
8         return volume();
9     }
10
11     public float getLuasPermukaan() {
12         return luasPermukaan();
13     }
14 }
15
16
```

Class Kubus



```
1 package tugasBangunRuang;
2
3 public class Kubus extends MethodAbstract {
4     float sisi = 10;
5
6     public float volume() {
7         return sisi*sisi*sisi;
8     }
9
10    public float luasPermukaan() {
11        return 6*sisi*sisi;
12    }
13 }
14
15
```

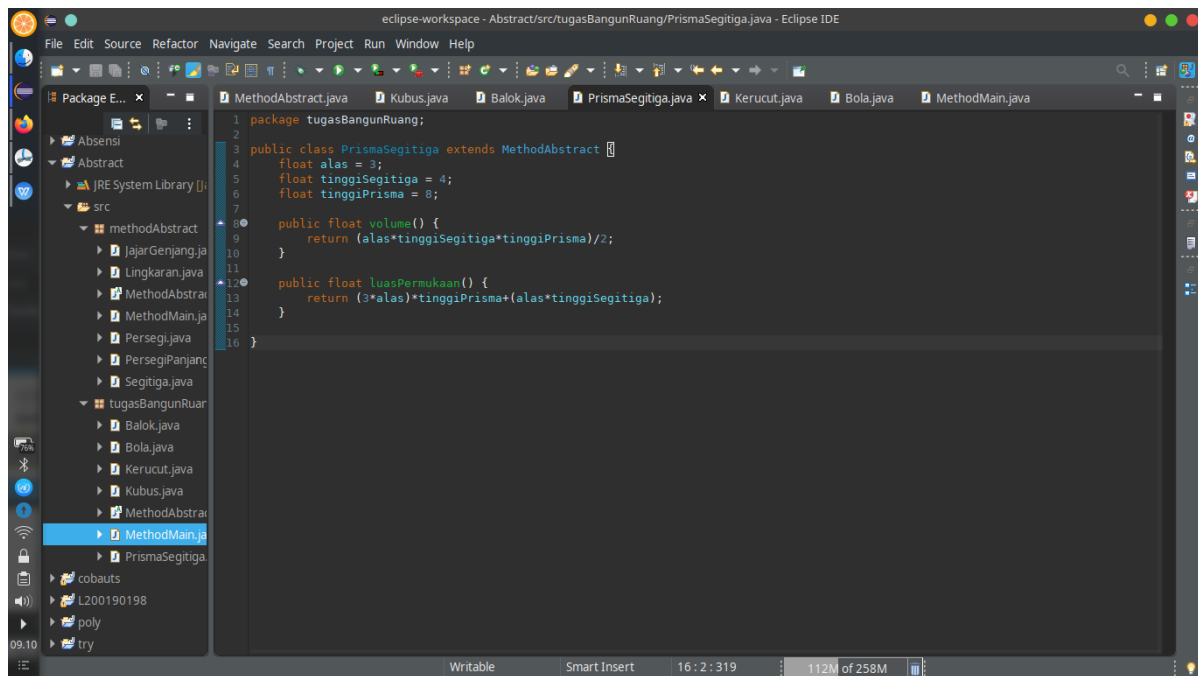
Class Balok



The screenshot shows the Eclipse IDE with the 'Balok.java' file open. The code defines a class 'Balok' that extends 'MethodAbstract'. It includes attributes for length, width, and height, and methods for calculating volume and surface area.

