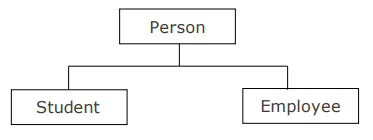
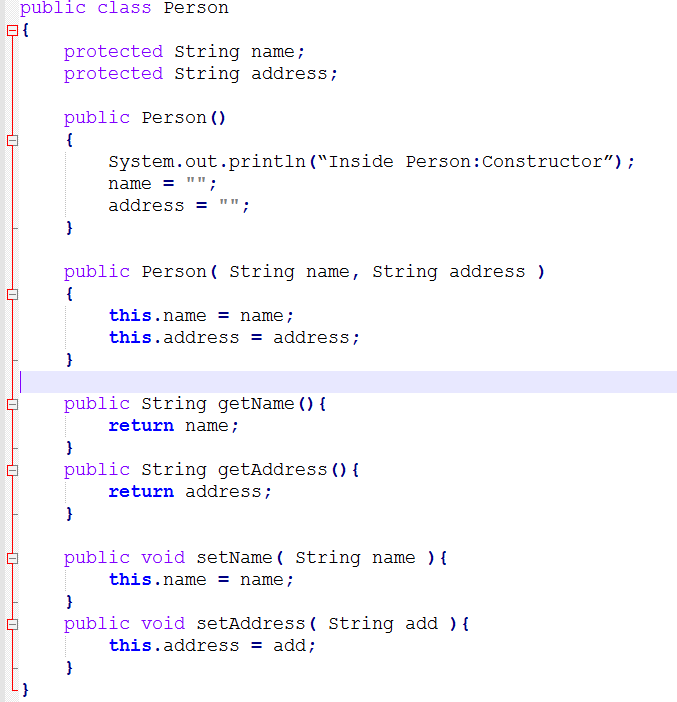
**Polimorfisme**

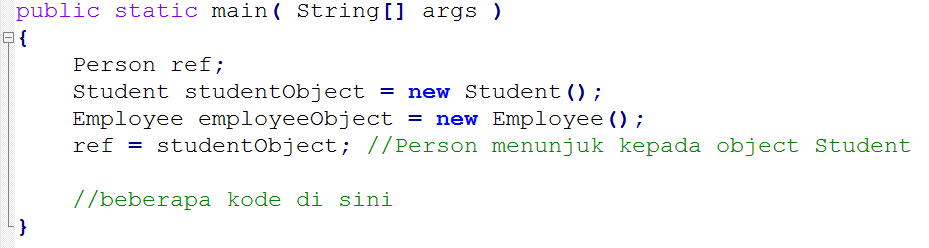
Perhatikan hierarki berikut (class induk Person, subclass Student dan subclass Employee):



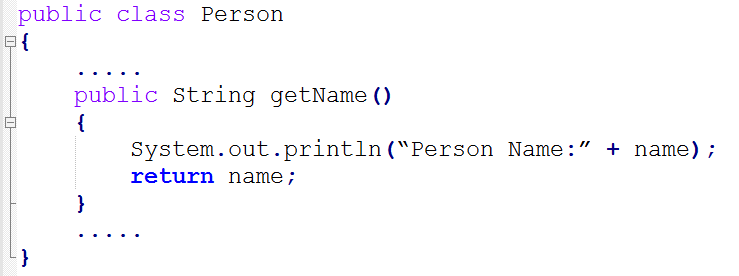
Gambar 1: Hirarki dari class induk Person

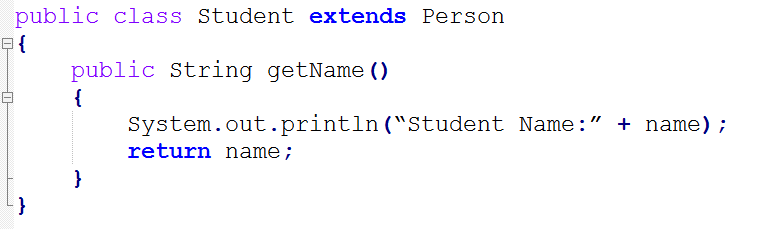


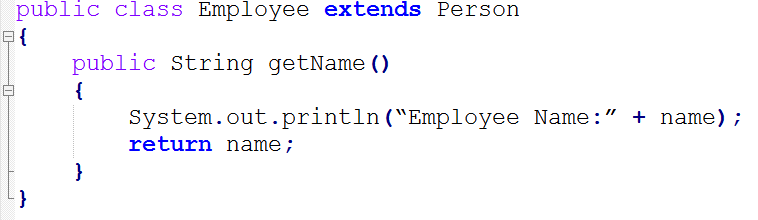
Dalam Java, kita dapat membuat referensi yang merupakan tipe dari superclass ke sebuah object dari subclass tersebut. Sebagai contohnya,



