

LAPORAN TUGAS HARI RAYA
Sistem Menejemen Karyawan di Perusahaan
Oleh Muhammad Nizar Wicaksono
242410103066

Kode

```
using System;
```

```
class Karyawan
```

```
{
```

```
    private string nama;
```

```
    private string id;
```

```
    private double gajiPokok;
```

```
    public string Nama
```

```
    {
```

```
        get { return nama; }
```

```
        set { nama = value; }
```

```
    }
```

```
    public string ID
```

```
    {
```

```
        get { return id; }
```

```
        set { id = value; }
```

```
    }
```

```
    public double GajiPokok
```

```
    {
```

```
        get { return gajiPokok; }
```

```
        set { gajiPokok = value; }
```

```
}
```

```
public Karyawan(string nama, string id, double gajiPokok)
```

```
{
```

```
    this>Nama = nama;
```

```
    this.ID = id;
```

```
    this.GajiPokok = gajiPokok;
```

```
}
```

```
public virtual double HitungGaji()
```

```
{
```

```
    return GajiPokok;
```

```
}
```

```
}
```

```
class KaryawanTetap : Karyawan
```

```
{
```

```
    private const double BonusTetap = 500000;
```

```
public KaryawanTetap(string nama, string id, double gajiPokok)
```

```
    : base(nama, id, gajiPokok) { }
```

```
public override double HitungGaji()
```

```
{
```

```
    return GajiPokok + BonusTetap;
```

```
}
```

```
}
```

```
class KaryawanKontrak : Karyawan
```

```

{
    private const double PotonganKontrak = 200000;

    public KaryawanKontrak(string nama, string id, double gajiPokok)
        : base(nama, id, gajiPokok) { }

    public override double HitungGaji()
    {
        return GajiPokok - PotonganKontrak;
    }
}

```

```

class KaryawanMagang : Karyawan
{
    public KaryawanMagang(string nama, string id, double gajiPokok)
        : base(nama, id, gajiPokok) { }

    public override double HitungGaji()
    {
        return GajiPokok;
    }
}

```

```

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Pilih jenis karyawan:");
        Console.WriteLine("1. Karyawan Tetap");
    }
}

```

```
Console.WriteLine("2. Karyawan Kontrak");  
Console.WriteLine("3. Karyawan Magang");  
Console.Write("Pilihan Anda (1/2/3): ");  
int pilihan = Convert.ToInt32(Console.ReadLine());
```

```
Console.Write("Masukkan Nama: ");  
string nama = Console.ReadLine();  
Console.Write("Masukkan ID: ");  
string id = Console.ReadLine();  
Console.Write("Masukkan Gaji Pokok: ");  
double gajiPokok = Convert.ToDouble(Console.ReadLine());
```

```
Karyawan karyawan;
```

```
switch (pilihan)
```

```
{  
    case 1:  
        karyawan = new KaryawanTetap(nama, id, gajiPokok);  
        break;  
    case 2:  
        karyawan = new KaryawanKontrak(nama, id, gajiPokok);  
        break;  
    case 3:  
        karyawan = new KaryawanMagang(nama, id, gajiPokok);  
        break;  
    default:  
        Console.WriteLine("Pilihan tidak valid!");  
        return;  
}
```

```
        Console.WriteLine($"Gaji akhir karyawan atas nama {karyawan>Nama} dengan ID {karyawan.ID} adalah Rp {karyawan.HitungGaji()}");  
    }  
}
```

Penjelasan

Pada bagian awal saya membuat parent class bernama Karyawan yang mana class ini nantinya akan menjadi cetakan untuk class turunannya. Dalam class Karyawan ini nantinya akan menyimpan nama, id, dan gajipokok yang dimana untuk nama dan id bertipe data string, dan untuk gajipokok bertipe data double. Semua data tersebut di set private agar tidak bisa diubah langsung dari luar class. Selanjutnya saya membuat atribut public string nama berisi command get untuk mereturn nama dan set yang akan menetapkan nilai nama menjadi value. Kemudian ada atribut public string id berisi get untuk mereturn id dan set yang akan menetapkan nilai id menjadi value. Kemudian ada atribut public double gajipokok yang berisi get untuk mereturn gajipokok dan set untuk menetapkan nilai gajipokok menjadi value. Kemudian saya membuat constructor Karyawan yang memiliki 3 parameter yaitu string nama, string id, dan double gajipokok. Constructor ini digunakan untuk menginisialisasi atau mengisi nilai dari atribut yang ada di dalam class Karyawan, yaitu Nama, ID, dan GajiPokok. Kemudian dalam constructor ini masing-masing menggunakan this untuk menghindari ambiguitas antara variabel dari parameter dengan atribut yang ada pada objek. Dan yang terakhir ada method untuk menghitung gaji yaitu public virtual double HitungGaji yang nantinya akan mereturn nilai gajipokok. Method ini menggunakan virtual agar method ini bisa di override di class turunannya.

