JOBSHEET III

**ARRAY OF OBJEK**

**Nama: Rizqi Bagus Andrean**

**Absen: 25**

**Kelas: TI-1D**

# Tujuan Praktikum

Setelah melakukan materi praktikum ini, mahasiswa mampu:

1. Memahami dan menjelaskan fungsi array yang berisikan variabel objek.
2. Mahasiswa mampu menangkap logika tentang permasalahan array of object dalam Java
3. Mahasiswa mampu menerapkan pembuatan array of object dalam Java

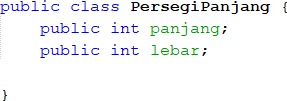
# Percobaan 1: Membuat Array dari Object, Mengisi dan Menampilkan

## Waktu: 50 menit

Didalam praktikum ini, kita akan mempraktekkan bagaimana membuat array dari object, kemudian mengisi dan menampilkan array tersebut.

## Langkah-langkah Percobaan

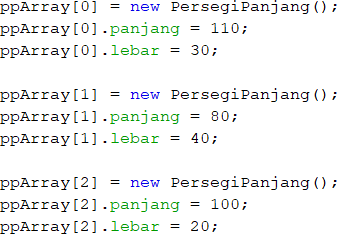
1. Buat Project baru, dengan nama “**ArrayObjects**”.
2. Buat class **PersegiPanjang**:



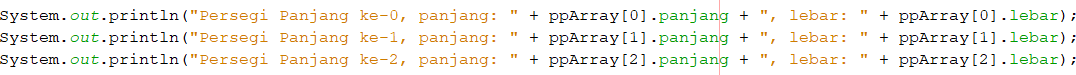
1. Pada fungsi main yaitu pada class **ArrayObjects**, buatlah array **PersegiPanjang** yang berisi 3 elemen:



1. Kemudian isikan masing-masing atributnya:



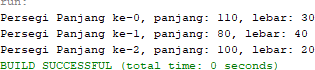
1. Cetak ke layar semua atribut dari objek **ppArray**:



1. Jalankan dan amati hasilnya.

## Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program anda dengan gambar berikut ini.



## Pertanyaan

1. Berdasarkan uji coba 3.2, apakah class yang akan dibuat array of object harus selalu memiliki atribut dan sekaligus method?Jelaskan!

Tidak, Class yang akan dibuat array of object **tidak harus** memiliki atribut dan sekaligus method, **tetapi harus memiliki minimal satu dari keduanya**.

1. Apakah class PersegiPanjang memiliki konstruktor?Jika tidak, kenapa dilakukan pemanggilan konstruktur pada baris program berikut :



Kalau tidak ada konstruktor, maka otomatis akan ada constructor defaultnya.

1. Apa yang dimaksud dengan kode berikut ini:



Membuat array dengan tipe data persegi panjang dengan jumlah maksimal 3.

1. Apa yang dimaksud dengan kode berikut ini:



Mengisi ppArray index ke 1 dengan instance dari Persegi panjang, lalu mengisi atributnya .

1. Mengapa class main dan juga class PersegiPanjang dipisahkan pada uji coba 3.2?

Tidak apa-apa, karena class PersegiPanjang bersifat public jadi bisa dipanggil dari class lain.

# Percobaan 2: Menerima Input Isian Array Menggunakan Looping

## Waktu: 50 menit

Pada praktikum ini kita akan mengubah hasil program dari praktikum 3.2 sehingga program dapat menerima input dan menggunakan looping untuk mengisikan atribut dari semua persegi panjang yang ada di ppArray.

## Langkah-langkah Percobaan

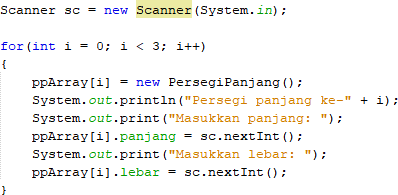
1. Import scanner pada class **ArrayObjects**.



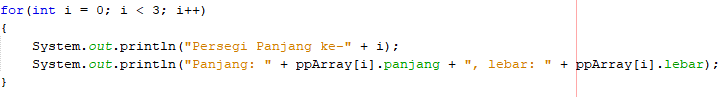
Note: Letakkan kode import dibawah kode package.

