

## TUGAS 3

### Topik: Interface

#### Pendahuluan

Anda diminta membuat program untuk perusahaan Square Inc. Program yang dihasilkan harus dapat membantu dalam bidang **administrasi** ( menghitung gaji, potongan, dsb. ) dan bidang **pekerjaan**. Dalam perusahaan terdapat 2 jenis pegawai, **pegawai tetap** dan **intern**. Kedua jenis pegawai tersebut memiliki penghitungan administrasi dan tugas yang berbeda.

Anda diminta membuat kelas yang mewakili administrasi, pekerjaan, serta kedua jenis pegawai.

#### SOAL 1

Buatlah kelas interface *Administrasi* dan *Kerja*.

Interface *Administrasi* mendeklarasikan metode *hitungGaji()* yang berfungsi untuk menghitung Gaji seorang pegawai berdasarkan jumlah anak dan lama bekerja, metode *hitungPotongan()* yang berfungsi untuk menghitung jumlah pengurangan gaji berdasarkan jumlah cuti yang diambil.

Interface *Kerja* mendeklarasikan metode *kerjaPersegi()* yang berfungsi untuk membantu pekerjaan yang berhubungan dengan persegi, dan metode *kerjaKubus()* yang berfungsi untuk membantu pekerjaan yang berhubungan dengan kubus. Seluruh Prosedur pada Interface yang didefinisikan berfungsi **mencetak hasil masing-masing perhitungan pada layar sesuai dengan aturan yang diberikan**. (Aturan akan dijelaskan di bagian berikutnya)

```
interface Administrasi {
    public void hitungGaji(int anak, int lama);
    public void hitungPotongan(int cuti);
}

interface Kerja {
    public void kerjaPersegi(int s);
    public void kerjaKubus(int r);
}
```

Buatlah kelas abstrak *Pegawai* yang memiliki atribut Lama Kerja, Jumlah Anak, Lama cuti, serta metode abstrak untuk menampilkan informasi dirinya.

```
abstract class Pegawai {
    protected int
    lamaKerja; protected
    int jumlahAnak;
    protected int lamaCuti;
```

```

    public Pegawai(int k, int a, int c) {
        lamaKerja =
        k; jumlahAnak
        = a; lamaCuti
        = c;
    }

    //Memprint seluruh Informasi
    Pegawai abstract void
    printKeterangan();
}

```

Buatlah kelas *Intern* (pekerja magang) yang merupakan anak dari kelas *Pegawai* dan memiliki atribut interface *Administrasi* dan *Kerja*. Kelas *Intern* juga memiliki atribut tambahan yaitu **Asal Sekolah** berupa String. *Intern* memiliki konstruktor yang berparameter Lama Bekerja, Jumlah Anak, Jumlah Cuti, dan Asal Sekolah.

Aturan untuk *Intern*:

- Penghitungan Gaji : Lama Bekerja x **500.000**
- Penghitungan Potongan : Jumlah Cuti x **100.000**
- Pekerjaan : Menghitung **Keliling** Persegi dan **Keliling** Kubus.

Buatlah kelas *Tetap* (pekerja tetap) yang merupakan anak dari kelas *Pegawai* dan memiliki atribut interface *Administrasi* dan *Kerja*. Kelas *Tetap* juga memiliki atribut tambahan yaitu **Jabatan** berupa String. *Tetap* memiliki konstruktor yang berparameter Lama Bekerja, Jumlah Anak, Jumlah Cuti, dan Jabatan.

Aturan untuk *Tetap*:

- Penghitungan Gaji : Lama Bekerja x **1.500.000** + **Jumlah Anak \* 500.000**
- Penghitungan Potongan : Jumlah Cuti x **50.000**
- Pekerjaan : Menghitung **Luas** Persegi dan **Luas** Kubus.

```

public class Intern extends Pegawai {
    public Administrasi adminIntern;
    public Kerja kerjaIntern;
    public String asalSekolah;
    public Intern(int k, int a, int
        c, String as)
    {
        //Konstruktor Intern
    }

    public void setAdministrasi(Administrasi a){
        //Mengeset interface administrasi
    }

    public void setKerja(Administrasi a){
        //Mengeset interface kerja untuk objek
    }
}

```

```

@Override
public void printKeterangan(){
// Mencetak seluruh atribut ke layar
}

public void InternBekerja(){
//Memanggil seluruh prosedur pada interface kerja
}

public void AdministrasiIntern(){
//Memanggil seluruh prosedur pada interface administrasi
}
}

public class Tetap extends Pegawai {
public Administrasi adminTetap;
public Kerja kerjaTetap;
public String jabatan;

public Tetap(int k, int a, int
c, String as)

{
//Konstruktor Intern
}

public void setAdministrasi(Administrasi a){
//Mengeset interface administrasi
}

public void setKerja(Administrasi a){
//Mengeset interface kerja untuk objek
}

@Override
public void printKeterangan(){
// Mencetak seluruh atribut ke layar
}

public void TetapBekerja(){
//Memanggil seluruh prosedur pada interface kerja
}

public void AdministrasiTetap(){
//Memanggil seluruh prosedur pada interface administrasi
}
}

```

Gunakan prosedur setKerja dan setAdministrasi untuk mengimplementasikan interface pada tiap-tiap kelas pada Driver. Interface pada parameter langsung diinstansiasi dan diberi implementasi langsung pada saat pemanggilan prosedur set. Contoh sebagai berikut:

```

Intern Hayoung = new Intern(k1,a1,c1,skol);
Hayoung.setAdministrasi(new Administrasi() {

    @Override
    public void hitungGaji(int anak, int lamakerja) {
        // Isi dengan perhitungan Gaji
    }

    @Override
    public void hitungPotongan(int cuti) {
        // Isi dengan perhitungan potongan
    }

});

```

Buatlah kelas Main sebagai driver untuk mengimplementasikan seluruh fungsi dan metode yang dibuat.

```

public class Main {

    public static void main(String[] args) {

    }

}

```