

# Praktikum 3 PBO D3MI4001

---

Catatan: untuk diperhatikan, soal hanya merupakan contoh penggunaan pewarisan. Kasus ini bukan kasus real. Di dunia nyata, jenis tabungan dan deposito memiliki kemungkinan tidak memiliki hubungan pewarisan. Konsep pewarisan dapat dipelajari lebih lanjut pada materi/mata kuliah Design Pattern (tidak disampaikan di D3 Manajemen Informatika, FIT, Universitas Telkom, Kurikulum 2016).

## **Materi: Class & Object**

Sebuah software house XCI membuat sebuah aplikasi untuk mendata tabungan bagi pemilik rekening.

Tabungan ini memiliki atribut saldo dan pemilik. Keduanya bersifat protected untuk kemungkinan pengembangan lebih lanjut. Pemilik rekening dapat membuka akun tabungan tanpa proses setor pertama kali. Untuk itu, terdapat 2 buah konstruktor yang menangani:

1. Konstruktor 2 parameter (saldo dan pemilik)
2. Konstruktor 1 parameter (hanya pemilik)

Selain konstruktor tersebut, terdapat method getSaldo, getPemilik, deposit (setor tabungan), withdraw (ambil tabungan), hitungBunga (bunga tabungan 1% dari total saldo), mekanisme tutup rekening, dan display (menampilkan data pemilik dan saldo tabungan). Tidak ada saldo minimal pada rekening ini.

**Berdasarkan deskripsi di atas, bantulah software house XCI untuk melakukan hal berikut:**

- a) Buatlah class diagram dari skenario di atas
- b) Kodekan dengan bahasa pemrograman Java
- c) Buatlah sebuah Main dengan skenario sbb:

<b>Skenario 1:</b>  Membuat tabungan dengan pemilik Gon tanpa setoran awal. Menambah saldo 500 ribu. Melakukan pengambilan dan sebesar 300 ribu. Menghitung bunga. Menabung sebesar 1 juta. Hitung bunga. Setelah setiap transaksi, tampilkan data ke layar dengan memanggil display.	<b>Skenario 2:</b>  Membuat tabungan dengan pemilik Killua dengan setoran awal 500 ribu. Menambah saldo 700 ribu. Melakukan pengambilan dan sebesar 1.300.000. Hitung Bunga. Ambil 300 ribu. Menabung 200 ribu. Hitung bunga. Tutup rekening. Setelah setiap transaksi, tampilkan data ke layar dengan memanggil display.
--	---

**Materi: Pewarisan**

Software house XCI mendapat requirement baru. Untuk melakukan perluasan usaha investasi, terdapat jenis tabungan baru, yaitu “Deposito”. Sifat deposito mirip dengan tabungan biasa, dengan beberapa perbedaan berikut:

- a) Nilai setor pertama minimal 500 ribu.
- b) Nilai nominal saldo tidak dapat ditambah atau dikurangi.
- c) Hanya bisa ditutup akun selama mencapai batas periode tertentu (hitungan periode dalam bulan).
- d) Kelebihan bagi pemilik akun adalah bunga yang lebih besar, yaitu 5% dari jumlah nominal yang disetorkan di awal.
- e) Terdapat display untuk menampilkan data deposito.

Pemilik akun deposito tidak harus membuka tabungan terlebih dahulu.

Skenario di atas membuat software house XCI untuk menggunakan konsep pewarisan pada sistem yang akan dibangun.

Berdasarkan hal di atas, bantulah software house XCI untuk:

- a) Membuat class diagram Deposito
- b) Mengkodekan deposito menggunakan bahasa pemrograman Java
- c) Buatlah sebuah Main dengan skenario sebagai berikut:

**Membuat sebuah objek deposito dengan pemilik “Chrollo Lucifer”, setoran awal 3 juta, selama 3 bulan. Melakukan pengambilan sebesar 1 juta. Melakukan setor 500 ribu. Hitung bunga. Setiap transaksi, tampilkan data di layar.**

**Materi: Polimorfisme (overloading & overriding).**

Keterangan: Polimorfisme merupakan penulisan kembali sebuah method dengan algoritma/body method yang berbeda satu sama lain. Terdapat 2 jenis polimorfisme: overriding & overloading. Overriding merupakan penulisan kembali sebuah method super class pada sub class. Sedangkan overloading merupakan penulisan kembali sebuah method pada class yang sama dengan syarat jumlah dan/atau tipe parameter yang berbeda.

Diketahui 2 buah class dengan kode program sebagai berikut:

```
public class KelasSuper {
    public void methodSatu(int x) { }
}

class KelasAnak extends KelasSuper{
    public void methodSatu(){ }
}
```

**Apakah methodSatu pada KelasAnak merupakan method overriding? Jika keyword public diubah menjadi protected, apa yang terjadi? Jelaskan!**

Berdasarkan method dengan kode di atas, tambahkan kode berikut pada KelasSuper

```
public void methodSatu() { }
```

**Apakah methodSatu di atas pada KelasSuper merupakan method overloading?**

Berdasarkan method dengan kode sebelumnya, tambahkan kode berikut pada KelasSuper

```
public void methodDua(int x, int y){ }
public void methodDua(int y, int x){ }
public void methodDua(String c){ }
```

**Konsep apa yang digunakan pada methodDua? Apakah yang terjadi pada kode di atas? Apakah terjadi error? Jelaskan!**

Tambahkan method berikut pada KelasSuper

```
public int methodSatu(int x){return 0;}
```

**Apakah yang terjadi pada kode di atas? Jelaskan mengapa?**

Tambahkan method berikut pada KelasSuper

```
protected void methodSatu(){ }
public void methodSatu(){ }
```

**Apakah yang terjadi pada kode di atas? Jelaskan mengapa?**

Hal apa yang bisa disimpulkan dari pertanyaan-pertanyaan di atas?