1. **DEFINISI MASALAH**

1. Buatlah class Employee yang d-inherit oleh class Manager, Pegawai tetap, Pegawai tidak tetap. Setelah itu program akan meminta user untuk menginputkan nama, idKerja, jabatan employee, mempunyai istri atau tidak, mempunyai anak atau tidak dan memasukkan tahun masuk kerja. Ketentuan untuk mengerjakan soal ini adalah sebagai berikut :

1. Berlaku untuk semua jabatan (kecuali non PNS)

-  Jika masa kerja di bawah 5 tahun maka tidak mendapatkan bonus dan tidak  mendapatkan tunjangan (sama dengan employee)

-  Jika masa kerja di antara 6-10 tahun maka mendapatkan bonus 0.05 dari lama  kerja dan tidak mendapatkan tunjangan

-  Jika masa kerja di atas 10 tahun maka mendapatkan bonus 0.1 dari lama kerja  dan mendapatkan tunjangan 0.1 dari lama kerja

-  Jika mempunyai istri maka akan mendapatkan tunjangan 0.1 dari total gaji yang  didapat (gaji+lama kerja+tunjangan) jika tidak maka tunjangan istri 0 rupiah

-  Jika mempunyai anak maka mendapatkan tunjangan anak sebesar 0.15 dari jumlah gaji yang di dapatkan (gaji+lama kerja+tunjangan), tunjangan anak hanya  diberikan sampai anak ketiga saja.

1. Berlaku untuk Manager saja

- Untuk manager selain mendapatkan bonus kerja dan tunjangan, pada manager mendapatkan tunjangan jabatan sebesar 0.1 dari jumlah gaji (gaji+lama kerja+tunjangan).

1. Berlaku untuk pegawai tidak tetap - Gaji yang didapatkan adalah sama dengan gaji employee, namun terdapat gaji

-lembur. Untuk mendapatkan gaji lembur, pegawai non PNS harus bekerja lebih dari 10 jam dengan mendapatkan gaji lembur sebesar 10.000 perjamnya. Namun jika di bawah 10 jam maka tidak akan mendapatkan gaji lembur.

-Untuk mengerjakan soal ini gunakan test case sebanyak 1 Manager, 10 Pegawai tetap dan 5 pegawai tidak tetap yang mencakup seluruh ketentuan yang ada.

1. **SOURCE CODE**

|  |  |
| --- | --- |
| CLASS Employee | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  2324  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59 | package tugasprakbab6;  public class Employee {  protected String nama, idKerja, jabatan, status;  double tahun;  double lama;  double anak;  double gaji;  double bonus;  double tunjangan, tunIstri, tunAnak;  Employee(String nama, String idKerja, String jabatan, String status, double anak, double tahun) {  this.nama = nama;  this.idKerja = idKerja;  this.jabatan = jabatan;  this.status = status;  this.anak = anak;  this.tahun = tahun;  this.lama = 2016 - tahun;  }  public void setBonus() {  if (jabatan.equalsIgnoreCase("manager")) {  this.gaji = 20000000;  } else if (jabatan.equalsIgnoreCase("Pegawai")) {  this.gaji = 10000000;  } else if (jabatan.equalsIgnoreCase("Pegawaitidaktetap")) {  this.gaji = 5000000;  }  if (6 < this.lama && this.lama < 10) {  this.bonus = 0.05 \* this.lama;  this.gaji = (this.bonus \* this.gaji) + this.gaji;  } else if (lama > 10) {  this.bonus = 0.1 \* this.lama;  this.tunjangan = 0.1 \* this.lama;  this.gaji = (this.bonus \* this.gaji) + (this.tunjangan \* this.gaji) + this.gaji;  }  if (status.equalsIgnoreCase("nikah")) {  this.tunIstri = 0.1 \* this.gaji;  }  if (anak > 0 && anak < 4) {  this.tunAnak = 0.15 \* this.gaji \* anak;  } else if (anak > 3) {  this.tunAnak = 0.15 \* this.gaji \* anak \* 3;  }  }  } |

|  |  |
| --- | --- |
| Class Manager | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | package tugasprakbab6;  public class Manager extends Employee {  private double tunJabatan;  public Manager(String nama, String idKerja, String jabatan, String status, double anak, double tahun) {  super(nama, idKerja, jabatan, status, anak, tahun);  }  public void setTunjangan() {  this.tunJabatan = 0.1 \* (gaji + tunIstri + tunAnak);  gaji = gaji + tunJabatan;  }  } |