1. Pada praktikum 3.2 poin nomor 4, ganti kodenya dengan kode berikut ini, yaitu membuat objek

**Scanner** untuk menerima input, kemudian melakukan looping untuk menerima input:



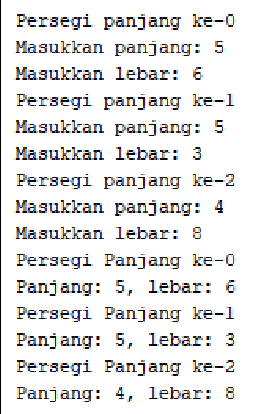
1. Pada praktikum 3.2 poin nomor 5, ganti kodenya dengan berikut ini, yaitu melakukan looping untuk mengakses isi array **ppArray** dan menampilkannya ke layar:



1. Jalankan dan amati hasilnya.

## Verifikasi Hasil Percobaan

Contoh verifikasi hasil percobaan ini.



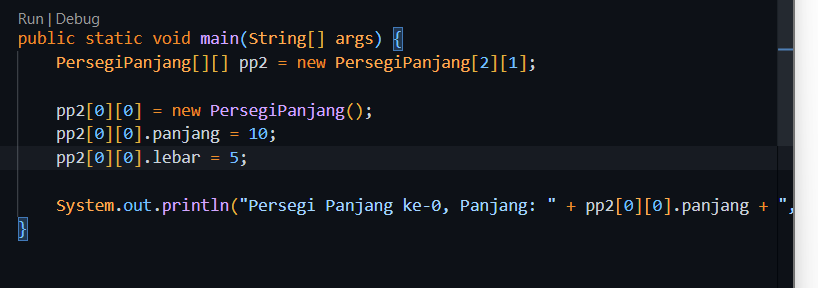
## 

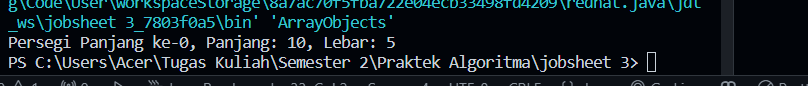
## Pertanyaan

1. Apakah array of object dapat diimplementasikan pada array 2 Dimensi?

Ya

1. Jika jawaban soal no satu iya, berikan contohnya! Jika tidak, jelaskan!



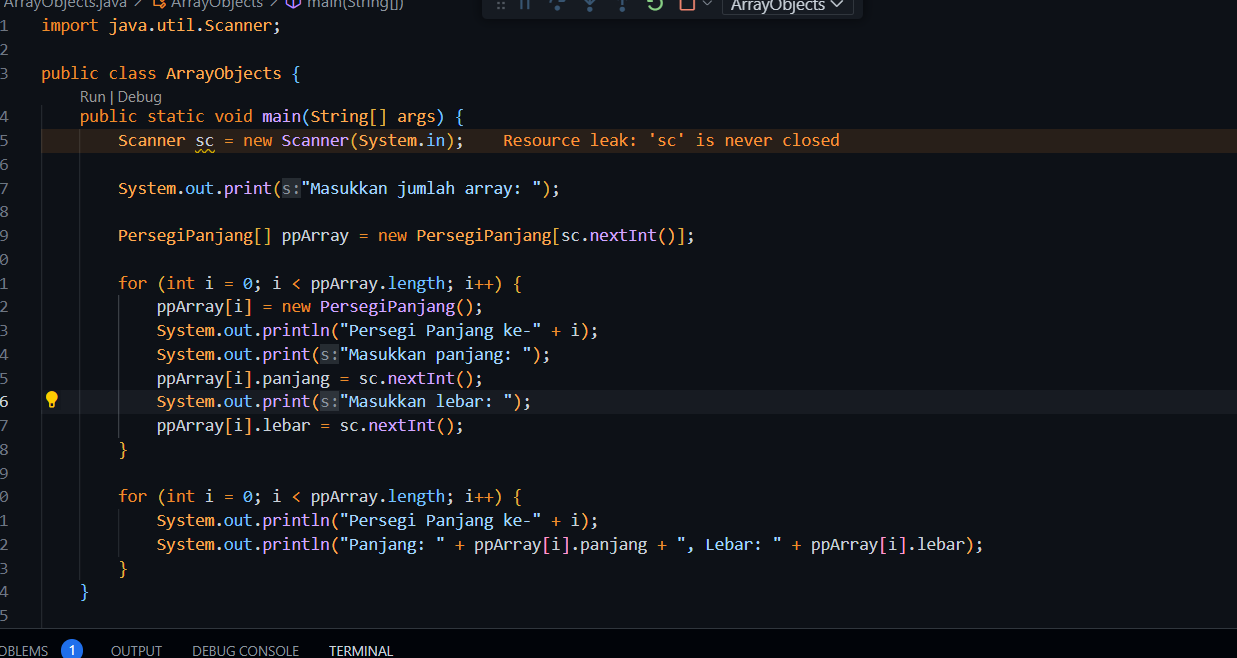


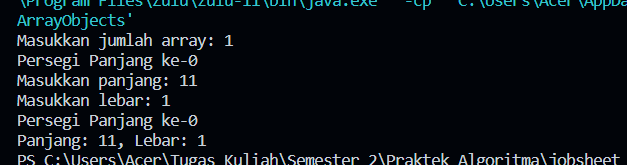
1. Jika diketahui terdapat class **Persegi** yang memiliki atribut sisi bertipe integer, maka kode dibawah ini akan memunculkan error saat dijalankan. Mengapa?



Karen ppArray pada index ke lima belum ada instance dari Persegi.

1. Modifikasi kode program pada praktikum 3.3 agar length array menjadi inputan dengan Scanner!





