Nama : Kelvianto Pratama Harum

NIM : 200210500016

MK : Pemrograman Lanjut

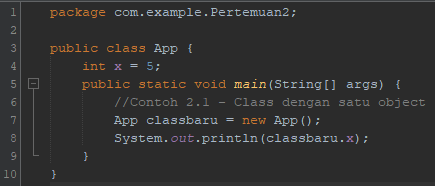
Dosen : Muhammad Fajar B, S.Pd., M.Cs.

**PERTEMUAN II**

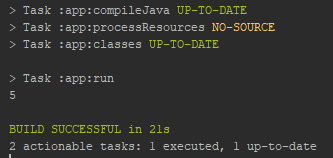
**CLASS DAN OBJECT**

1. **Contoh 2.1 – Class dengan satu object**

Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **App.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **App**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **10**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **App**.

Pada baris **4**, dideklarasikan **member variable** dengan nama **x**, tipe data **integer** (**signed, 4 bytes/32 bit**), dan diinitialisasi dengan nilai **5**.

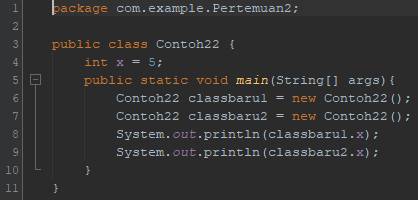
Pada baris **5**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **5** hingga baris **9**.

Pada baris **7**, dibuat sebuah object dari class **App** dengan nama **classbaru** tanpa argumen konstruktor.

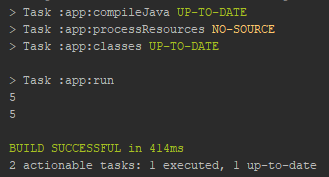
Pada baris **8**, digunakan untuk menampilkan nilai **member variable** dengan nama **x** dari object **classbaru** (yang merupakan object dari class **App**). Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

1. **Contoh 2.2 – Class dengan lebih dari satu object**

Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Contoh22.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **Contoh22**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **11**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **App**.

Pada baris **4**, dideklarasikan **member variable** dengan nama **x**, tipe data **integer** (**signed, 4 bytes/32 bit**), dan diinitialisasi dengan nilai **5**.

Pada baris **5**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **5** hingga baris **10**.

Pada baris **7**, dibuat sebuah object dari class **Contoh22** dengan nama **classbaru1** tanpa argumen konstruktor.

Pada baris **8**, dibuat sebuah object dari class **Contoh22** dengan nama **classbaru2** tanpa argumen konstruktor.

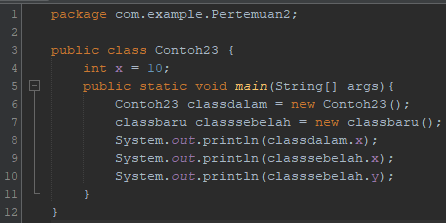
Pada baris **9**, digunakan untuk menampilkan nilai **member variable** dengan nama **x** dari object **classbaru1** (yang merupakan object dari class **Contoh22**). Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **10**, digunakan untuk menampilkan nilai **member variable** dengan nama **x** dari object **classbaru2** (yang merupakan object dari class **Contoh22**). Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

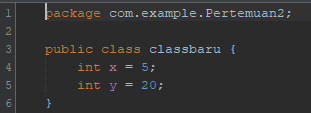
1. **Contoh 2.3 – Multiple class dan multiple object**

Source code:

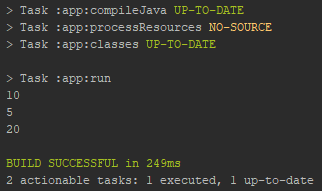
Contoh23.java



classbaru.java



Output:



Penjelasan:

Contoh23.java

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Contoh23.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **Contoh23**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **12**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Contoh23**.

Pada baris **4**, dideklarasikan **member variable** dengan nama **x**, tipe data **integer** (**signed, 4 bytes/32 bit**), dan diinitialisasi dengan nilai **10**.

Pada baris **5**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **5** hingga baris **11**.