|  |  |
| --- | --- |
| CLASS PegawaiTetap | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | package tugasprakbab6;  public class PegawaiTetap extends Employee {    public PegawaiTetap(String nama, String idKerja, String jabatan, String status, double anak, double tahun) {  super(nama, idKerja, jabatan, status, anak, tahun);  }  public void setGatot() {  gaji = gaji + tunIstri+tunAnak;  }    } |

|  |  |
| --- | --- |
| CLASS PegawaiTidakTetap | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | package tugasprakbab6;  public class PegawaiTdkTetap extends Employee {  double lembur;  double kerja;  public PegawaiTdkTetap(String nama, String idKerja, String jabatan, String status, double anak, double tahun) {  super(nama, idKerja, jabatan, status, anak, tahun);  }  public void setLembur(double kerja){  this.kerja =kerja;  if (kerja > 10){  this.lembur = this.kerja \* 10000;  gaji = gaji + this.lembur ;  }  }  } |

|  |  |
| --- | --- |
| CLASS test | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136 | package tugasprakbab6;  import java.util.Scanner;  public class test {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  int pilihan = 0;  String nama = null, id = null, jabatan = null, status = null;  double anak = 0, tahun = 0;  double kerja = 0;  do {  System.out.println("1.Input data");  System.out.println("2.Cek gaji");  System.out.println("3.Exit");  System.out.print("Masukan Pilihan mu : ");  pilihan = in.nextInt();  switch (pilihan) {  case 1:  System.out.print("Masukan nama : ");  nama = in.next();  System.out.print("Masukan idKerja : ");  id = in.next();  System.out.print("Masukan jabatan(manager/pegawai/pegawaitidaktetap) : ");  jabatan = in.next();  if (jabatan.equalsIgnoreCase("pegawaitidaktetap")) {  System.out.print("Masukan Jam kerja : ");  kerja = in.nextDouble();  }  System.out.print("Status pernikahan : ");  status = in.next();  System.out.print("Jumlah anak : ");  anak = in.nextDouble();  System.out.print("Masukan tahun bekerja : ");  tahun = in.nextDouble();  break;  case 2:  if (jabatan.equalsIgnoreCase("manager")) {  Manager a = new Manager(nama, id, jabatan, status, anak, (int) tahun);  a.setBonus();  a.setTunjangan();  System.out.println("Nama : " + a.nama);  System.out.println("IdKerja : " + a.idKerja);  System.out.println("Jabatan : " + a.jabatan);  System.out.println("Status Pernikahan : " + a.status);  System.out.printf("Jumlah Anak : %1.0f ", a.anak);  System.out.println("");  System.out.printf("Tahun bekerja : %4.0f ", a.tahun);  System.out.println("");  System.out.printf("Gaji = Rp. %8.0f ", a.gaji);  System.out.println("");  } else if (jabatan.equalsIgnoreCase("pegawai")) {  PegawaiTetap b = new PegawaiTetap(nama, id, jabatan, status, anak, (int) tahun);  b.setBonus();  b.setGatot();  System.out.println("Nama : " + b.nama);  System.out.println("IdKerja : " + b.idKerja);  System.out.println("Jabatan : " + b.jabatan);  System.out.println("Status Pernikahan : " + b.status);  System.out.printf("Jumlah Anak : %1.0f ", b.anak);  System.out.println("");  System.out.printf("Tahun bekerja : %4.0f ", b.tahun);  System.out.println("");  System.out.printf("Gaji =Rp. %8.0f ", b.gaji);  System.out.println("");  } else if (jabatan.equalsIgnoreCase("pegawaitidaktetap")) {  PegawaiTdkTetap c = new PegawaiTdkTetap(nama, id, jabatan, status, anak, (int) tahun);  c.setBonus();  c.setLembur(kerja);  System.out.println("Nama : " + c.nama);  System.out.println("IdKerja : " + c.idKerja);  System.out.println("Jabatan : " + c.jabatan);  System.out.println("Status Pernikahan : " + c.status);  System.out.printf("Jumlah Anak : %1.0f ", c.anak);  System.out.println("");  System.out.printf("Tahun bekerja : %4.0f ", c.tahun);  System.out.println("");  System.out.printf("Gaji =Rp. %8.0f ", c.gaji);  System.out.println("");  }  break;  case 3:  break;  }  } while (pilihan != 3);  }  } |