1. Apakah boleh Jika terjadi duplikasi instansiasi array of objek, misalkan saja instansiasi dilakukan pada **ppArray[i]** sekaligus **ppArray[0]**?Jelaskan !

Bisa, instance yang ada di ppArray index yang lama akan diganti yang baru..

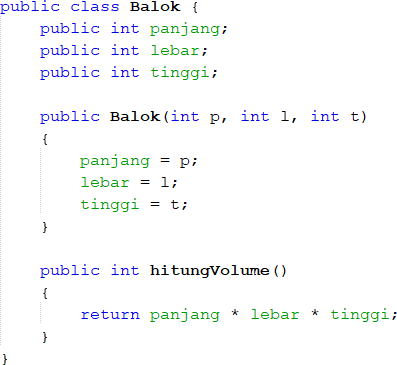
# Percobaan 3: Penambahan Operasi Matematika di Dalam Method

## Waktu: 50 menit

Pada praktikum ini kita akan melakukan pengoperasian matematika beberapa atribut pada masing-masing anggota array.

## Langkah-langkah Percobaan

1. Buat package baru “**ArrayBalok**”.
2. Buat class **Balok**:



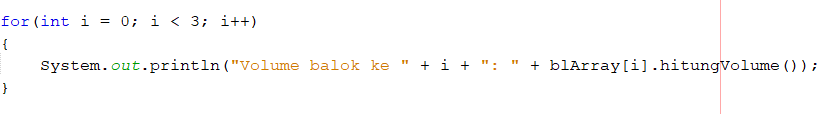
1. Pada fungsi **main** yaitu pada class **ArrayBalok**, buat array **Balok** yang berisi 3 elemen:



1. Kemudian tambahkan kode berikut ini untuk mengisi array **blArray** menggunakan konstruktor dari class **Balok**:



1. Tampilkan semua volume balok tersebut dengan cara memanggil method **hitungVolume()** di dalam looping seperti berikut ini:

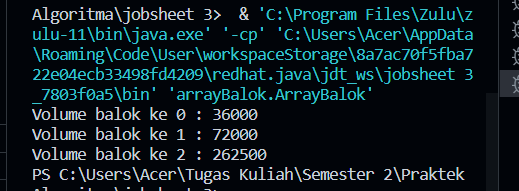


1. Jalankan dan amati hasilnya.

## Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program anda dengan gambar berikut ini.