Pada baris **6**, dibuat sebuah object dari class **Contoh23** dengan nama **classdalam** tanpa argumen konstruktor.

Pada baris **7**, dibuat sebuah object dari class **classbaru** (dari file **classbaru.java**) dengan nama **classluar** tanpa argumen konstruktor.

Pada baris **8**, digunakan untuk menampilkan nilai **member variable** dengan nama **x** dari object **classdalam** (yang merupakan object dari class **Contoh23**). Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **9**, digunakan untuk menampilkan nilai **member variable** dengan nama **x** dari object **classluar** (yang merupakan object dari class **classbaru** pada file **classbaru.java**). Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **10**, digunakan untuk menampilkan nilai **member variable** dengan nama **y** dari object **classluar** (yang merupakan object dari class **classbaru** pada file **classbaru.java**). Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

classbaru.java

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **classbaru.java** tergabung pada package **com.example.Pertemuan2**

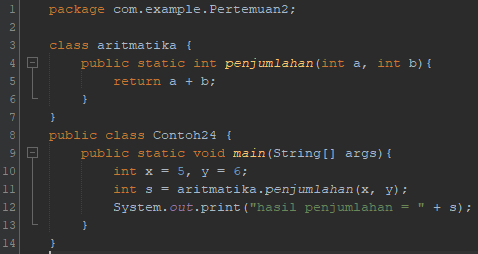
Pada baris **3**, dideklarasikan Class bernama **classbaru** dengan access modifier **public** yang berarti class ini dapat diakses dari file manapun dalam **package** yang sama, class ini mencakup semua kode yang ada pada baris **3** hingga baris **6**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **classbaru**.

Pada baris **4**, dideklarasikan **member variable** dengan nama **x**, tipe data **integer** (**signed, 4 bytes/32 bit**), dan diinitialisasi dengan nilai **5**.

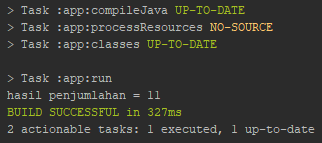
Pada baris **5**, dideklarasikan **member variable** dengan nama **y**, tipe data **integer** (**signed, 4 bytes/32 bit**), dan diinitialisasi dengan nilai **20**.

1. **Contoh 2.4 – Static method**

Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Contoh24.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **aritmatika**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **7**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **aritmatika**.

Pada baris **4**, dideklarasikan method static bernama **penjumlahan** dengan tipe data balik **integer (signed, 4 bytes/32 bit)**, parameter variabel **a** bertipe data **integer (signed, 4 bytes/32 bit)** dan variabel **b** bertipe data **integer (signed, 4 bytes/32 bit)**. Method ini memiliki scope dari baris **4** hingga baris **6**.

Pada baris **5**, dideklarasikan nilai balik dari method **penjumlahan** yaitu dengan menjumlahkan argumen dari variabel **a** dan **b**.

Pada baris **8**, dideklarasikan Class bernama **Contoh24**, class ini mencakup semua kode yang ada pada baris **8** hingga baris **14**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Contoh24**.

Pada baris **9**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **9** hingga baris **13**.

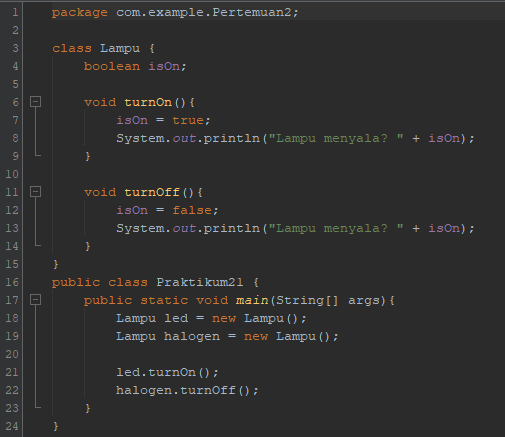
Pada baris **10**, dideklarasikan variabel bertipe data **integer (signed, 4 bytes/32 bit)** dengan nama **x** dan **y**. Variabel **x** bernilai **5**, dan variabel **y** bernilai **6**.