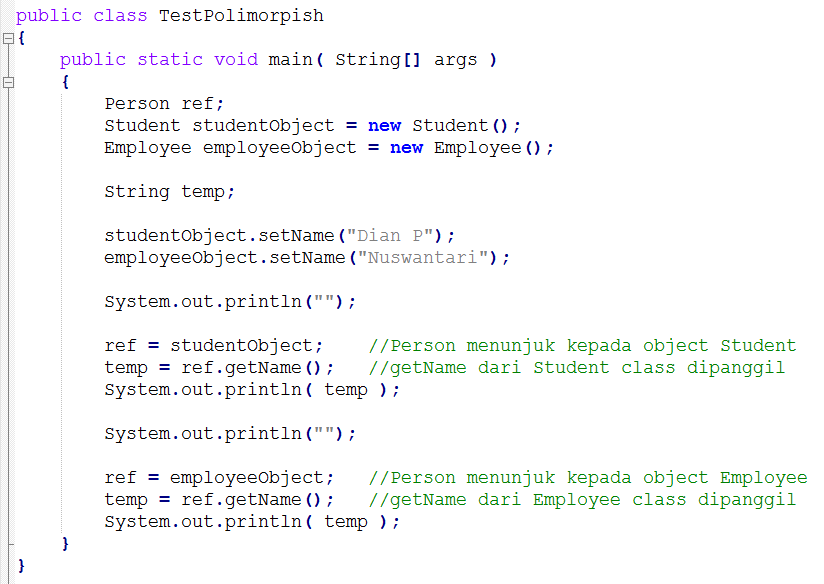
Sekarang dimisalkan kita punya method getName dalam superclass Person kita, dan kita override method ini dalam kedua subclasses Student dan Employee,



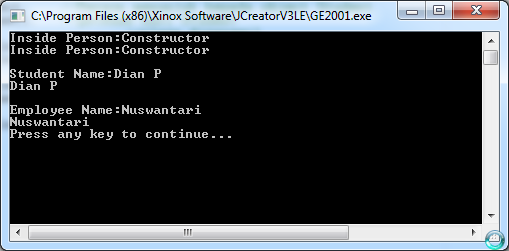




Kembali ke method utama kita, ketika kita mencoba memanggil method getName dari reference Person ref, method getName dari object Student akan dipanggil. Sekarang, jika kita berikan ref ke object Employee, method getName dari Employee akan dipanggil.

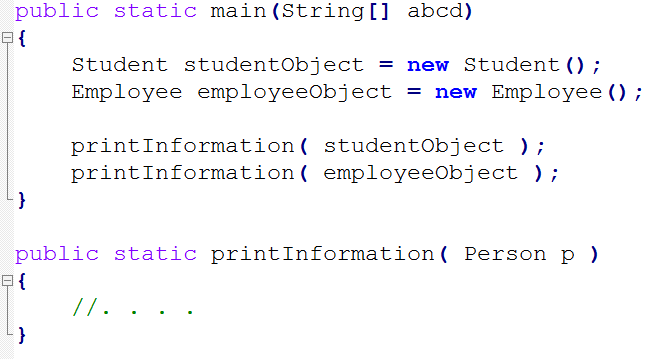


Output Program :



**Kemampuan dari reference untuk mengubah sifat menurut object apa yang dijadikan acuan dinamakan polimorfisme**. Polimorfisme menyediakan multiobject dari subclasses yang berbeda untuk diperlakukan sebagai object dari superclass tunggal, secara otomatis menunjuk method yang tepat untuk menggunakannya ke particular object berdasar subclass yang termasuk di dalamnya.

Contoh lain yang menunjukkan properti polimorfisme adalah ketika kita mencoba melalui reference ke method. Misalkan kita punya method static printInformation yang mengakibatkan object Person sebagai reference, kita dapat me-reference dari tipe Employee dan tipe Student ke method ini selama itu masih subclass dari class Person.



**Latihan**

Buatlah contoh program polimosfisme untuk hirarki berikut :

**Mamalia**

**Sapi**

**Kucing**