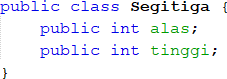
## Pertanyaan

* + - 1. Dapatkah konstruktor berjumlah lebih dalam satu kelas? Jelaskan dengan contoh!

Bisa.

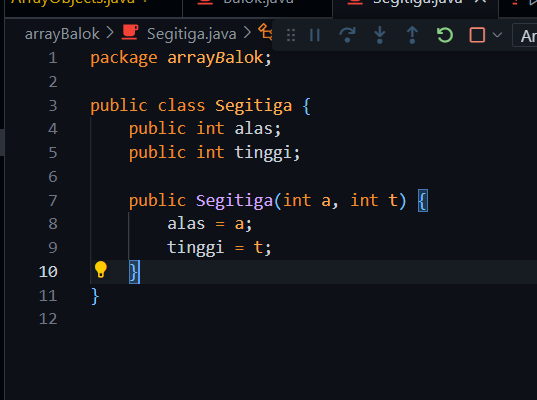


* + - 1. Jika diketahui terdapat class **Segitiga** seperti berikut ini:

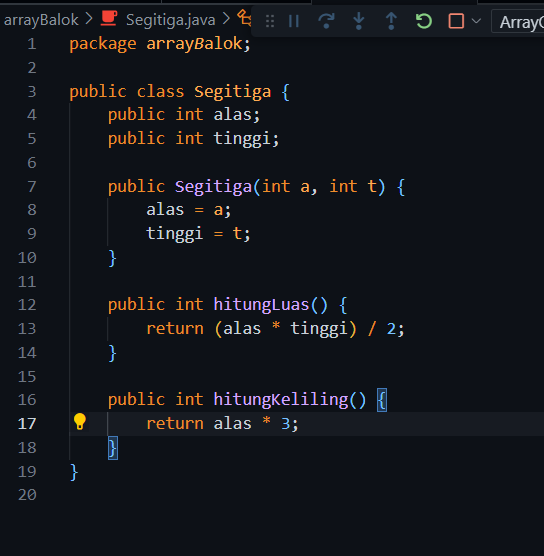


Tambahkan konstruktor pada class **Segitiga** tersebut yang berisi parameter **int a**, **int t**

yang masing-masing digunakan untuk mengisikan atribut alas dan tinggi.



* + - 1. Tambahkan method **hitungLuas()** dan **hitungKeliling()** pada class **Segitiga** tersebut. **Asumsi segitiga adalah segitiga siku-siku**. (*Hint: Anda dapat menggunakan bantuan library Math pada Java untuk mengkalkulasi sisi miring*)



* + - 1. Pada fungsi **main**, buat array Segitiga **sgArray** yang berisi 4 elemen, isikan masing-masing atributnya sebagai berikut:

sgArray ke-0 alas: 10, tinggi: 4

sgArray ke-1 alas: 20, tinggi: 10

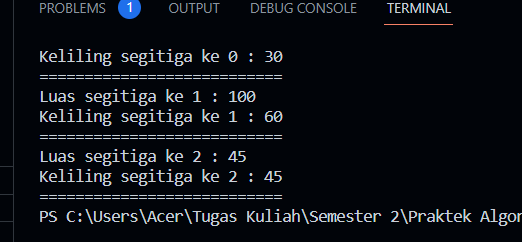
sgArray ke-2 alas: 15, tinggi: 6

sgArray ke-3 alas: 25, tinggi: 10



* + - 1. Kemudian menggunakan looping, cetak luas dan keliling dengan cara memanggil method