Pada baris **11**, dideklarasikan variabel bertipe data **integer (signed, 4 bytes/32 bit)** dengan nama **s** dan nilai dari nilai balik method **penjumlahan** pada class **aritmatika**. Pemanggilan method ini dapat dilakukan langsung tanpa harus membuat object **(instance)** terlebih dahulu karena method ini telah di-load pada saat class pertama kali dijalankan **(run)**.

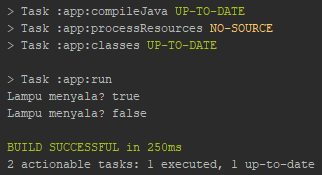
Pada baris **12**, digunakan untuk menampilkan string “hasil penjumlahan = “ diikuti dengan nilai variabel **s**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **print()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

1. **Praktikum 2.1**

Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Praktikum21.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **Lampu**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **15**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Lampu**.

Pada baris **4**, dideklarasikan **member variable** dengan tipe data **boolean (1 bit)** dan nama **isOn**.

Pada baris **6**, dideklarasikan method tanpa nilai balik **(void)** dengan nama **turnOn**, dalam method ini tidak dideklarasikan parameter pemanggilan **(call parameter)**. Scope method ini mulai dari baris **6** hingga baris **9**.

Pada baris **7**, diberikan nilai pada **member variable isOn**, dengan nilai **true**.

Pada baris **8**, digunakan untuk menampilkan string “Lampu menyala? “ diikuti dengan nilai variabel **isOn**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **11**, dideklarasikan method tanpa nilai balik **(void)** dengan nama **turnOff**, dalam method ini tidak dideklarasikan parameter pemanggilan **(call parameter)**. Scope method ini mulai dari baris **11** hingga baris **14**.

Pada baris **12**, diberikan nilai pada **member variable isOn**, dengan nilai **false**.

Pada baris **13**, digunakan untuk menampilkan string “Lampu menyala? “ diikuti dengan nilai variabel **isOn**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **16**, dideklarasikan Class bernama **Praktikum21**, class ini mencakup semua kode yang ada pada baris **16** hingga baris **24**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Praktikum21**.

Pada baris **17**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **17** hingga baris **23**.

Pada baris **18**, dideklarasikan sebuah **object** dari class **Lampu** dengan nama **led** dan tidak menggunakan argumen konstruktor.

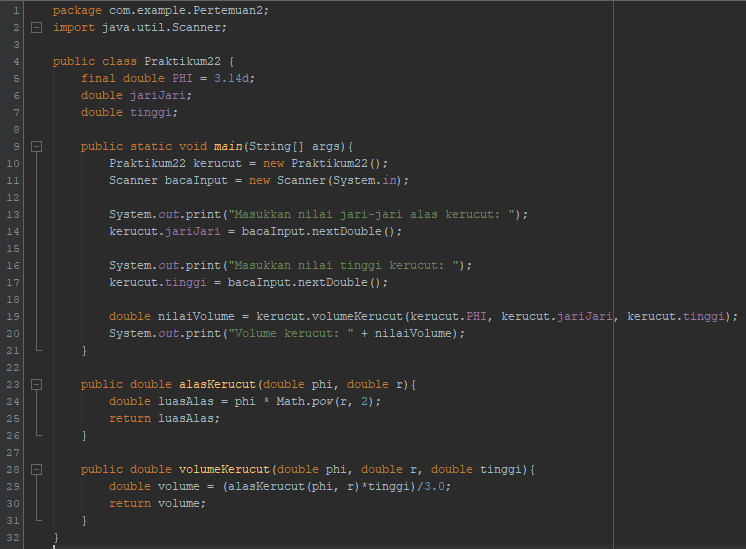
Pada baris **19**, dideklarasikan sebuah **object** dari class **Lampu** dengan nama **halogen** dan tidak menggunakan argumen konstruktor.

Pada baris **21**, dilakukan pemanggilan method **(method call)** **turnOn** dari object **led** (yang merupakan object dari class **Lampu**.

