**Laporan PBO**

Sistem Management Gaji Karyawan di Perusahaan

Mohammad Alfin Kamal Saputra / 242410103039

using System;

namespace gajikaryawan

{

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("Masukkan Pilihan :");

string pilihan = Console.ReadLine();

Console.Write("masukkan nama karyawan: ");

string nama = Console.ReadLine();

Console.Write("masukkan id karyawan:");

string idnama = Console.ReadLine();

Karyawan karyawan;

switch (pilihan)

{

case "1":

karyawan = new Karyawantetap(nama, idnama, 5000000);

break;

case "2":

karyawan = new Karyawankontrak(nama, idnama, 5000000);

break;

case "3":

karyawan = new Karyawanmagang(nama, idnama, 1000000);

break;

default:

Console.WriteLine("Pilihan tidak valid.");

return;

}

Console.WriteLine($"\nNama: {karyawan.Nama}");

Console.WriteLine($"ID: {karyawan.Idnama}");

Console.WriteLine($"Gaji Akhir: {karyawan.Hitunggaji()}");

}

class Karyawan

{

private string nama;

private string idnama;

private double gajipokok;

public Karyawan(string nama, string idnama, double gajipokok)

{

this.nama = nama;

this.idnama = idnama;

this.gajipokok = gajipokok;

}

public string Nama

{

get { return nama; }

set { nama = value; }

}

public string Idnama

{

get { return idnama; }

set { idnama = value; }

}

public double Gajipokok

{

get { return gajipokok; }

set { gajipokok = value; }

}

public virtual double Hitunggaji()

{

return gajipokok;

}

}

class Karyawantetap : Karyawan

{

public Karyawantetap(string nama, string idnama, double gajipokok) : base(nama, idnama, gajipokok) { }

public override double Hitunggaji()

{

return Gajipokok + 500000;

}

}

class Karyawankontrak : Karyawan

{

public Karyawankontrak(string nama, string idnama, double gajipokok) : base(nama, idnama, gajipokok) { }

public override double Hitunggaji()

{

return Gajipokok - 200000;

}

}

class Karyawanmagang : Karyawan

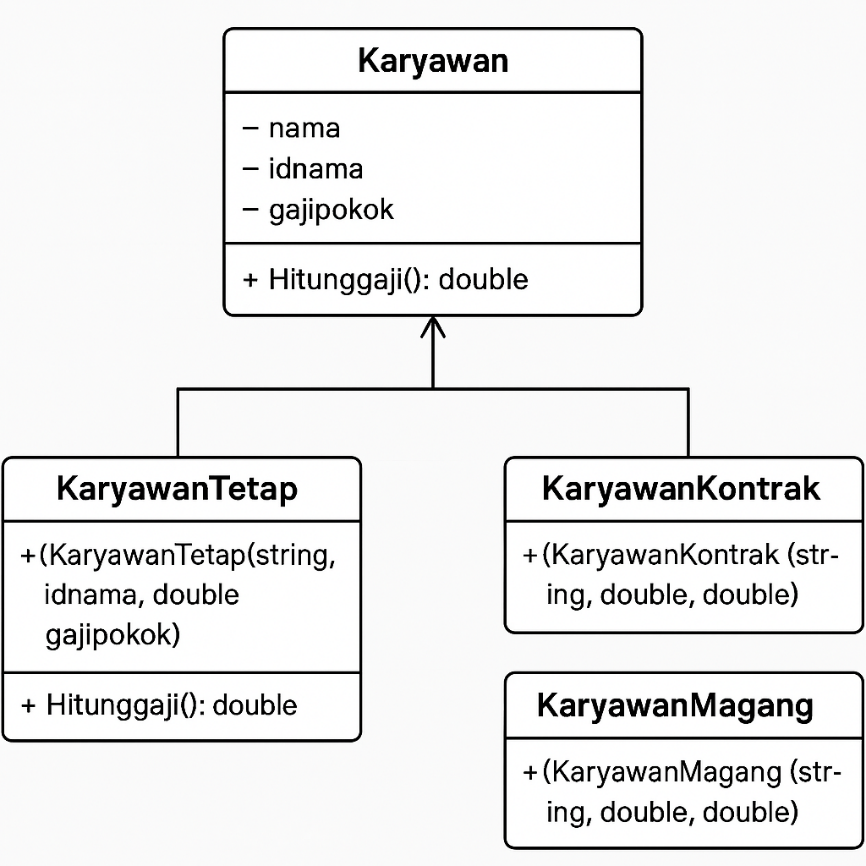
{

public Karyawanmagang(string nama, string idnama, double gajipokok) : base(nama, idnama, gajipokok) { }

}

}

}

**Penjelasan dari struktur kode diatas :**

1. **Namespace dan kelas program**

namespace gajikaryawan

{

class Program

{

static void Main(string[] args)

{

* Kode ini dimulai dengan mendefinisikan sebuah namespace bernama **gajikaryawan**, yang berfungsi untuk mengelompokkan kelas-kelas yang terkait.
* Di dalam namespace, terdapat kelas **Program** yang berisi metode **Main**, yang merupakan titik masuk (entry point) dari program.