**hitungLuas()** dan **hitungKeliling()**.



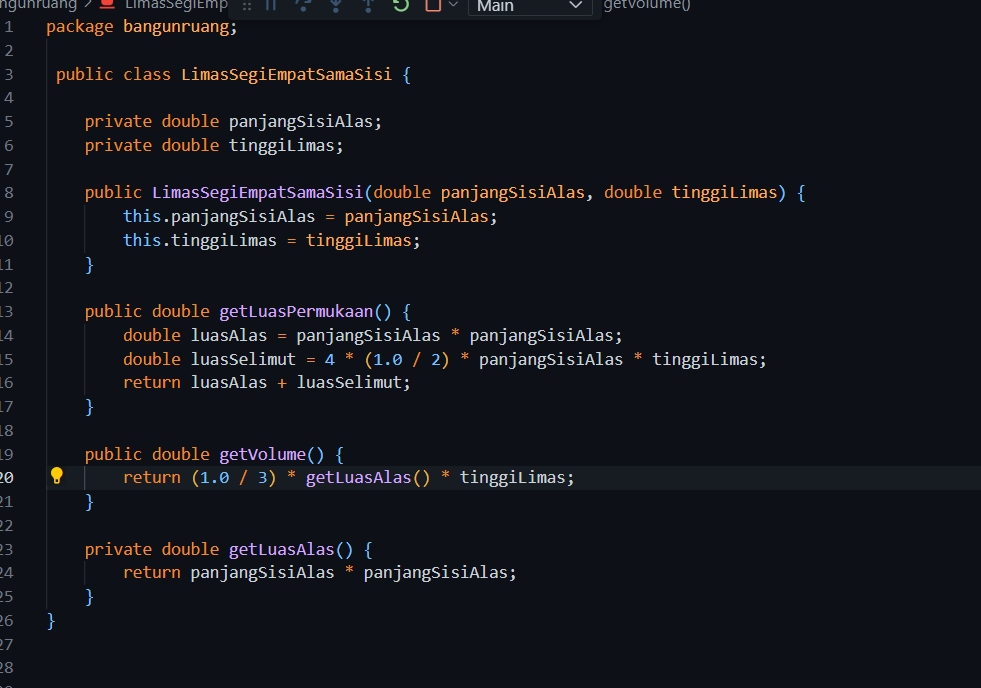
# Latihan Praktikum

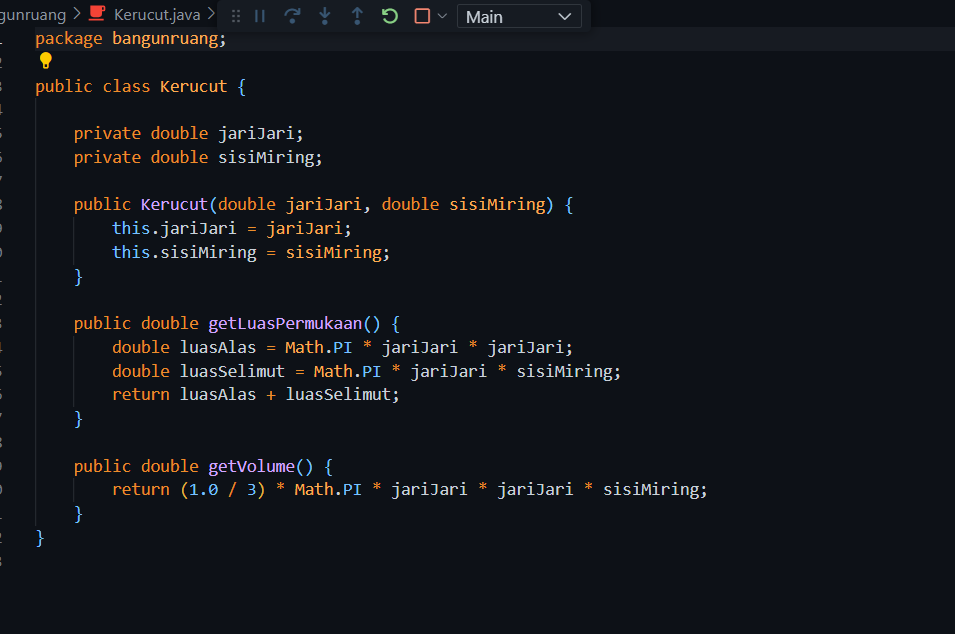
## Waktu: 150 menit

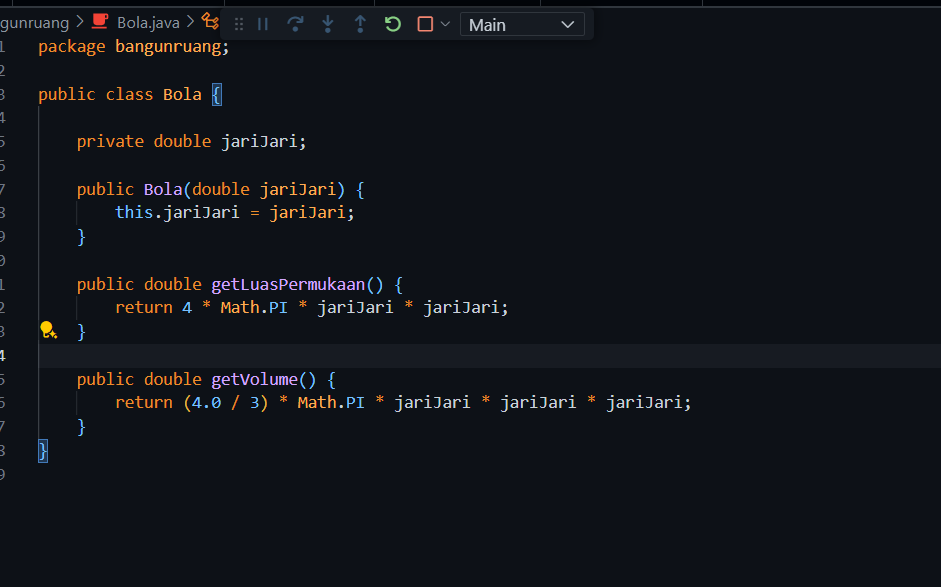
1. Buatlah program yang dapat **menghitung luas permukaan dan volume** bangun ruang **kerucut, limas segi empat sama sisi, dan bola**. Buatlah 3 (tiga) class sesuai dengan jumlah jenis bangun ruang. Buatlah satu main class untuk membuat *array of objects* yang menginputkan atribut- atribut yang ada menggunakan konstruktor semua bangun ruang tersebut. Dengan ketentuan,