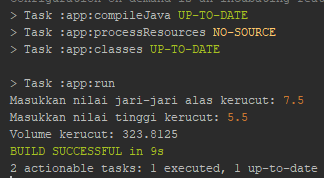
Pada baris **22**, dilakukan pemanggilan method **(method call)** **turnOff** dari object **halogen** (yang merupakan object dari class **Lampu**.

1. **Praktikum 2.2**

Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Praktikum22.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **2**, dilakukan import dari class bawaan java yaitu **java.util.Scanner**, class ini digunakan untuk menangkap user-input dan menyimpannya ke dalam variabel.

Pada baris **4**, dideklarasikan Class bernama **Praktikum22**, class ini mencakup semua kode yang ada pada baris **4** hingga baris **32**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Praktikum22**.

Pada baris **5**, dideklarasikan sebuah **member konstanta** (menggunakan keyword **final**) dengan tipe data **double (signed, 8 bytes/64 bit)**, nama **PHI** dan nilai **3.14d**. Huruf “**d**” menandakan bahwa nilai tersebut harus diperlakukan seperti data **double (64 bit)**.

Pada baris **6**, dideklarasikan sebuah **member variable** dengan tipe data **double (signed, 8 bytes/64 bit)** dan nama **jariJari**.

Pada baris **7**, dideklarasikan sebuah **member variabel** dengan tipe data **double (signed, 8 bytes/64 bit)** dan nama **tinggi**.

Pada baris **9**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **9** hingga baris **21**.

Pada baris **10**, dideklarasikan object bernama **kerucut** dari class **Praktikum22**.

Pada baris **11**, dideklarasikan object **bacaInput** dengan class **Scanner** dan parameter **System.in**. Baris ini mendeklarasikan object **bacaInput**, yang akan digunakan untuk menangkap **console user-input** (data yang masuk melalui **System.in**).

Pada baris **13**, digunakan untuk menampilkan string **“Masukkan nilai jari-jari alas kerucut: “**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **print()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **14**, field **jariJari** pada object **kerucut** didefinisikan dengan menangkap **console user-input** yang dikonversi (**casting**) menjadi nilai **double**.

Pada baris **16**, digunakan untuk menampilkan string **“Masukkan nilai tinggi kerucut: “**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **print()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **17**, field **tinggi** pada object **kerucut** didefinisikan dengan menangkap **console user-input** yang dikonversi (**casting**) menjadi nilai **double**.

Pada baris **19**, dideklarasikan variabel **nilaiVolume** dengan tipe data **double (signed, 8 bytes/64 bit)** dengan nilai dari nilai balik pemanggilan method **volumeKerucut** pada object **kerucut**, dengan argumen **field PHI** pada object **kerucut**, **field jariJari** pada object **kerucut, field tinggi** pada object **kerucut.**

Pada baris **20**, digunakan untuk menampilkan string **“Volume kerucut: “** diikuti dengan nilai variabel **nilaiVolume**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **print()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **23**, dideklarasikan method bernama **alasKerucut** dengan tipe data balik **double (signed, 8 bytes/64 bit)** dan parameter pemanggilan variabel **phi** dan **r**, yang semuanya adalah tipe data **double (signed, 8 bytes/64 bit)**. Scope method ini adalah dari baris **23** hingga baris **26**.

Pada baris **24**, dideklarasikan variabel bernama **luasAlas** dengan tipe data **double**, nilai variabel ini didefinisikan dari perhitungan **phi \*** **Math.pow(r,2)**.

Pada baris **25**, didefinisikan nilai balik dari method ini dari variabel **luasAlas**.