1. **PEMBAHASAN**

|  |  |
| --- | --- |
| CLASS Employee | |
| 1  3  5  6  7  8  9  10  11  13  14  15  16  17  18  19  20  21  24  25  26  27  28  29  32  34  35  37  40  41  42  43  44  47  48  51  52  54  55 | Deklarasi package tugas.bab6  Deklarasi class dengan nama Employee  Deklarasi variabel nama,idKerja,jabatan , status dengan tipe Data String  Deklarasi variabel tahun dengan tipe data double  Deklarasi variabel lama dengan tipe data double  Deklarasi variabel anak dengan tipe data double  Deklarasi variabel gaji dengan tipe data double  Deklarasi variabel bonus dengan tipe data double  Deklarasi variabel tunjungan,tunIstri,tunAnak dengan tipe data double  Membuat construktor class Employee dengan parameter String nama, String idKerja, String jabatan, String status, double anak, double tahun  Menginisiasi variabel this.nama dengan nilai dari variabel nama  Menginisiasi variabel this.idKerja dengan nilai dari variabel idKerja  Menginisiasi variabel this.jabatan dengan nilai dari variabel jabatan  Menginisiasi variabel this.status dengan nilai dari variabel status  Menginisiasi variabel this.anak dengan nilai dari variabel anak  Menginisiasi variabel this.tahun dengan nilai dari variabel tahun  Menginisiasi variabel this.lama dengan nilai dari 2016 dikurangin dengan nilai dari variabel tahun  Membuat method void setBonus()  Seleksi kondisi if jabatan.equalsIgnoreCase("manager")) maka  Menginisiasi variabel this.gaji dengan nilai sebesar 2000000  else if jika jabatan.equalsIgnoreCase("Pegawai"))  Menginisiasi variabel this.gaji dengan nilai sebesar 1000000  else if jika jabatan.equalsIgnoreCase("Pegawaitidaktetap"))  Menginisiasi variabel this.gaji dengan nilai sebesar 500000  Seleksi kondisi if 6 < this.lama && this.lama < 10 maka  Menginisiasi variabel this.gaji dengan nilai dari perkalian variabel lama dengan 0.05  Menginisiasi variabel this.gaji dengan nilai dari perkalian variabel bonus dan gaji ditambah dengan variabel gaji  Else if jika lama > 10 maka  Menginisiasi variabel this.bonus dengan nilai dari perkalian variabel lama dengan 0.1  Menginisiasi variabel this.tunjangan dengan nilai dari perkalian variabel lama dengan 0.1  Menginisiasi variabel this.gaji dengan nilai dari perkalian variabel bonus dengan gaji lalu ditambahkan perkalian tunjangan dengan gaji ditambah gaji  Seleksi kondisi if else jika status.equalsIgnoreCase("nikah")) maka  Menginisiasi variabel this.tunIstri dengan nilai dari perkalian variabel gaji dengan 0.1  Seleksi kondisi if else jika anak > 0 && anak < 4 maka  Menginisiasi variabel this.tunAnak dengan nilai dari perkalian 0.15 dengan variabel gaji dan anak  Seleksi kondisi if else jika anak > 3 maka  Menginisiasi variabel this.tunAnak dengan nilai dari perkalian 0.15 dengan variabel anak dan 3 |