Selanjutnya saya membuat child class dari Karyawan yang pertama ada class KaryawanTetap yang di dalam class ini saya membuat private const double BonusTetap yang diisi dengan 500000. Kemudian saya membuat constructor yang memanggil constructor Karyawan dengan base karena class KaryawanTetap merupakan kelas turunan dari class Karyawan. Setelah itu ada public override HitungGaji yang mereturn gaji pokok ditambah dengan BonusTetap untuk menghitung gaji akhir karyawan.

Selanjutnya saya membuat child class kedua dari class Karyawan yaitu class KaryawanKontrak yang di dalam class ini saya membuat private const double PotonganKontrak yang diisi dengan 200000. Kemudian saya membuat constructor yang memanggil constructor Karyawan dengan base karena class KaryawanKontrak merupakan kelas turunan dari class Karyawan. Setelah itu ada public override HitungGaji yang mereturn gaji pokok dikurangi dengan PotonganKontrak untuk menghitung gaji akhir karyawan.

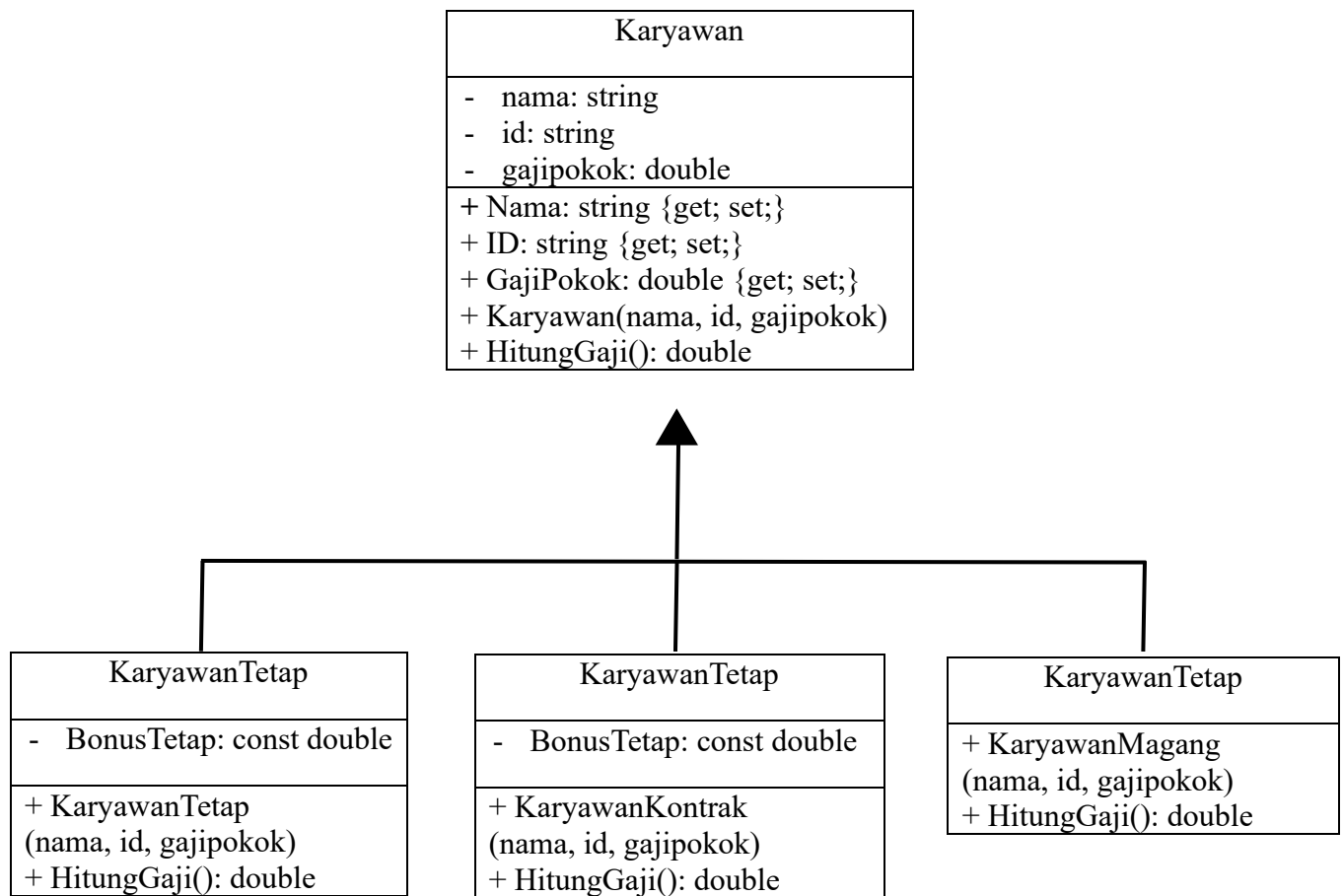
Selanjutnya saya membuat child class ketiga dari class Karyawan yaitu class KaryawanMagang yang di dalam class ini saya membuat constructor yang memanggil constructor Karyawan dengan base karena class KaryawanMagang merupakan kelas turunan

