**LAPORAN PRAKTIKUM**

**PEMROGRAMAN BERBASIS OBJEK**

# JOBSHEET 6

### INHERITANCE (PEWARISAN)

**Dosen Pembimbing:**

**Luqman Affandi, S.Kom., M.MSI.**



**Disusun oleh:**

**Cakra Wangsa M.A.W**

**2341720032**

**2F**

**D-4 Teknik Informatika**

**Soekarno Hatta No.9, Jatimulyo, Kec. Lowokwaru, Kota Malang, Jawa Timur 6**

**Phone: (0341) 404424,404425**

**E-email: Polinema.ac.id**

# JOBSHEET 6

### INHERITANCE (PEWARISAN)

1. **KOMPETENSI**
   1. Memahami konsep dasar inheritance atau pewarisan.
   2. Mampu memahami jenis-jenis inheritance.
   3. Mampu mengimplementasikan konsep inheritance pada Bahasa pemrograman Java

# PENDAHULUAN

**Inheritance** atau **pewarisan sifat** merupakan suatu cara untuk menurunkan suatu class yang lebih umum menjadi suatu class yang lebih spesifik. Inheritance adalah salah satu ciri utama suatu bahasa program yang berorientasi pada objek. Inti dari pewarisan adalah sifat reusable dari konsep object oriented. Setiap **subclass** akan “mewarisi” sifat dari **superclass** selama bersifat protected ataupun public.

Dalam inheritance terdapat dua istilah yang sering digunakan. Kelas yang menurunkan disebut kelas dasar (**base class**/**super class**), sedangkan kelas yang diturunkan disebut kelas turunan (**derived class**/**sub class**/**child class**) . Di dalam Java untuk mendeklarasikan suatu class sebagai subclass dilakukan dengan cara menambahkan kata kunci **extends** setelah deklarasi nama class, kemudian diikuti dengan nama parent class-­‐nya. Kata kunci extends tersebut memberitahu kompiler Java bahwa kita ingin melakukan perluasan class. Berikut adalah contoh deklarasi inheritance.

public class B extends A {

...

}

Contoh diatas memberitahukan kompiler Java bahwa kita ingin meng-­‐extend class A ke class B. Dengan kata lain, class B adalah subclass (class turunan) dari class A, sedangkan class A adalah parent class dari class B.

Karakteristik pada super class akan dimiliki juga oleh subclassnya. Terdapat 5 bentuk pewarisan: single inheritance, multilevel inheritance, hierarchical inheritance, hybrid inheritance, dan multiple inheritance.

1. Single Inheritance

Single inheritance adalah Suatu class yang hanya mempunyai satu parent class. Contoh:

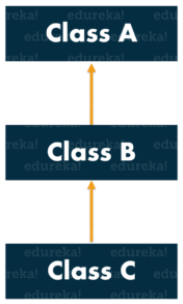


Gambar 1. Contoh Single Inheritance

Berdasarkan Gambar 1 dapat diketahui bahwa class B merupakan subclass yang mempunyai satu parent yaitu class A sehingga disebut single inheritance.

1. Multilevel Inheritance

Multilevel inheritance adalah Suatu subclass bisa menjadi superclass bagi class yang lain. Contoh:

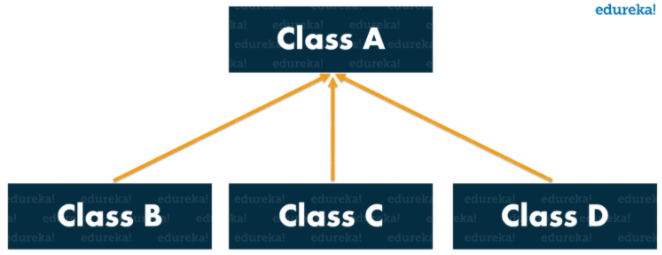


Gambar 2. Contoh Multilevel Inheritance

Berdasarkan Gambar 2 diatas dapat dilihat bahwa class B merupakan subclass dari class A, sehingga dalam hal ini class A adalah superclass dan class B adalah subclass. Kemudian class B yang awalmya merupakan subclass mempunyai subclass lagi yaitu class C sehingga class B menjadi superclass dari class C, begitu juga seterunya jika class C memilki subclass lagi.