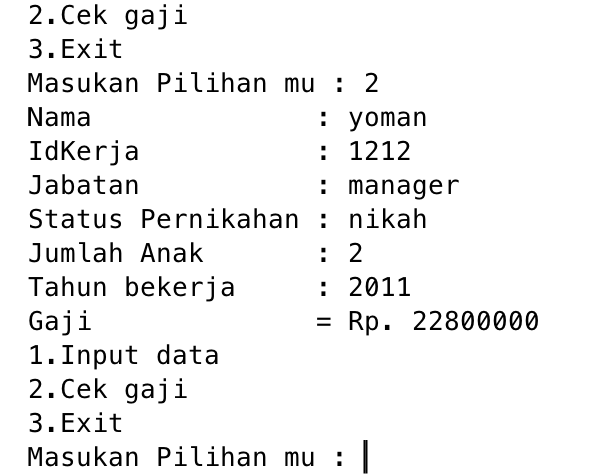
|  |  |
| --- | --- |
| CLASS Manager | |
| 1  3  5  7  8  9  10  13  14  15  16 | Deklarasi package dengan nama tugas.bab6  Deklarasi class Manager yang diextends dari class Employee  Deklarasi variabel tunJabatan dengan tipe data double  Membuat construktor class Manager dengan parameter parameter String nama, String idKerja, String jabatan, String status, double anak, double tahun  Membuat constuktor super dari superclass dengan parameter parameter nama, idKerja, jabatan, status, anak, tahun  Membuat method void setTunjangan()  Menginisiasi variabel this.tunJabatan dengan nilai dari perkalian 0.1 \* (gaji + tunIstri +tunAnak)  Menginisiasi variabel gaji dengan nilai dari pertambahan variabel gaji dengan tunJabatan |

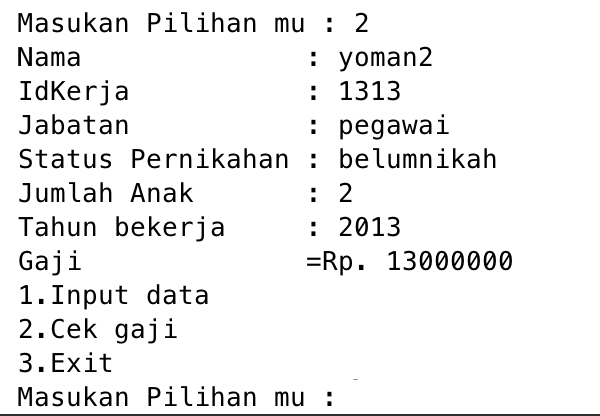
|  |  |
| --- | --- |
| CLASS PegawaiTetap | |
| 2  4  6  7  9  10  12  13 | Deklarasi package tugas.bab6  Deklarasi class dengan nama PegawaiTetap yang diextends dari kelas Employee  Membuat construktor class PegawaiTetap dengan parameter parameter String nama, String idKerja, String jabatan, String status, double anak, double tahun  Membuat constuktor super dari superclass dengan parameter parameter nama, idKerja, jabatan, status, anak, tahun  Membuat method void setGatot()  Menginisiasi variabel gaji dengan nilai dari pertambahan variabel gaji dengan tunIstri ditambah tunAnak |

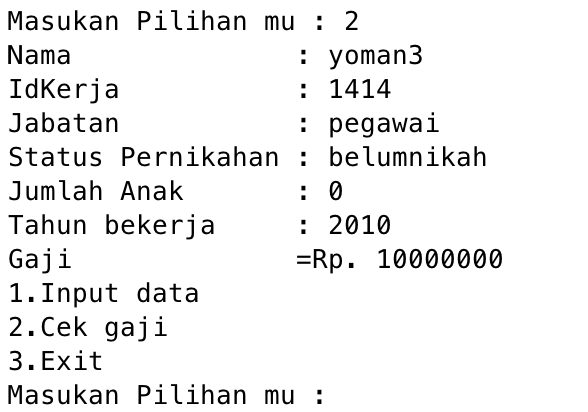
|  |  |
| --- | --- |
| CLASS PegawaiTidakTetap | |
| 2  4  5  6  7  8  10  11  13  14  15  16  17 | Deklarasi package dengan nama tugas.bab6  Deklarasi class dengan nama PegawaiTdkTetap yang diextends dari kelas Employee  Deklarasi variabel lembur dengan tipe data double  Deklarasi variabel kerja dengan tipe data double  Membuat construktor class PegawaiTdkTetap dengan parameter parameter String nama, String idKerja, String jabatan, String status, double anak, double tahun  Membuat constuktor super dari superclass dengan parameter parameter nama, idKerja, jabatan, status, anak, tahun  Membuat method void setLembur() dengan parameter double kerja  Menginisiasi variabel this.kerja dengan nilai dari variabel kerja  Seleksi kondisi if jika kerja > 10 maka  Menginisiasi variabel this.lembur dari nilai This.kerja \* 1000  MenginisiasI variabel gaji dari gaji ditambah nilai dari variabel this.lembur |