dari class Karyawan. Setelah itu ada public override HitungGaji yang mereturn gaji pokok untuk menghitung gaji akhir karyawan.

Yang terakhir saya membuat class Program yang berisikan static void Main(string[] args) yang akan mengembalikan nilai string, dan juga dimana ini merupakan method yang pertama dijalankan. Di dalamnya berisi perintah kepada user. Yang pertama akan memunculkan “Pilih Jenis Karyawan” yang nantinya user diminta menginputkan nomer sesuai dengan status karyawan dari user. Setelah memilih jenis karyawan, nantinya akan muncul “Masukkan Nama” yang mana user diminta untuk menginputkan nama dari user. Selanjutnya akan muncul “Masukkan ID” yang mana user diminta untuk menginputkan ID karyawan user. Selanjutnya akan muncul “Masukkan Gaji Pokok” yang mana user diminta untuk menginputkan jumlah gaji pokok karyawan.

Selanjutnya saya menggunakan switch case yang memiliki parameter pilihan. Jika case 1, nantinya akan membuat new KaryawanTetap dan mengeksekusi class KaryawanTetap dan di brake. Jika case 2, nantinya akan membuat new KaryawanKontrak dan mengeksekusi class KaryawanKontrak dan di brake. Jika case 3, nantinya akan membuat new KaryawanMagang dan mengeksekusi class KaryawanMagang dan di brake. Dan diakhir ada kondisi default dari switch case yaitu menampilkan “Pilihan tidak valid”. Setelah semua inputan sudah selesai di input, nantinya akan menampilkan gaji akhir dari karyawan, yang dimana karyawan.Nama sebagai nama, karyawan.ID sebagai id, dan karyawan.GajiAkhir() sebagai nilai dari gaji akhir karyawan.

Diagram Class



Penjelasan Diagram Class

1. Kelas Karyawan (Parent Class)

- o Memiliki atribut nama, id, dan gaji pokok.
- o Menggunakan getter dan setter untuk mengakses data.
- o Menggunakan constructor yang menerima nama, id, dan gajipokok sebagai parameter
- o Memiliki method `HitungGaji` yang akan di override oleh kelas turunannya

2. Kelas KaryawanTetap (Child Class)

- o Menambahkan bonus 500000 ke gaji pokok
- o Memanggil constructor `Karyawan` dengan base
- o Override method `HitungGaji()`

3. Kelas KaryawanKontrak (Child Class)

- o Mengurangi 200000 dari gaji pokok
- o Memanggil constructor `Karyawan` dengan base
- o Override method `HitungGaji()`

4. Kelas KarywanMagang (Child Class)

- o Memanggil constructor `Karyawan` dengan base
- o Override method `HitungGaji()` tanpa perubahan