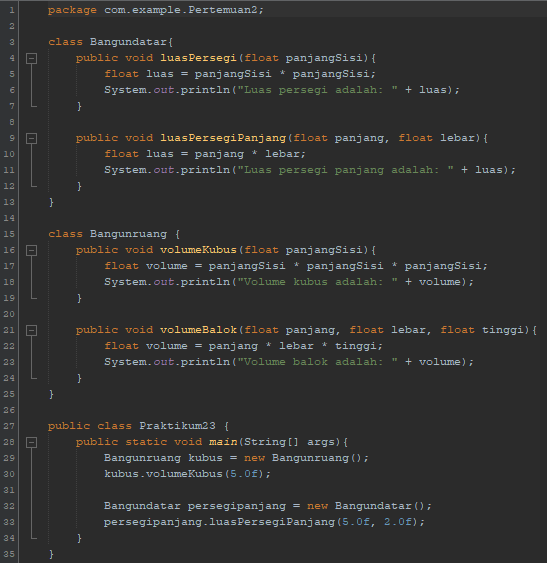
Pada baris **28**, dideklarasikan method bernama **volumeKerucut** dengan tipe data balik **double (signed, 8 bytes/64 bit)** dan parameter pemanggilan variabel **phi**, **r**, dan **tinggi**, yang semuanya adalah tipe data **double (signed, 8 bytes/64 bit)**. Scope method ini adalah dari baris **28** hingga baris **31**.

Pada baris **29**, dideklarasikan variabel bernama **luasAlas** dengan tipe data **double**, nilai variabel ini didefinisikan dari perhitungan nilai balik pemanggilan **alasKerucut** dengan argumen **phi** dan **r** dikali **variabel tinggi** lalu semua dibagi **3**.

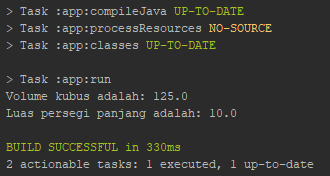
Pada baris **30**, didefinisikan nilai balik dari method ini dari variabel **volume**.

1. **Praktikum 2.3**

Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Praktikum23.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **Bangundatar**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **13**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Bangundatar**.

Pada baris **4**, dideklarasikan method tanpa nilai balik **(void)** dengan nama **luasPersegi**, dalam method ini parameter yang digunakan adalah variabel **panjangSisi**. Scope method ini mulai dari baris **4** hingga baris **7**.

Pada baris **5**, dideklarasikan variabel **float** dengan nama **luas** dengan nilai hasil perkalian variabel **panjangSisi** dan **panjangSisi**.

Pada baris **6**, digunakan untuk menampilkan string **“Luas persegi adalah: “** diikuti dengan nilai variabel **luas**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **9**, dideklarasikan method tanpa nilai balik **(void)** dengan nama **luasPersegiPanjang**, dalam method ini parameter yang digunakan adalah variabel **panjang** dan **lebar**. Scope method ini mulai dari baris **9** hingga baris **12**.

Pada baris **10**, dideklarasikan variabel **float** dengan nama **luas** dengan nilai hasil perkalian variabel **panjang** dan **lebar**.

Pada baris **11**, digunakan untuk menampilkan string **“Luas persegi panjang adalah: “** diikuti dengan nilai variabel **luas**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **15**, dideklarasikan Class bernama **Bangunruang**, class ini mencakup semua kode yang ada pada baris **15** hingga baris **25**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Bangunruang**.

Pada baris **16**, dideklarasikan method tanpa nilai balik **(void)** dengan nama **volumeKubus**, dalam method ini parameter yang digunakan adalah variabel **panjangSisi**. Scope method ini mulai dari baris **16** hingga baris **19**.

Pada baris **17**, dideklarasikan variabel **float** dengan nama **volume** dengan nilai hasil perkalian variabel **panjangSisi** sebanyak 3 kali.

Pada baris **18**, digunakan untuk menampilkan string **“Volume kubus adalah: “** diikuti dengan nilai variabel **luas**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**.

Pada baris **21**, dideklarasikan method tanpa nilai balik **(void)** dengan nama **volumeBalok**, dalam method ini parameter yang digunakan adalah variabel **panjang**, **lebar**, dan **tinggi**. Scope method ini mulai dari baris **21** hingga baris **24**.

Pada baris **22**, dideklarasikan variabel **float** dengan nama **volume** dengan nilai hasil perkalian variabel parameter **panjang, lebar** dan **tinggi**.