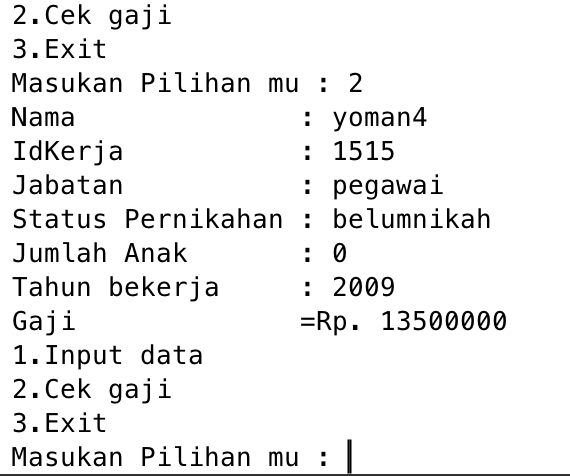
|  |  |
| --- | --- |
| CLASS test | |
| 1  3  5  7  8  9  10  11  12  13  14  15  16  17  18  19  21  22  23  25  26  28  29  31  32  34  36  39  41  42  43  44  46  48  49  51  53  54  55  56  57  59  61  63  65  67  68  70  71  73  75  77  80  81  82  84  86  88  90  92  93  95  96  98  100  102  105  106  107  109  111  113  115  117  118  120  121  123  125  127  128  132 | Deklarasi package dengan nama tugas.bab6  Mengimport fungsi java.util.Scanner  Deklarasi class dengan nama test  Deklarasi main method  Memanggil fungsi Scanner in  Deklarasi variabel pilihan dengan tipe data int dan mengisi nilainya dengan 0  Deklarasi variabel nama , id , jabatan ,status dengan tipe data String dan mengisi nilainya dengan null  Deklarasi variabel anak tahun dengan tipe data double dan mengisi nilainya dengan 0  Deklarasi variabel kerja dengan tipe data double dan mengisi nilainya dengan 0  Perulangan do  Proses mencetak " "1.Input data""  Proses mencetak " 2.Cek gaji""  Proses mencetak " "3.Exit""  Proses mencetak " Masukan Pilihan mu : "  Proses input variabel pilihan  Membuat seleksi kondisi Switch case pada variabel pilihan  Seleksi kondisi jika user memilih 1  Proses mencetak "Masukan nama : "  Proses input variabel nama  Proses mencetak "Masukan idKerja : "  Proses input variabel id  Proses mencetak "Masukan jabatan(manager/pegawai/pegawaitidaktetap) : "  Proses input variabel jabatan  Seleksi kondisi if jika jabatan.equalsIgnoreCase("pegawaitidaktetap") maka  Proses mencetak "Masukan Jam kerja : "  Proses input variabel kerja  Proses mencetak "Status pernikahan : "  Proses input variabel status  Proses mencetak "Jumlah anak : "  Proses input variabel anak  Proses mencetak "Masukan tahun bekerja : "  Proses input variabel tahun  berhenti  Seleksi kondisi jika user memilih 2  Seleksi kondisi if jika jabatan.equalsIgnoreCase("manager") maka  Membuat objek a dari class manager dengan mengirim nilai dari variabel nama, id, jabatan, status, anak, (int) tahun  Memanggil method a.setBonus()  Memanggil method a.setTunjangan()  Proses mencetak "Nama : " + a.nama  Proses mencetak "IdKerja : " + a.idKerja  Proses mencetak "Jabatan : " + a.jabatan  Proses mencetak "Status Pernikahan : " + a.status  Proses mencetak "Jumlah Anak : %1.0f ", a.anak  Proses mencetak “”  Proses mencetak "Tahun bekerja : %4.0f ", a.tahun  Proses mencetak “”  Proses mencetak "Gaji = Rp. %8.0f ", a.gaji  Proses mencetak “”  Seleksi kondisi if jika jabatan.equalsIgnoreCase(“pegawai")  Membuat objek b dari class PegawaiTetap dengan mengirim nilai dari variabel nama, id, jabatan, status, anak, (int) tahun  Memanggil method b.setBonus()  Memanggil method b.setGatot()  Proses mencetak "Nama : " + b.nama  Proses mencetak "IdKerja : " + b.idKerja  Proses mencetak "Jabatan : " + b.jabatan  Proses mencetak "Status Pernikahan : " + b.status  Proses mencetak "Jumlah Anak : %1.0f ", b.anak  Proses mencetak “”  Proses mencetak "Tahun bekerja : %4.0f ", b.tahun  Proses mencetak “”  Proses mencetak "Gaji = Rp. %8.0f ", b.gaji  Proses mencetak “”  Seleksi kondisi if jika jabatan.equalsIgnoreCase(“pegawaitidaktetap")  Membuat objek b dari class PegawaiTdkTetap dengan mengirim nilai dari variabel nama, id, jabatan, status, anak, (int) tahun  Memanggil method c.setBonus()  Memanggil method c.setLembur(kerja)  Proses mencetak "Nama : " + c.nama  Proses mencetak "IdKerja : " + c.idKerja  Proses mencetak "Jabatan : " + c.jabatan  Proses mencetak "Status Pernikahan : " + c.status  Proses mencetak "Jumlah Anak : %1.0f ", c.anak  Proses mencetak “”  Proses mencetak "Tahun bekerja : %4.0f ", c.tahun  Proses mencetak “”  Proses mencetak "Gaji = Rp. %8.0f ", c.gaji  Proses mencetak “”  berhenti  Seleksi kondisi jika user memilih 3  behenti  Kondisi perulangan do while akan berhenti jika user menginputkan variabel pilihan dengan nilai 3 |

