**LAPORAN PENGERJAAN TUGAS PERTEMUAN 4**

**TEKNIK PEMOGRAMAN PRAKTIKUM**

Disusun untuk memenuhi salah satu tugas pertemuan keempat mata kuliah

Teknik Pemograman



**Oleh:**

Fadilah Akbar (231524041)

**Kelas :**

Teknik Informatika D4 – 1B

**PROGRAM STUDI SARJANA TERAPAN TEKNIK INFORMATIKA**

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2024**

DAFTAR ISI

[Exercise 1 : The Circle and Cylinder Classes 4](#_Toc160049818)

[Task 1.1 4](#_Toc160049819)

[Menambahkan Sebuah Variable Bernama Color Bertype String 4](#_Toc160049820)

[Constructor Circle dengan arguments double radius dan String color. 4](#_Toc160049821)

[Membuat Getter Dan Setter Untuk Attributes Color 4](#_Toc160049822)

[Task 1.2 5](#_Toc160049823)

[Task 1.3 5](#_Toc160049824)

[Sebelum 5](#_Toc160049825)

[Sesudah 5](#_Toc160049826)

[Exercise 2 : Superclass Shape and its Subclasses Circle, Rectangle and Square 6](#_Toc160049827)

[Task 2.1 6](#_Toc160049828)

[Source Code Class Shape 7](#_Toc160049829)

[Source Code Class Circle 7](#_Toc160049830)

[Source Code Class Rectangle 8](#_Toc160049831)

[Source Code Class Square 8](#_Toc160049832)

[Output dari setiap class 9](#_Toc160049833)

[Output Shape 9](#_Toc160049834)

[Output Circle 9](#_Toc160049835)

[Output rectangle 9](#_Toc160049836)

[Output Square 9](#_Toc160049837)

[Exercise 3 : Multiple Inheritance 10](#_Toc160049838)

[Task 3.1 10](#_Toc160049839)

[ Class Sortable 10](#_Toc160049840)

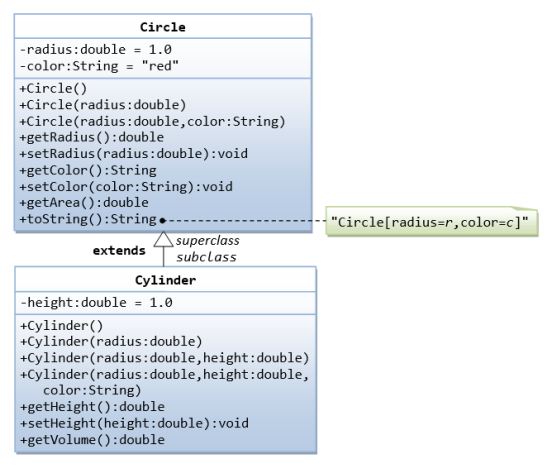
[ Class Employee 10](#_Toc160049841)

[ Class EmployeeTest 11](#_Toc160049842)

[ Class Manager 12](#_Toc160049843)

[ Class ManagerTest 13](#_Toc160049844)

# Exercise 1 : The Circle and Cylinder Classes



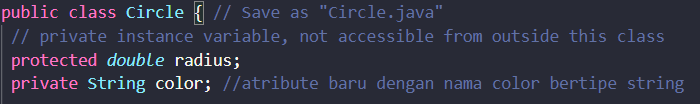
## Task 1.1

Pada task 1.1 ini, saya diberikan perintah untuk

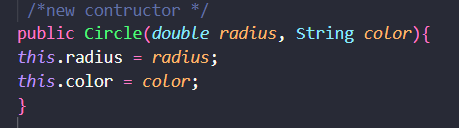
* Memodifikasi class Circle dengan menambahkan sebuah variable bernama color bertype string
* Membuat constructor Circle dengan arguments double radius dan String color.
* Membuat getter dan setter untuk attributes Color.

Berikut ini source code yang sudah saya modifikasi :

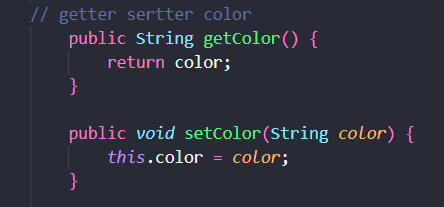
### Menambahkan Sebuah Variable Bernama Color Bertype String



### Constructor Circle dengan arguments double radius dan String color.



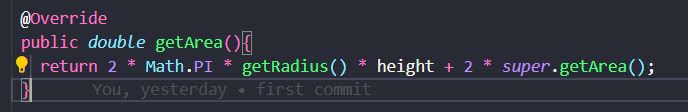
### Membuat Getter Dan Setter Untuk Attributes Color



## Task 1.2

Pada task 1.2 ini diberikan perintah untuk melakukan override kepada method getArea dalam subclass Cylinder yang dimana method ini di inherit dari superclass Circle.