```
1 package tugasBangunRuang;
2
3 public class Balok extends MethodAbstract {
4     float panjang = 3;
5     float lebar = 2;
6     float tinggi = 4;
7
8     public float volume() {
9         return panjang*lebar*tinggi;
10    }
11
12    public float luasPermukaan() {
13        return (2*panjang*lebar)+(2*panjang*tinggi)+(2*lebar*tinggi);
14    }
15 }
16
```

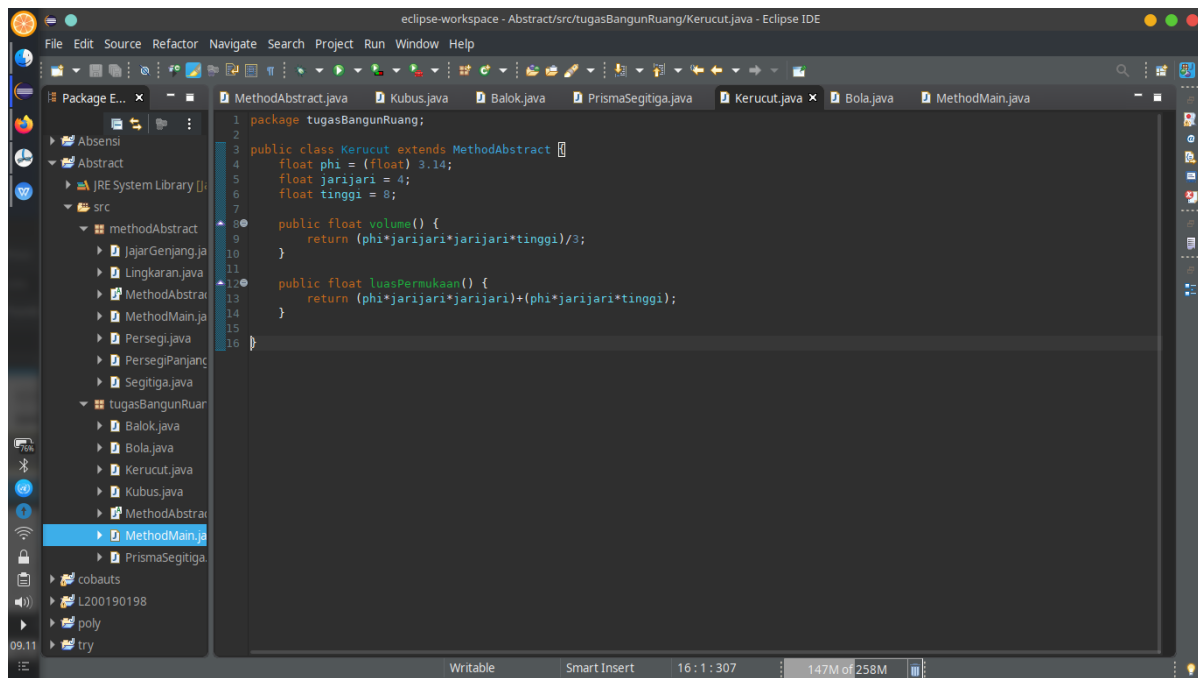
Class PrismaSegitiga



The screenshot shows the Eclipse IDE with the 'PrismaSegitiga.java' file open. The code defines a class 'PrismaSegitiga' that extends 'MethodAbstract'. It includes attributes for base length, height of the triangle, and height of the prism, and methods for calculating volume and surface area.

```
1 package tugasBangunRuang;
2
3 public class PrismaSegitiga extends MethodAbstract {
4     float alas = 3;
5     float tinggiSegitiga = 4;
6     float tinggiPrisma = 8;
7
8     public float volume() {
9         return (alas*tinggiSegitiga*tinggiPrisma)/2;
10    }
11
12    public float luasPermukaan() {
13        return (3*alas)*tinggiPrisma+(alas*tinggiSegitiga);
14    }
15 }
16
```

Class Kerucut

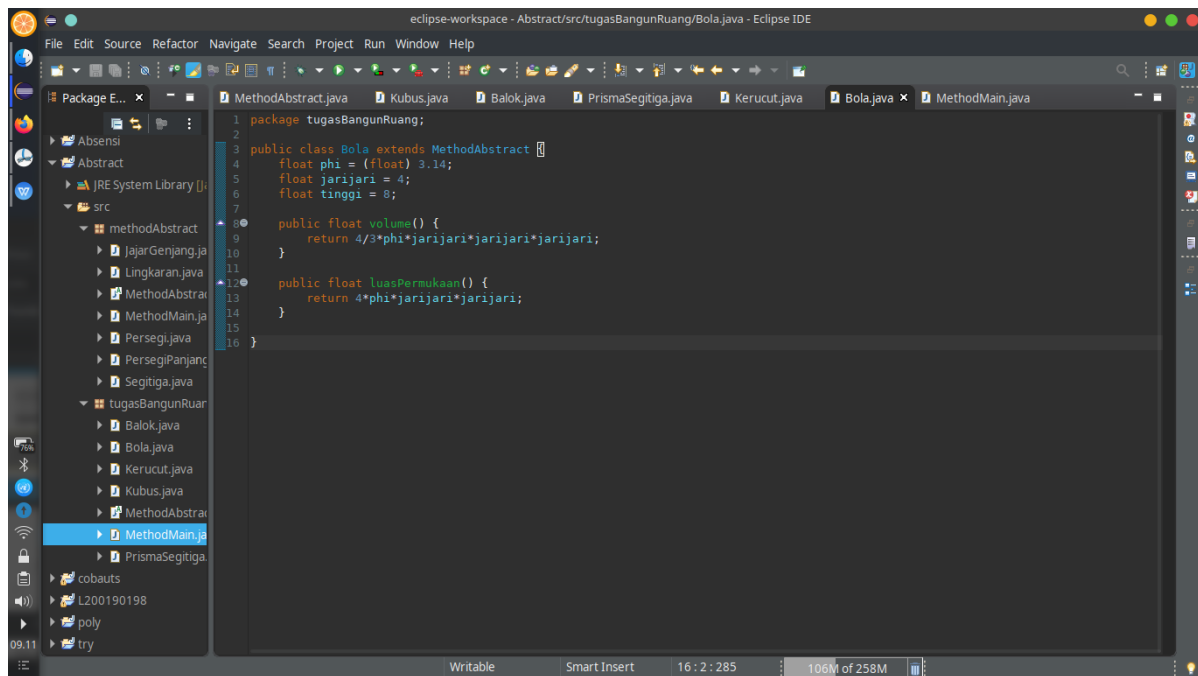


The screenshot shows the Eclipse IDE with the 'Kerucut.java' file open. The file is part of the 'tugasBangunRuang' package and extends the 'MethodAbstract' class. It contains the following code:

```
1 package tugasBangunRuang;
2
3 public class Kerucut extends MethodAbstract {
4     float phi = (float) 3.14;
5     float jarijari = 4;
6     float tinggi = 8;
7
8     public float volume() {
9         return (phi*jarijari*jarijari*tinggi)/3;
10    }
11
12    public float luasPermukaan() {
13        return (phi*jarijari*jarijari)+(phi*jarijari*tinggi);
14    }
15 }
16
```

The IDE interface includes a Package Explorer on the left showing the project structure, a top toolbar, and a bottom status bar indicating 'Writable', 'Smart Insert', and memory usage (16:1:307, 147M of 258M).

Class Bola

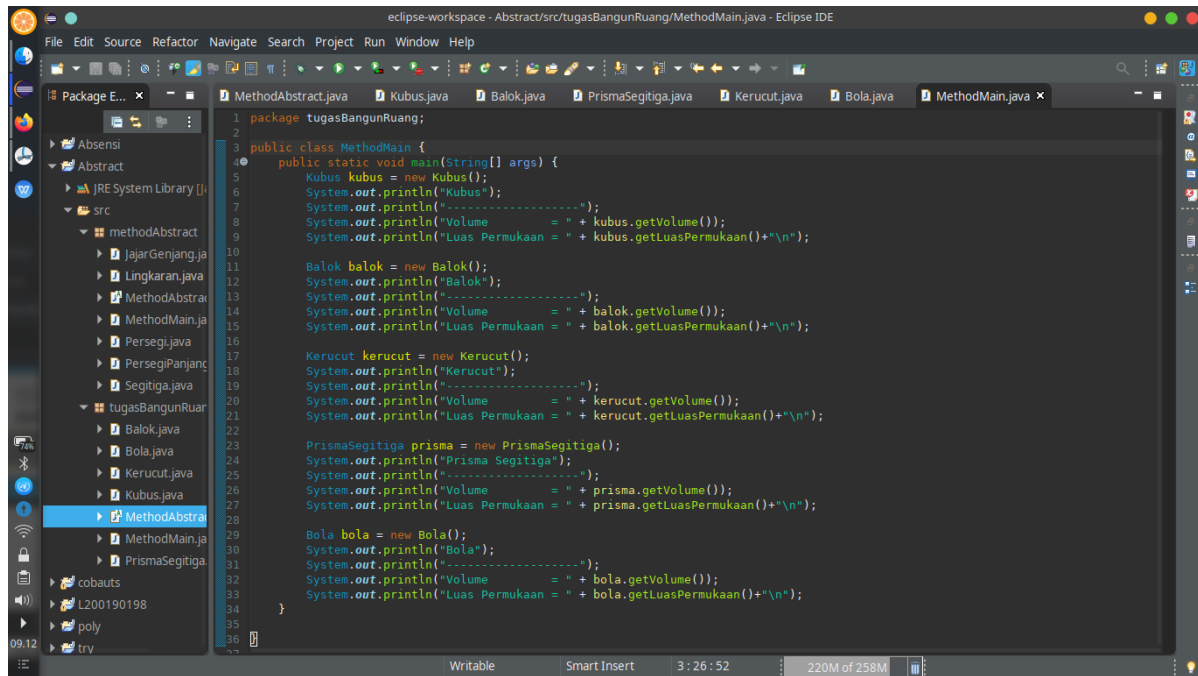


The screenshot shows the Eclipse IDE with the 'Bola.java' file open. The file is part of the 'tugasBangunRuang' package and extends the 'MethodAbstract' class. It contains the following code:

```
1 package tugasBangunRuang;
2
3 public class Bola extends MethodAbstract {
4     float phi = (float) 3.14;
5     float jarijari = 4;
6     float tinggi = 8;
7
8     public float volume() {
9         return 4/3*phi*jarijari*jarijari*jarijari;
10    }
11
12    public float luasPermukaan() {
13        return 4*phi*jarijari*jarijari;
14    }
15 }
16
```

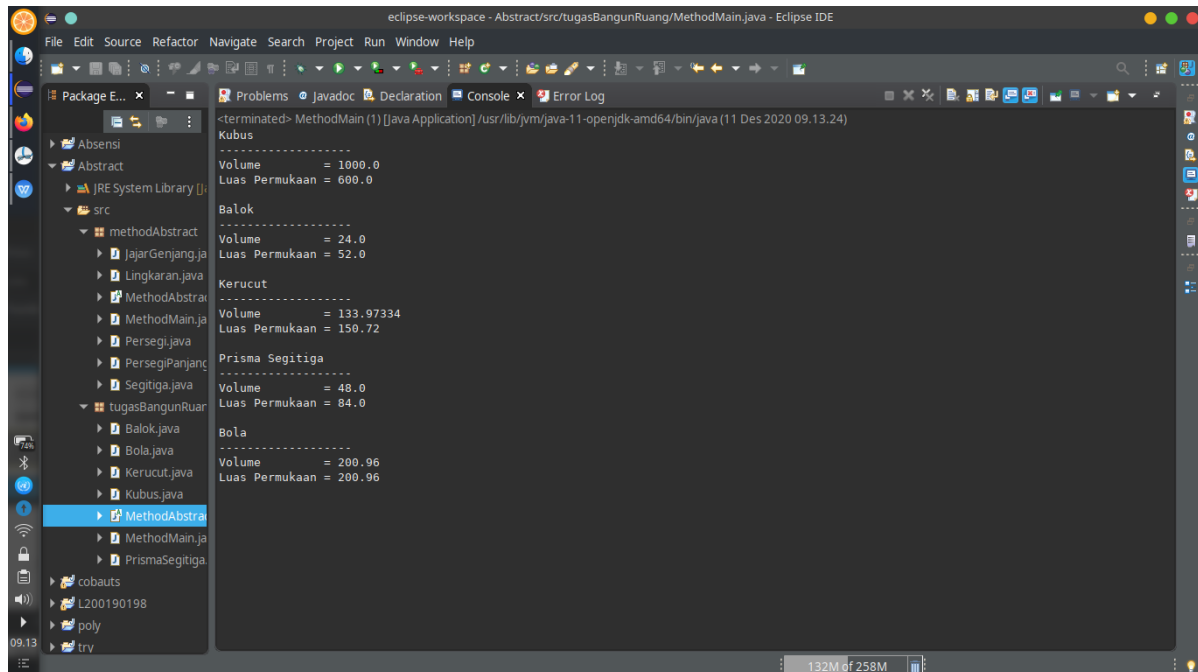
The IDE interface is similar to the previous screenshot, showing the same project structure and status bar information (16:2:285, 106M of 258M).

Main Class MethodMain



```
1 package tugasBangunRuang;
2
3 public class MethodMain {
4     public static void main(String[] args) {
5         Kubus kubus = new Kubus();
6         System.out.println("Kubus");
7         System.out.println("-----");
8         System.out.println("Volume      = " + kubus.getVolume());
9         System.out.println("Luas Permukaan = " + kubus.getLuasPermukaan()+"\n");
10
11         Balok balok = new Balok();
12         System.out.println("Balok");
13         System.out.println("-----");
14         System.out.println("Volume      = " + balok.getVolume());
15         System.out.println("Luas Permukaan = " + balok.getLuasPermukaan()+"\n");
16
17         Kerucut kerucut = new Kerucut();
18         System.out.println("Kerucut");
19         System.out.println("-----");
20         System.out.println("Volume      = " + kerucut.getVolume());
21         System.out.println("Luas Permukaan = " + kerucut.getLuasPermukaan()+"\n");
22
23         PrismaSegitiga prisma = new PrismaSegitiga();
24         System.out.println("Prisma Segitiga");
25         System.out.println("-----");
26         System.out.println("Volume      = " + prisma.getVolume());
27         System.out.println("Luas Permukaan = " + prisma.getLuasPermukaan()+"\n");
28
29         Bola bola = new Bola();
30         System.out.println("Bola");
31         System.out.println("-----");
32         System.out.println("Volume      = " + bola.getVolume());
33         System.out.println("Luas Permukaan = " + bola.getLuasPermukaan()+"\n");
34     }
35 }
36
```

Output



```
<terminated>- MethodMain (1) [Java Application] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (11 Des 2020 09:13:24)
Kubus
Volume      = 1000.0
Luas Permukaan = 600.0

Balok
Volume      = 24.0
Luas Permukaan = 52.0

Kerucut
Volume      = 133.97334
Luas Permukaan = 150.72

Prisma Segitiga
Volume      = 48.0
Luas Permukaan = 84.0

Bola
Volume      = 200.96
Luas Permukaan = 200.96
```