Pada baris **23**, digunakan untuk menampilkan string **“Volume balok adalah: “** diikuti dengan nilai variabel **luas**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **27**, dideklarasikan Class bernama **Praktikum23**, class ini mencakup semua kode yang ada pada baris **27** hingga baris **35**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Praktikum23**.

Pada baris **28**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **28** hingga baris **34**.

Pada baris **29**, dideklarasikan object bernama **kubus** dari class **Bangunruang** tanpa argumen konstruktor.

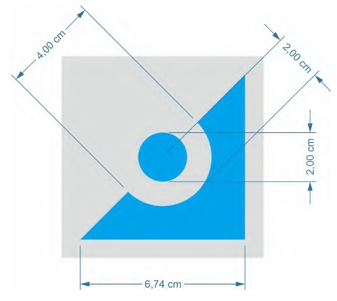
Pada baris **30**, dilakukan pemanggilan method **volumeKubus** dengan argumen nilai **5.0f** pada object **kubus**.

Pada baris **32**, dideklarasikan object bernama **persegipanjang** dari class **Bangundatar** tanpa argumen konstruktor.

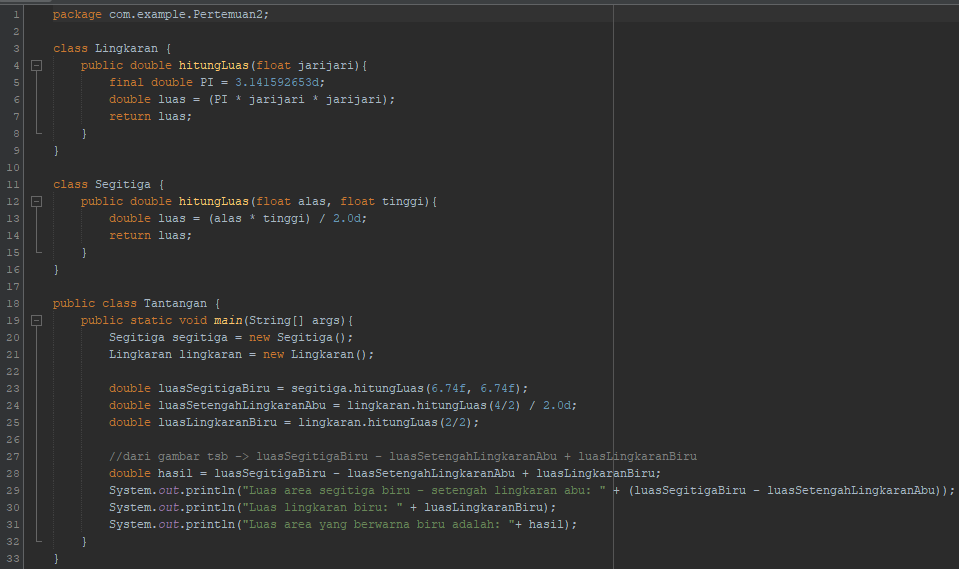
Pada baris **33**, dilakukan pemanggilan method **luasPersegiPanjang** dengan argumen nilai **5.0f** dan nilai **2.0f** pada object **persegipanjang**.

1. **Soal Tantangan**

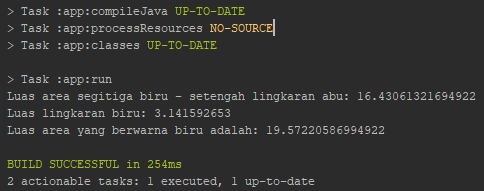
Perhatikan bangun datar berikut ini. Buatlah sebuah program untuk mengetahui berapa luas area yang berwarna biru. Informasi terkait ukuran/dimensi pembantu dapat dilihat pada gambar. Program yang dibuat wajib menerapkan konsep class, object dan method.



Source code:



Output:



Penjelasan:

Pada baris **1**, dideklarasikan package dimana file source ini bergabung pada contoh ini file **Praktikum23.java** tergabung pada package **com.example.Pertemuan2**

Pada baris **3**, dideklarasikan Class bernama **Lingkaran**, class ini mencakup semua kode yang ada pada baris **3** hingga baris **9**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Lingkaran**.