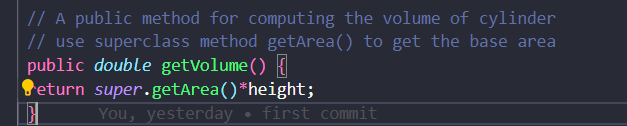
Jika method getArea dalam class Circle akan mengembalikan nilai dari luas lingkaran, maka pada method getArea dalam subclass Cylinder ini akan merubah behaviournya menjadi mengembalikan nilai “2\*phi\*radius\*tinggi+2\*luas-alas”.



Dari gambar di atas dalam pemanggilan ***getArea()*** yang dipanggil adalah dari class Circle sehingga menggunakan kata kunci ***“super”*** . bahwa return ini memanggil juga getter radius karena akan dipakai.

Pada proses override methode ***getArea()***, terdapat methode ***getVolume()*** tidak bekerja sama. Dikarenakan untuk mendapatkan sebuah volume tabung kita harus mengalikan luas alat berupa lingkaran dengan tinggi dari tabung tersebut.

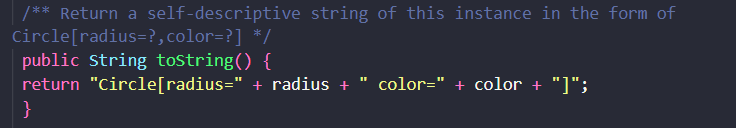
Untuk memperbaiki methode ini karena harus mengembalikan nilai luas lingkaran yang digunakan dalam method ***getVolume(),*** harus menambahkan kata kunci ***“super”*** sebelum ***getArea()*** untuk mendapatkan nilai luas alas.



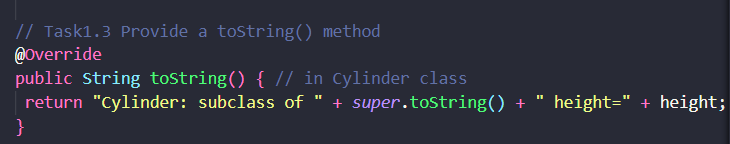
## Task 1.3

Diberikan perintah untuk melakukan override terhadap method toString yang sebelumnya ada pada superclass Circle. Berikut ini adalah source code sebelum dan sesudah dari method toString ini :

### Sebelum



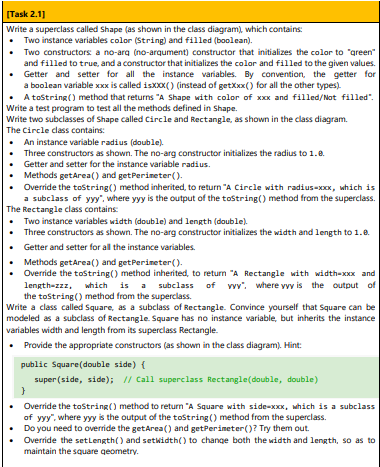
### Sesudah



# Exercise 2 : Superclass Shape and its Subclasses Circle, Rectangle and Square



## Task 2.1



pada task 2.1 ini diperintahkan membuat sebuah cource code dari class class ini dan isinya sesuai dengan spesifikasi dari setiap class seperti pada gambar di atas

### Source Code Class Shape



### Source Code Class Circle



### Source Code Class Rectangle



### Source Code Class Square



### Output dari setiap class

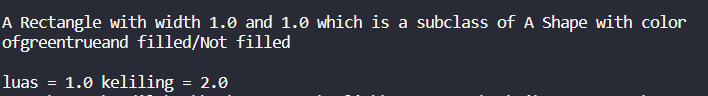
### Output Shape



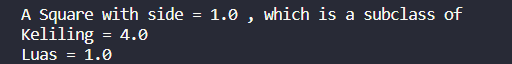
### Output Circle

A black screen with white text  Description automatically generated

### Output rectangle



### Output Square



# Exercise 3 : Multiple Inheritance

## Task 3.1

Pada task 3.1 ini, diperintahkan untuk menyalin 4 class berupa class Employee, EmployeeTest, Manager, dan ManagerTest, selain diperintahkan juga untuk membuat sebuah class abstract bernama Sortable yang akan menjadi superclass untuk class Employee.