1. Hierarchical Inheritance

Hierarchical inheritance merupakan contoh pewarisan yang terjadi ketika sebuah kelas memiliki lebih dari satu kelas turunan (*subclass*) atau dengan kata lain, lebih dari satu kelas turunan memiliki kelas induk yang sama. Konsep ini sesuai dengan contoh di dunia nyata, dimana satu orangtua bisa memiliki lebih dari satu anak. Contoh bentul hierarchical inheritance dapat dilihat pada Gambar 3.

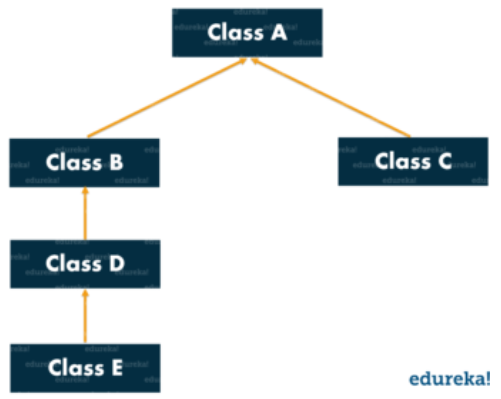


Gambar 3. Contoh Hierarchical Inheritance

Pada Gambar 3 dapat dilihat bahwa class A memiliki 3 subclass, yaitu class B, class C, dan class D. Dengan demikian, class B, class C, dan class D memiliki superclass yang sama, dan ada kemungkinan ketiga class cubclass tersebut memiliki atribut dan behaviour yang sama dari class A.

1. Hybrid Inheritance

Hybrid inheritance merupakan jenis pewarisan yang merupakan kombinasi dari dua atau lebih jenis pewarisan. Contoh hybrid inheritance dapat dilihat pada Gambar 4.

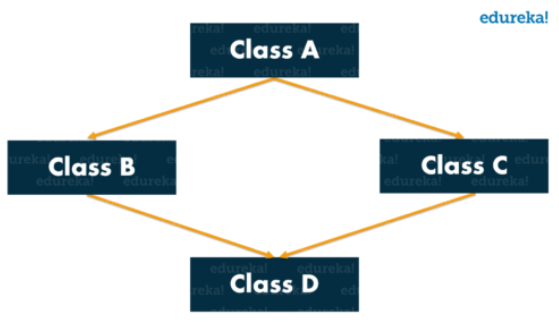


Gambar 4. Contoh Hybrid Inheritance

Pada Gambar 4 dapat dilihat bahwa bentuk pewarisan tersebut merupakan kombinasi dari hierarchical dan multilevel inheritance. Hierarchical inheritance ditunjukkan oleh Class A, class B, dan class C. Sedangkan multilevel inheritance ditunjukkan oleh class B, class D, dan class E.

Terdapat beberapa aturan pada pewarisan menggunakan Bahasa pemrograman Java, antara lain:

1. Multiple inheritance tidak dapat digunakan pada Bahasa pemrograman Java. Multiple inheritance mengacu pada proses dimana satu kelas turunan mencoba untuk memperluas (extend) lebih dari satu kelas induk. Contoh multiple inheritance dapat dilihat pada Gambar 5.

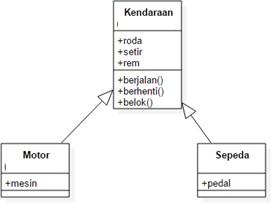


Gambar 5. Contoh multiple inheritance

Misal terdapat method show() pada kelas B dan C dengan fungsi yang berbeda. Kemudian kelas D meng-extend kelas B dan C. Ketika objek dari kelas D mencoba memanggil method show(), kompilator akan bingung method di kelas mana yang akan dieksekusi (dari kelas B atau C). Sehingga hal ini mengarah pada ambiguitas.

1. Cyclic inheritance tidak dapat digunakan di Java. Cyclic inheritance merupakan jenis inheritance dimana sebuah kelas memperluas (extend) dirinya sendiri. Jenis ini tidak diizinkan oleh Java karena tidak ada peluang untuk memperluas kelas Object.
2. Atribut dan method dengan access modifier Private tidak diwariskan.
3. Constructor tidak dapat diwariskan. Jika ingin menggunakan constructor kelas induk, harus menggunakan perintah super() pada constructor kelas anak.