1. **Input dari pengguna**

Console.WriteLine("pilih jenis karyawan:");

Console.WriteLine("1. Karyawan Tetap");

Console.WriteLine("2. Karyawan Kontrak");

Console.WriteLine("3. Karyawan Magang");

Console.Write("Masukkan Pilihan :");

string pilihan = Console.ReadLine();

Console.Write("masukkan nama karyawan: ");

string nama = Console.ReadLine();

Console.Write("masukkan id karyawan:");

string idnama = Console.ReadLine();

* Program meminta pengguna untuk memilih jenis karyawan dengan menampilkan pilihan yang tersedia.
* Input dari pengguna disimpan dalam variabel **pilihan**.

1. **Input data karyawan**

Console.Write("masukkan nama karyawan: ");

string nama = Console.ReadLine();

Console.Write("masukkan id karyawan:");

string idnama = Console.ReadLine();

* Program meminta pengguna untuk memasukkan nama dan ID karyawan, yang disimpan dalam variabel **nama** dan **idnama**.

1. **Instance karyawan**

Karyawan karyawan;

switch (pilihan)

{

case "1":

karyawan = new Karyawantetap(nama, idnama, 5000000);

break;

case "2":

karyawan = new Karyawankontrak(nama, idnama, 5000000);

break;

case "3":

karyawan = new Karyawanmagang(nama, idnama, 1000000);

break;

default:

Console.WriteLine("Pilihan tidak valid.");

return;

}

* Berdasarkan pilihan pengguna, program akan membuat objek dari kelas yang sesuai (**Karyawan tetap**, **Karyawan kontrak**, atau **Karyawan magang**).
* Jika pilihan tidak valid, program akan menampilkan pesan dan keluar.

1. **Tampilkan informasi karyawan**

Console.WriteLine($"\nNama: {karyawan.Nama}");

Console.WriteLine($"ID: {karyawan.Idnama}");

Console.WriteLine($"Gaji Akhir: {karyawan.Hitunggaji()}");

* Setelah objek karyawan dibuat, program menampilkan nama, ID, dan gaji akhir karyawan dengan memanggil metode **Hitunggaji()**.
* override dari metode **HitungGaji()** pada kelas induk. Gaji akhir diperoleh dengan menambahkan bonus tetap ke dalam gaji pokok.

1. **Kelas karyawan**

class Karyawan

{

private string nama;

private string idnama;

private double gajipokok;

public Karyawan(string nama, string idnama, double gajipokok)

{

this.nama = nama;

this.idnama = idnama;

this.gajipokok = gajipokok;

}

public string Nama

{

get { return nama; }

set { nama = value; }

}

public string Idnama

{

get { return idnama; }

set { idnama = value; }

}

public double Gajipokok

{

get { return gajipokok; }

set { gajipokok = value; }

}

public virtual double Hitunggaji()

{

return gajipokok;

}

}

* Class **Karyawan** adalah class dasar (superclass) yang memiliki tiga atribut: **nama**, **idnama**, dan **gajipokok**.
* Ada atribut **private** yang berfungsi untuk menjaga data agar tetap privasi
* Terdapat konstruktor untuk menginisialisasi atribut-atribut tersebut.
* Terdapat properti untuk mengakses dan memodifikasi atribut.
* Metode **Hitunggaji()** adalah metode virtual yang mengembalikan gaji pokok.

1. **Sub class karyawan**

class Karyawantetap : Karyawan

{

public Karyawantetap(string nama, string idnama, double gajipokok) : base(nama, idnama, gajipokok) { }

public override double Hitunggaji()

{

return Gajipokok + 500000;

}

}

class Karyawankontrak : Karyawan

{

public Karyawankontrak(string nama, string idnama, double gajipokok) : base(nama, idnama, gajipokok) { }

public override double Hitunggaji()

{

return Gajipokok - 200000;

}

}

class Karyawanmagang : Karyawan

{

public Karyawanmagang(string nama, string idnama, double gajipokok) : base(nama, idnama, gajipokok) { }

}

}

* **Karyawantetap**: Subclass dari **Karyawan** yang menambahkan tunjangan sebesar 500.000 pada gaji pokok.
* **Karyawankontrak**: Subclass dari **Karyawan** yang mengurangi gaji pokok sebesar 200.000.
* **Karyawanmagang**: Subclass dari **Karyawan** yang tidak mengubah gaji pokok.
* **Override** untuk mengembalikan nilai gaji pokok dengan tambahan tunjangan.