Untuk lebih jelasnya, berikut di bawah adalah rincian setiap classnya :

### Class Sortable

Pada class sortable ini memiliki 2 buah method namun tidak memiliki atribute, method yang pertama dengan nama ***compare(Sortable b)*** yang berisikan parameter b bertipe nama class nya, method ini bersifat abstrak yang dimana dalam implementasi harus dibuat behavior nya di subclass. Method kedua dengan nama ***shell\_sort(Sortable[] a)*** method ini statis menerima sebuah array dari objek ***“Sortable”*** yang akan diurutkan menggunakan algoritma Shell sort.

Algoritma shell sort yang diimplemtasikan menggunakan sebuah gap(jarak) yang dibagi dua (dalam loop ‘gap /= 2’) sampai gap mencapai 0. Dalam setiap proses dilakukan pertukaran element element array yang berjarak satu sama lainya, serta dalam setiap pertukaran dilakukan perbandingan menggunakan method ***compare()*** yang di implementasikan oleh objek objek Sortable.

### Class Employee

Class ini merupakan turunan dari ***Class Sortable*** yang dimana harus menerapkan metode ***compare()*** sesuai dengan kebutuhan dalam pengurutan objek Employee, class ini beberapa atribut seperti ***name*** bertipe string, ***salary*** bertipe double, ***hireday*** bertipe integer, ***hiremonth*** bertipe integer, dan ***hireyear*** bertipe integer yang merepresentasikan data seorang karyawan. Selain itu terdapat konstruktor yang digunakan untuk menginisialisasi data karyawan.

Selain itu, terdapat beberapa method diantaranya, method ***print()*** untuk mencetak informasi data karyawan, method ***raiseSalary()*** untuk menaikan gaji karyawan berdasarkan persentase yang diberikan, selanjutnya method ***hireYear()*** untuk mendapatkan tahun perekrutan karyawan, dan method ***compare()*** yang merupakan implementasi dari class Sortable untuk membandingkan karyawan berdasarkan gaji karyawan.



### Class EmployeeTest

Pada class ini sesuai dengan namanya yaitu untuk menguji implementasi dari class Employee dan pengurutan ***shell\_sort()*** yang telah didefinisikan dalam class Sortable.

Inisialisasi terdapat tiga objek Employee dibuat dan disampai kedalam array ***‘staff’,*** setiap objek mewakili seorang karyawan yang berisikan data nama, gaji, dan tanggal perekrutan yang berbeda. Method ***‘shell\_sort()’*** dari class Sortable dipanggil untuk mengurutkan array staff yang berisikan objek objek Emlpoyee, objek oebjek ini akan diurutkan berdasarkan gaji mereka dari yang terbesar hingga yang terkecil sesuai dengan implementasi pada method ***compare()***

Setelah itu gaji karyawan dinaikan sebesar5% menggunakan method ***raiseSalary(),*** dan informasi karyawan akan dicetak menggunakan method ***print()*** yang menampilkan nama, gaji, dan tahun perekrutan karyawan setelah penaikan gaji. Berikut adalah source code dan output dari program EmployeeTest





### Class Manager

Pada class ini untuk merepresentasikan seorang manager dalam sebuah prusahaan, class ini merupakan turunan dari class Employee yang berari akan mewarasi sifat dari kelas tersebut, namun dapat memiliki perilahu tambahan yang khusus seperti mendapatkan bonus berdasarkan tahun layanan dan memiliki sekertaris

Pada programnya terdapat sebuah konstraktor yang digunakan untuk membuat objek Manager dengan parameter n bertipe string, s bertipe double, atribut d, m , dan y bertipe integer. Yang diteruskan kelas Employee menggunakan kata kunci **‘super’.**

Selanjutnya terdapat method ***raiseSalary(double byPercent)*** yang merupakan override dari class Emloyee, serta terdapat method ***getSecretaryName()*** untuk mendapatkan nama sekertaris, sekertaris ini direpsentasikan oleh atribut ‘***secretartName’.***



### Class ManagerTest

Class ini adalah sebuah class yang digunakan untuk menguji/ implementasi hasil dari kelas Employee dan Manager seperti halnya class EmployeTest. Berikut adalah sourcecode dan hasil programnya.