Pada class diagram, pewarisan digambarkan dengan sebuah garis tegas, dengan segitiga di ujungnya. Class yang dekat pada segitiga merupakan superclass, sedangkan class yang jauh dari segitiga merupakan subclass. Untuk membentuk sebuah subclass, keyword “extends” digunakan (lihat contoh pada sesi “Implementasi Pewarisan”). Berikut ini adalah contoh class diagram dari pewarisan:

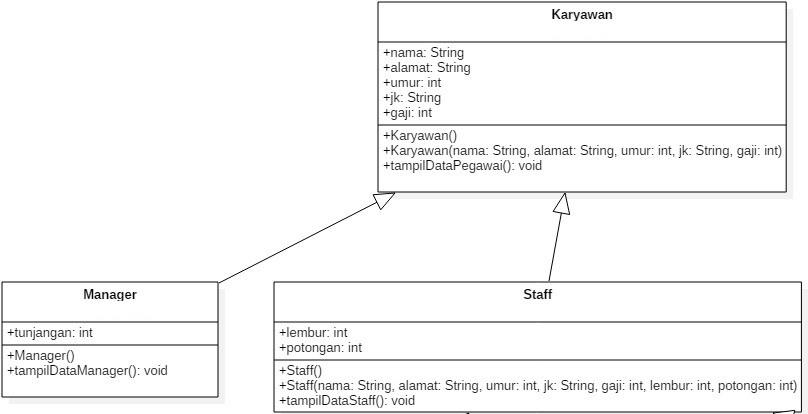


Gambar 6 Contoh class diagram dalam inheritance

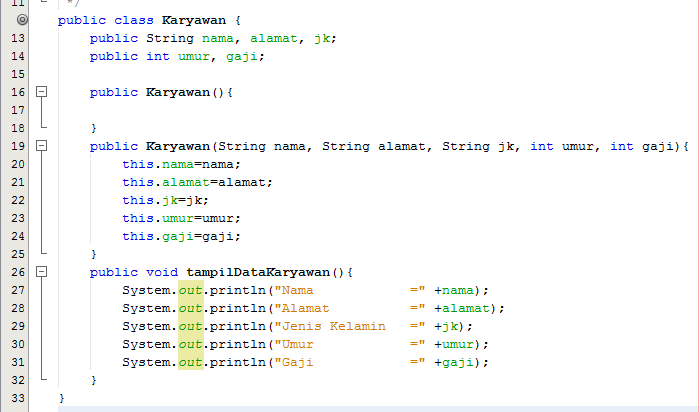
# PERCOBAAN 1

## TAHAPAN PERCOBAAN

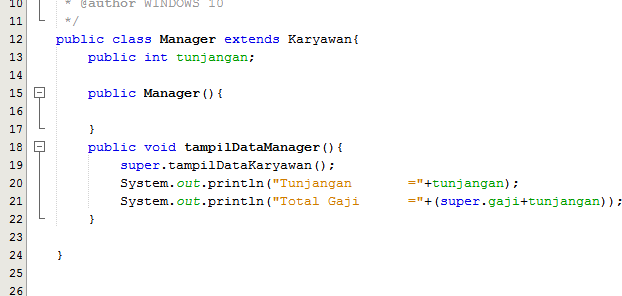
Perhatikan diagram class dibawah ini:



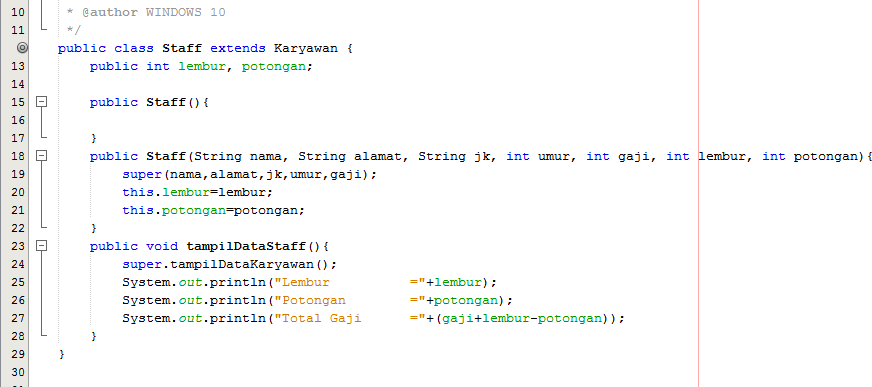
* + 1. Buatlah class Karyawan



* + 1. Buatlah class Manager

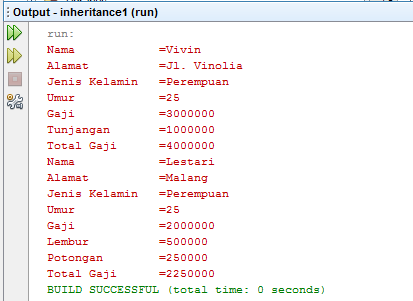


* + 1. Buatlah class Staff



* + 1. Buatlah class inheritance1

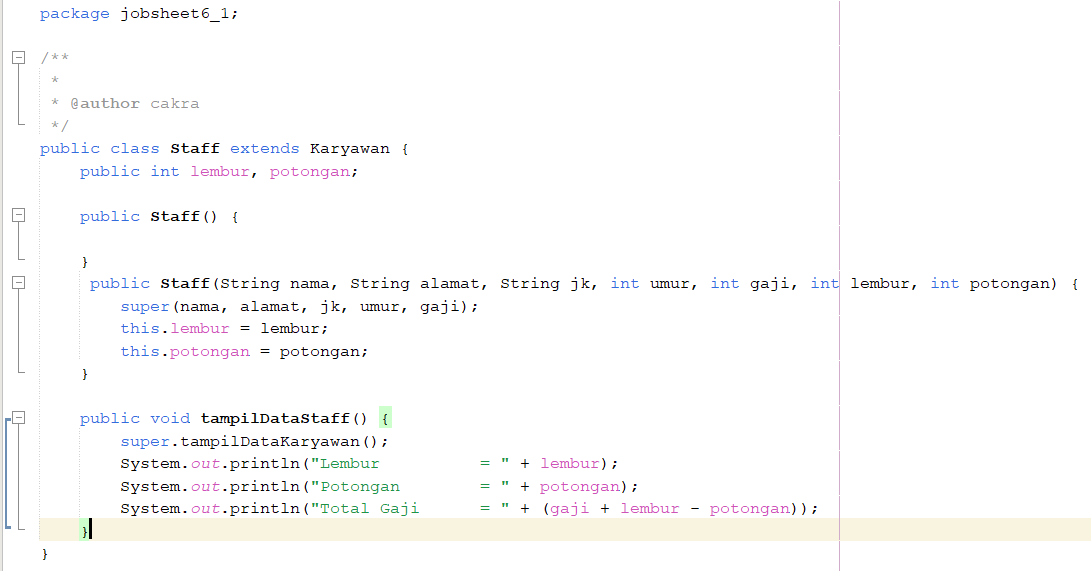


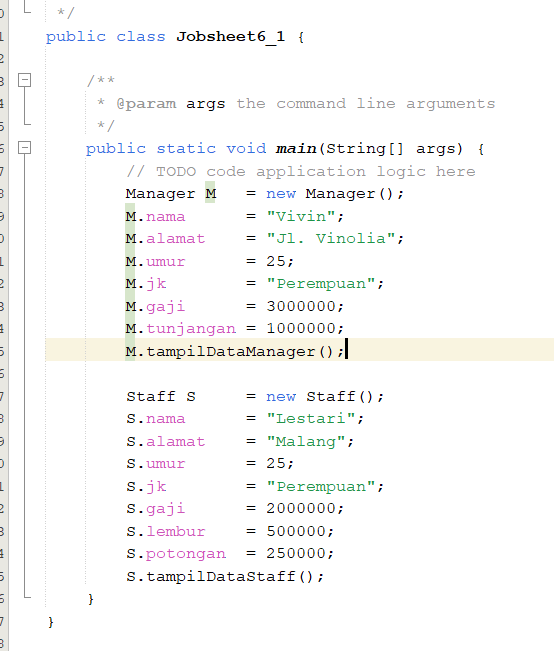
* + 1. Jalankan program, maka tampilanya adalah sebagai berikut:

Jawab:

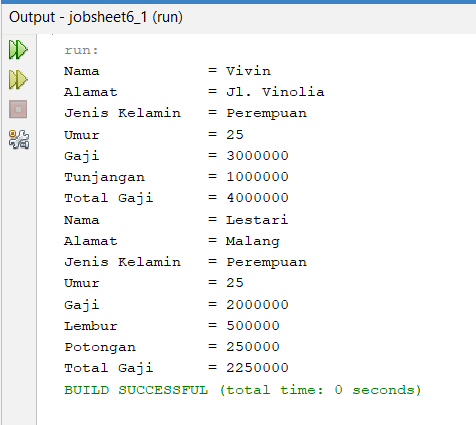








Output



## PERTANYAAN

* + 1. Sebutkan class mana yang termasuk super class dan sub class dari percobaan 1 diatas!