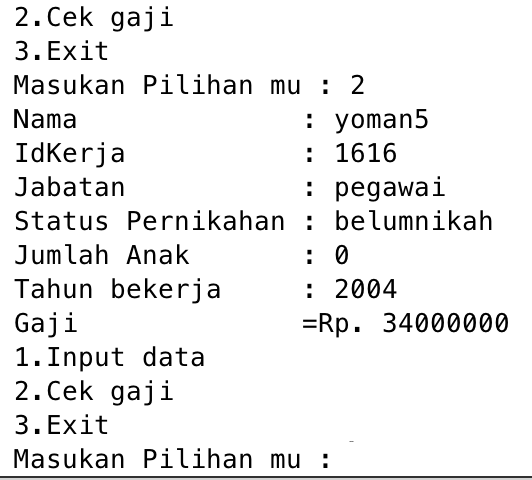
1. **SCREENSHOT PROGRAM**

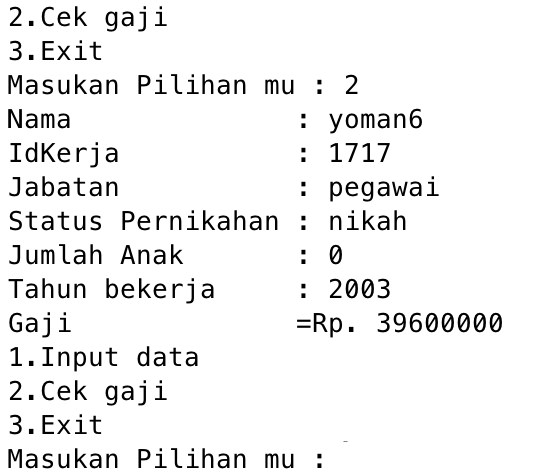
****

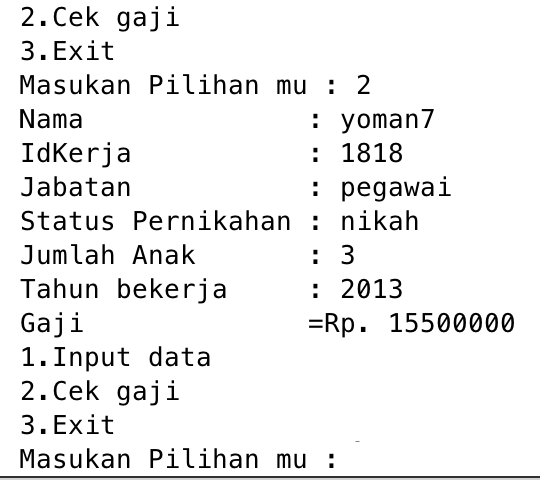
****

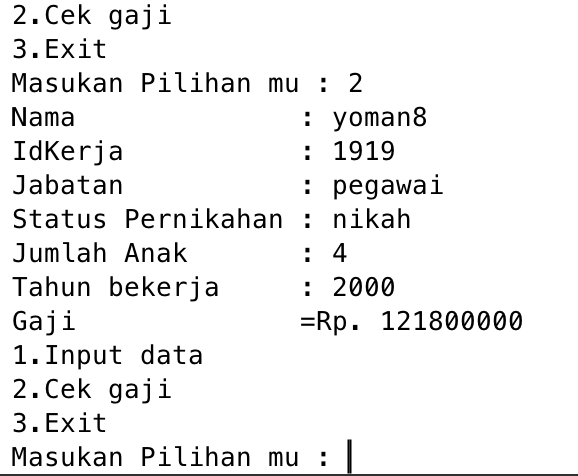