Pada baris **4**, dideklarasikan method dengan tipe data nilai balik **double** dengan nama **hitungLuas**, dalam method ini parameter yang digunakan adalah variabel **jariJari** dengan tipe data **float**. Scope method ini mulai dari baris **4** hingga baris **8**.

Pada baris **5**, dideklarasikan **konstanta** tipe data **double** dengan nama **PI** dengan nilai **3.141592653d**.

Pada baris **6**, dideklarasikan variabel **luas** tipe data **double** dengan nilai hasil perkalian **PI** dengan variabel **jariJari** dan variabel **jariJari**.

Pada baris **7**, dideklarasikan nilai balik dari method ini adalah nilai variabel **luas**.

Pada baris **11**, dideklarasikan Class bernama **Segitiga**, class ini mencakup semua kode yang ada pada baris **11** hingga baris **16**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Segitiga**.

Pada baris **12**, dideklarasikan method dengan tipe data nilai balik **double** dengan nama **hitungLuas**, dalam method ini parameter yang digunakan adalah variabel **alas** dan variabel **tinggi** dengan tipe data **float**. Scope method ini mulai dari baris **12** hingga baris **15**.

Pada baris **13**, dideklarasikan variabel **luas** tipe data **double** dengan nilai hasil perkalian variabel **alas** dan **tinggi** lalu dibagi dengan nilai **2.0d**.

Pada baris **14**, dideklarasikan nilai balik dari method ini adalah nilai variabel **luas**.

Pada baris **18**, dideklarasikan Class bernama **Tantangan**, class ini mencakup semua kode yang ada pada baris **18** hingga baris **33**, sehingga semua kode baik method dan identifiers dalam blok ini adalah scope dari class **Tantangan**.

Pada baris **19**, dideklarasikan method **main** dengan parameter String args[], method ini adalah method yang akan pertama kali dieksekusi pada saat runtime program berjalan. Scope method ini adalah baris **19** hingga baris **32**.

Pada baris **20**, dideklarasikan object bernama **segitiga** dari class **Segitiga** tanpa argumen konstruktor.

Pada baris **21**, dideklarasikan object bernama **lingkaran** dari class **Lingkaran** tanpa argumen konstruktor.

Pada baris **23**, dideklarasikan variabel **luasSegitigaBiru** dengan tipe data **double** dengan nilai hasil dari pemanggilan method **hitungLuas** dari object **segitiga** dengan argumen nilai **6.74f** dan **6.74f**.

Pada baris **24**, dideklarasikan variabel **luasSetengahLingkaranAbu** dengan tipe data **double** dengan nilai hasil dari pemanggilan method **hitungLuas** dari object **lingkaran** dengan argumen nilai **4** dibagi **2** lalu semuanya dibagi dengan **2.0d**.

Pada baris **25**, dideklarasikan variabel **luasLingkaranBiru** dengan tipe data **double** dengan nilai hasil dari pemanggilan method **hitungLuas** dari object **lingkaran** dengan argumen nilai **2** dibagi **2**.

Pada baris **28**, dideklarasikan variabel **hasil** dengan tipe data **double** yang merupakan hasil operasi matematika **luasSegitigaBiru – luasSetengahLingkaranAbu + luasLingkaranBiru**.

Pada baris **29**, digunakan untuk menampilkan string **“Luas area segitiga biru – setengah lingkaran abu: “** diikuti dengan hasil operasi **luasSegitigaBiru - luasSetengahLingkaranAbu**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **30**, digunakan untuk menampilkan string **“Luas lingkaran biru: “** diikuti dengan nilai variabel **luasLingkaranBiru**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.

Pada baris **31**, digunakan untuk menampilkan string **“Luas area yang berwarna biru adalah: “**, diikuti dengan nilai variabel **hasil**. Output ini dihasilkan menggunakan method bawaan dari Java yang berada pada class **System**, member **out**, method **println()**. Method ini pada dasarnya akan menampilkan apapun yang diberikan pada calling argument ke runtime console.