## Buat looping untuk menginputkan masing-masing atributnya, kemudian tampilkan luas permukaan dan volume dari tiap jenis bangun ruang tersebut.

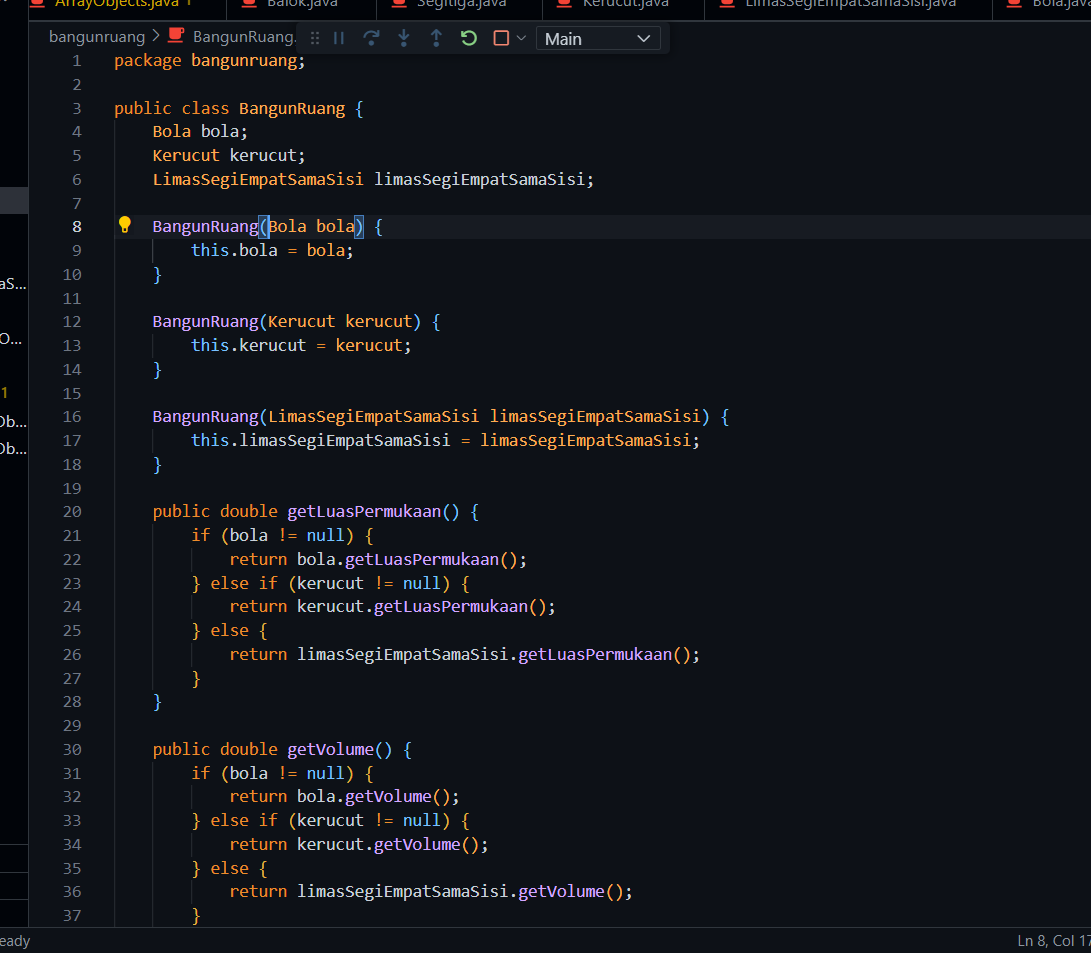
* 1. ***Pada kerucut, inputan untuk atribut hanya jari-jari dan sisi miring***
  2. ***Pada limas segi empat sama sisi, inputan untuk atribut hanya panjang sisi alas dan tinggi limas***
  3. ***Pada bola, inpuntan untuk atribut hanya jari-jari***