Jawab:

* superClass adalah Karyawan
* subClass adalah Manager dan Staff
  + 1. Kata kunci apakah yang digunakan untuk menurunkan suatu class ke class yang lain?

Jawab:

* Untuk menurunukan suatu class ke class yang lain menggunakan kata kunci extend

dengan contoh public class Manager extends Karyawan

* + 1. Perhatikan kode program pada class Manager, atribut apa saja yang dimiliki oleh class tersebut? Sebutkan atribut mana saja yang diwarisi dari class Karyawan!

Jawab:

* Pada class Manager memiliki atribut tunjangan
* Atribut yang diwarisi dari kelas karyawan adalah nama, alamat, jk, umur, gaji

* + 1. Jelaskan kata kunci super pada potongan program dibawah ini yang terdapat pada class Manager!

Jawab:

* Kata kunci **super** dalam class Manager digunakan untuk mengakses dan memanggil anggota (metode dan variabel) dari class induknya yaitu pada class (Karyawan)
  + 1. Program pada percobaan 1 diatas termasuk dalam jenis inheritance apa? Jelaskan alasannya!

Jawab:

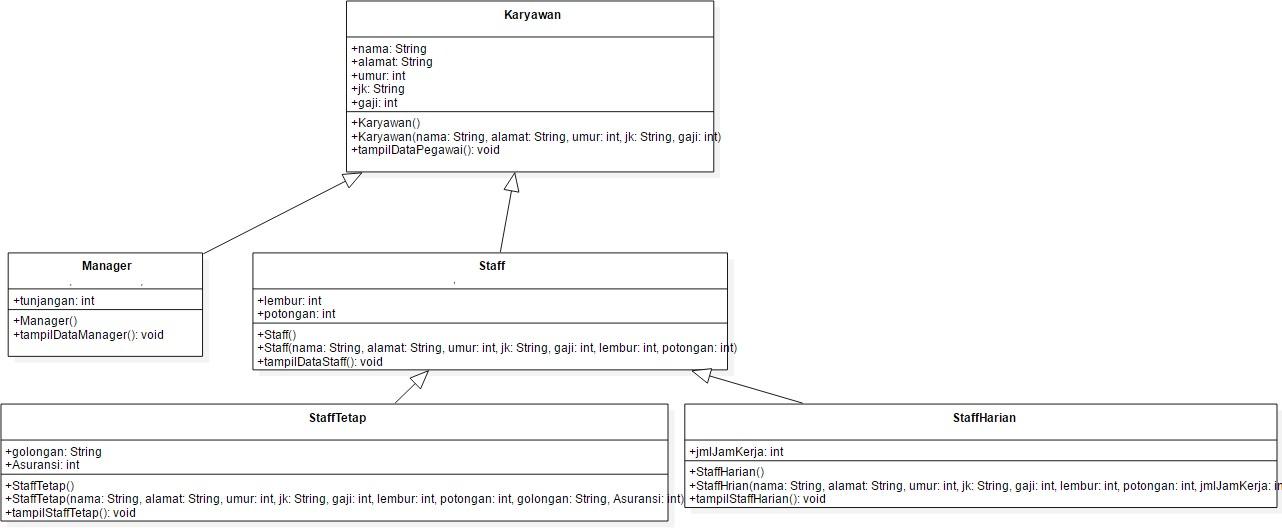
* Pada percobaan 1 diatas termasuk dalam jenis Hierarchical Inheritance dikarenakan

ada beberapa class seperti class Manager dan Staff yang mewarisi dari 1 class induk yang sama yaitu class Karyawan.

# PERCOBAAN 2

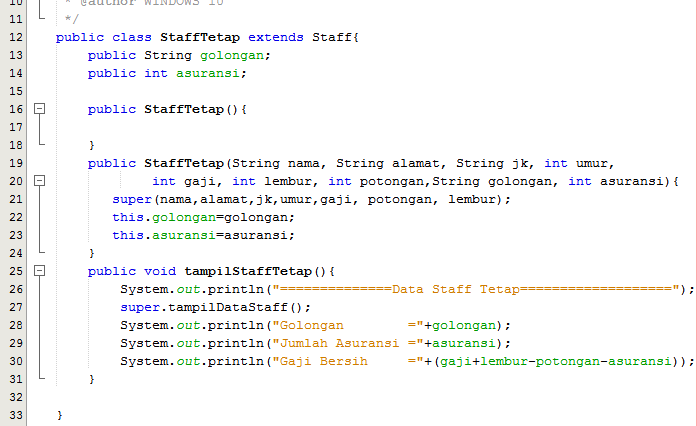
## TAHAPAN PERCOBAAN

Perhatikan digram class dibawah ini

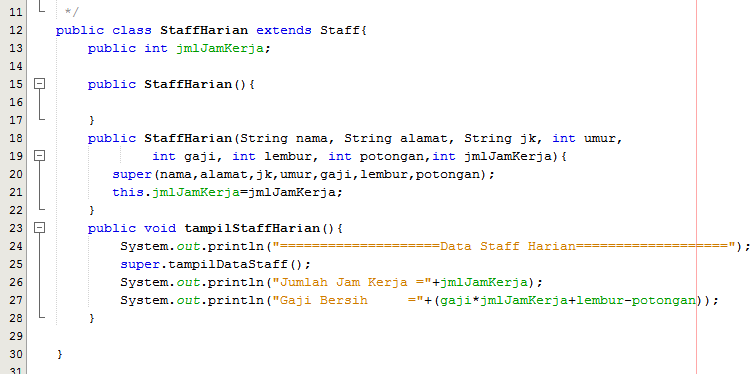


* + - Berdasarkan program yang sudah anda buat pada percobaan 1 sebelumnya tambahkan dua class yaitu class StaffTetap dan class StaffHarian. Kode Programnya adalah sebagai berikut

Class StaffTetap.java

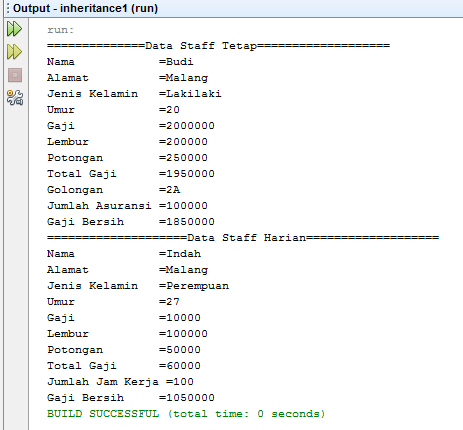


Class StaffHarian.java



* + - Setelah membuat dua class diatas kemudian edit class inheritance1.java menjadi sebagai berikut:

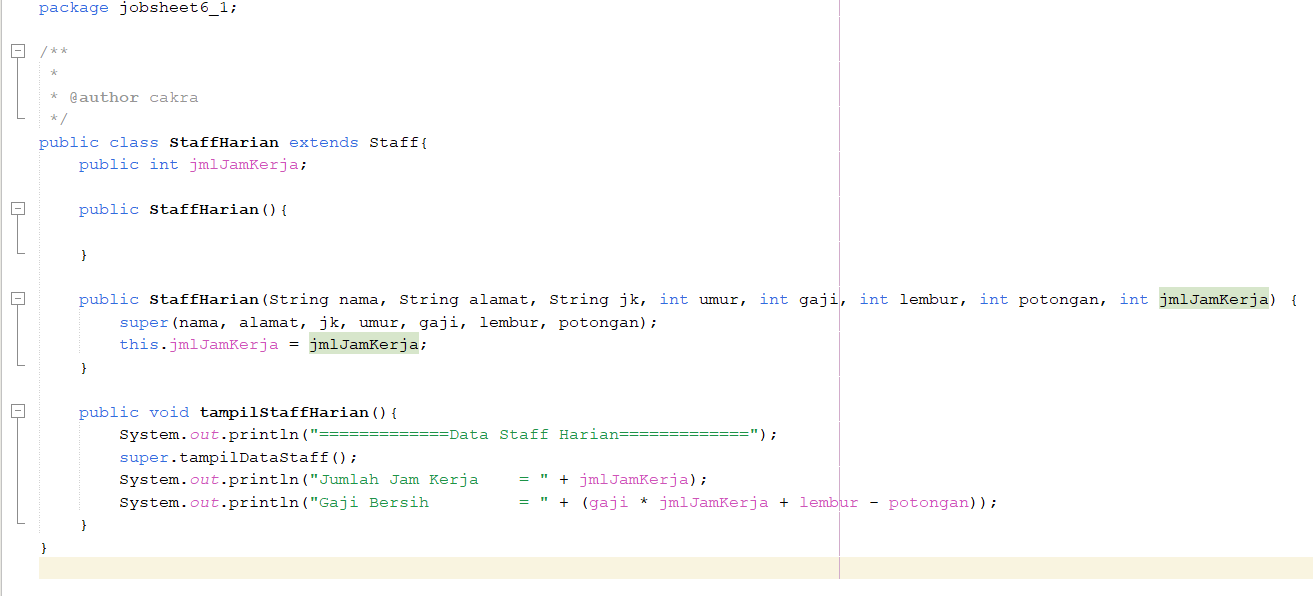


* + - Jalankan program maka tampilanny adalah sebagai berikut:

Jawab:

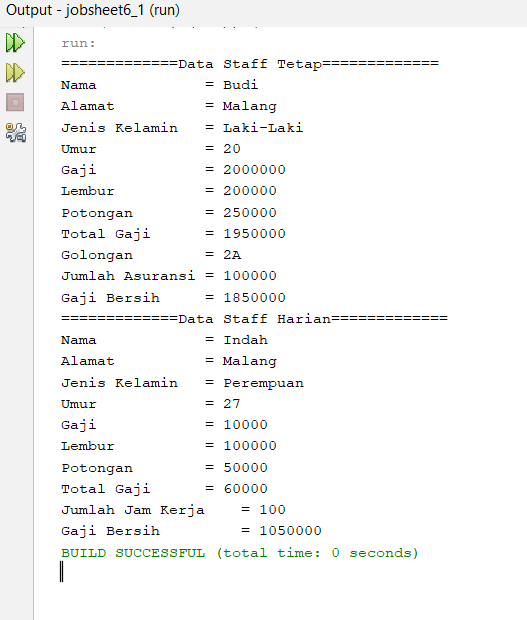
Kode program yang saya input







Output



## PERTANYAAN

* + - Berdasarkan class diatas manakah yang termasuk single inheritance dan mana yang termasuk multilevel inheritance?

Jawab:

* Pada single inheritance terdapat pada class Manager dikarenakan diwarisi dari struktur class Karyawan saja
* pada multilevel inheritance terdapat pada class StaffTetap dan StaffHarian dikarenakan diwarisi dari struktur class Karyawan-> Staff-> StaffTetap dan StaffHarian
  + - Perhatikan kode program class StaffTetap dan StaffHarian, atribut apa saja yang dimiliki oleh class tersebut? Sebutkan atribut mana saja yang diwarisi dari class Staff!

Jawab:

* Pada class StaffTetap mewarisi atribut dari class Staff yaitu nama, alamat, jk, umur, gaji, potongan, lembur
* Pada class StaffHarian mewarisi atribut dari class Staff yaitu nama, alamat, jk, umur, gaji, lembur, potongan
  + - Apakah fungsi potongan program berikut pada class StaffHarian



Jawab:

* Berfungsi untuk dapat memanggil konstruktor dari class induk yaitu (Staff) untuk dapat menginisialisasi atribut yang diwarisi
* Dan untuk dapat menghindari duplikasi kode untuk inisialisasi atribut yang ada pada class induk
  + - Apakah fungsi potongan program berikut pada class StaffHarian

Jawab:

* super.tampilDataStaff(); berfungsi untuk dapat memanggil metode pada tampilDataStaff() dari induk class yaitu class Staff
* dan untuk dapat menampilkan informasi pada class Staff yang sudah terdefinisi, termasuk data yang diwarisi dari class Karyawan

1. Perhatikan kode program dibawah ini yang terdapat pada class StaffTetap

Terlihat dipotongan program diatas atribut gaji, lembur dan potongan dapat diakses langsung. Kenapa hal ini bisa terjadi dan bagaimana class StaffTetap memiliki atribut gaji, lembur, dan potongan padahal dalam class tersebut tidak dideklarasikan atribut gaji, lembur, dan potongan?

Jawab:

* Dikarenakan pada class StaffTetap sudah berfungsi sebagai subclass sehingga dapat inheritance dari class Staff yang mana akan dapat memperoleh langsung pada atribut gaji, lembur, dan potongan.

# TUGAS

* 1. Buatlah sebuah program dengan konsep pewarisan seperti pada class diagram berikut ini. Kemudian buatlah instansiasi objek untuk menampilkan data pada class Mac, Windows dan Pc!.