**LimasSegiEmpat**

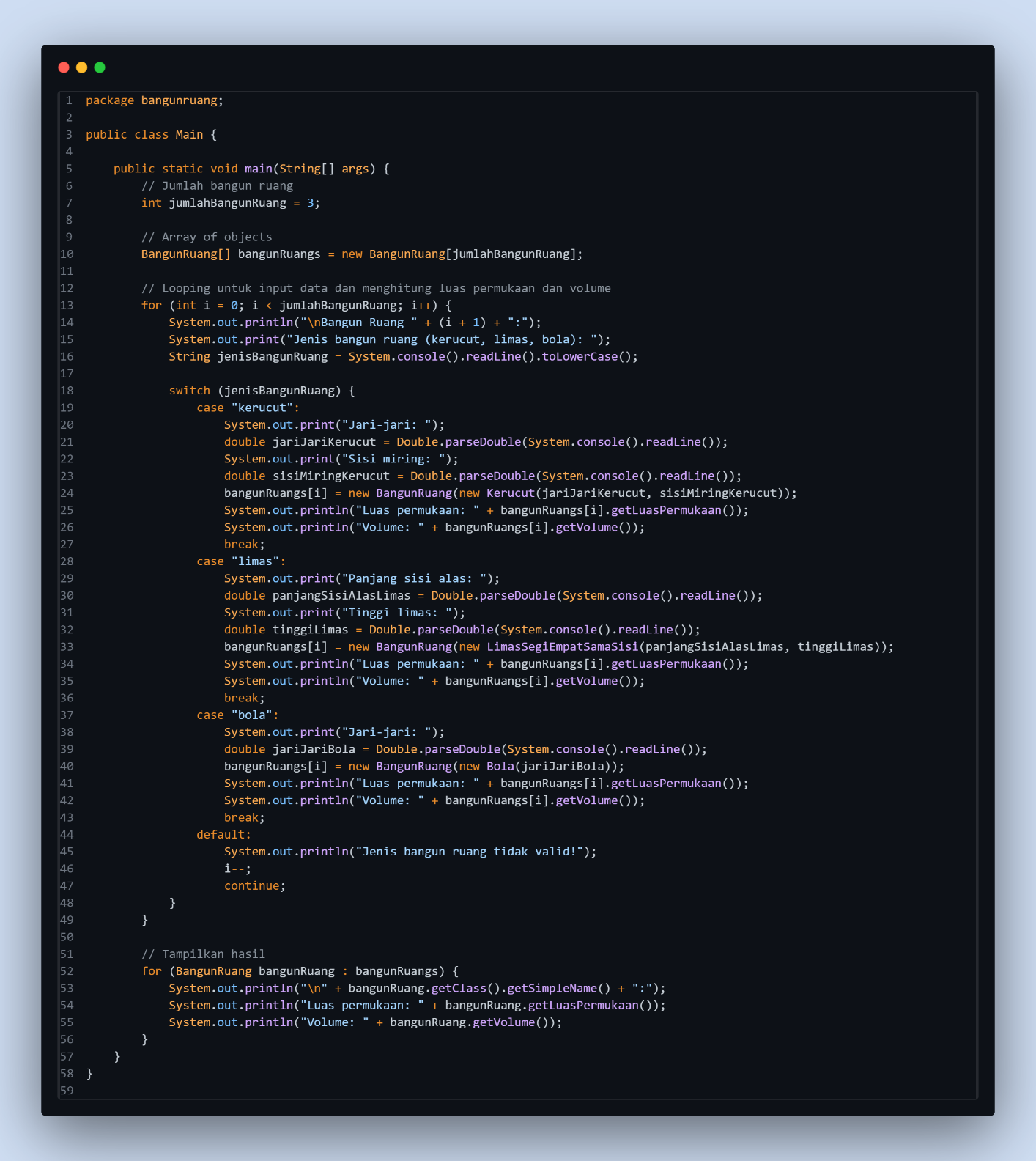
****

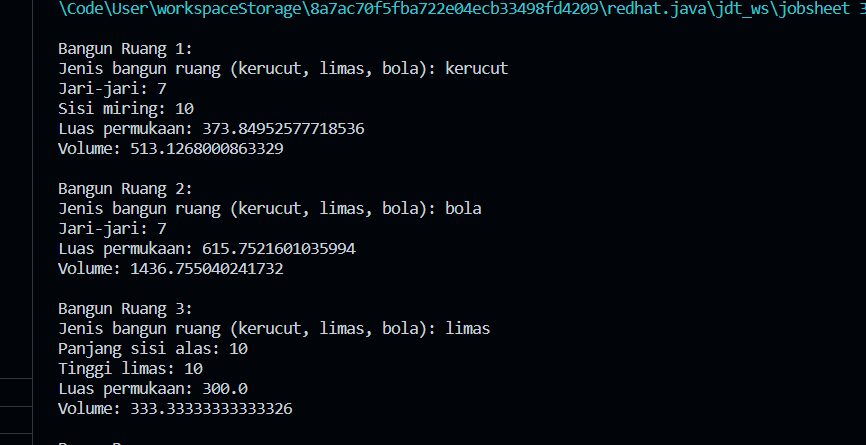
**Bola**

**BangunRuang**

****

**Main**



****

1. Sebuah kampus membutuhkan program untuk menampilkan informasi mahasiswa berupa nama, nim, jenis kelamin dan juga IPK mahasiswa. Program dapat menerima input semua informasi tersebut, kemudian menampilkanya kembali ke user. Implementasikan program tersebut jika dimisalkan terdapat 3 data mahasiswa yang tersedia. Contoh output program:

Masukkan data mahasiswa ke- 1 Masukkan nama :Rina

Masukkan nim :1234567 Masukkan jenis kelamin :P Masukkan IPK :3.5

Masukkan data mahasiswa ke- 2 Masukkan nama :Rio

Masukkan nim :7654321 Masukkan jenis kelamin :L Masukkan IPK :4.0

Masukkan data mahasiswa ke- 3 Masukkan nama :Reza

Masukkan nim :8765398 Masukkan jenis kelamin :L Masukkan IPK :3.8

Data Mahasiswa ke-1 nama : Rina

nim : 1234567

Jenis kelamin : P Nilai IPK : 3.5 Data Mahasiswa ke-2 nama : Rio

nim : 7654321

Jenis kelamin : L Nilai IPK : 4.0 Data Mahasiswa ke-3 nama : Reza

nim : 8765398

Jenis kelamin : L Nilai IPK : 3.8

1. Modify the resulting program at no.2, so that it could be used to calculate Average GPA, as well as to display student information that has biggest GPA! (use method to implement each process)