****

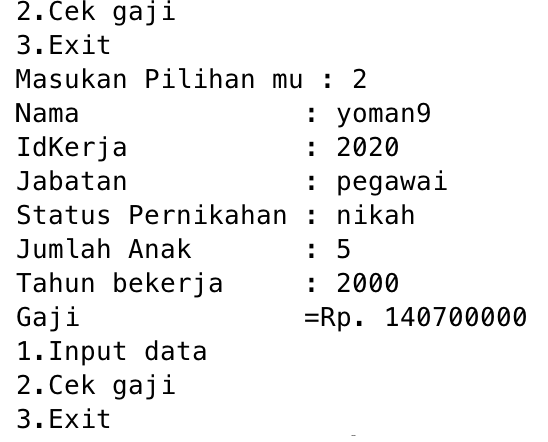
****

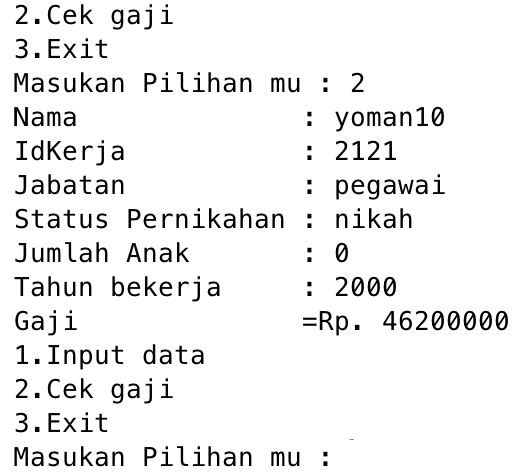
****

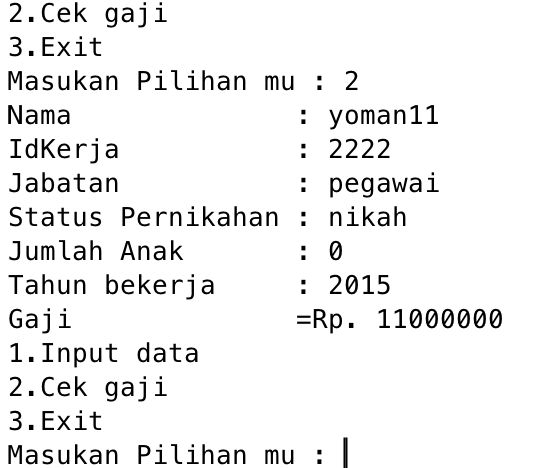
****

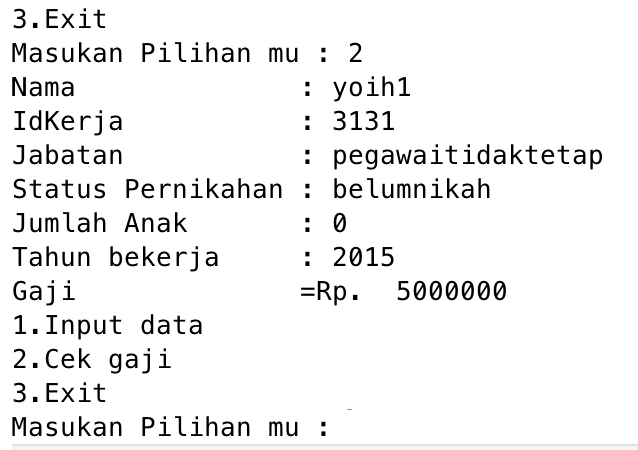
****

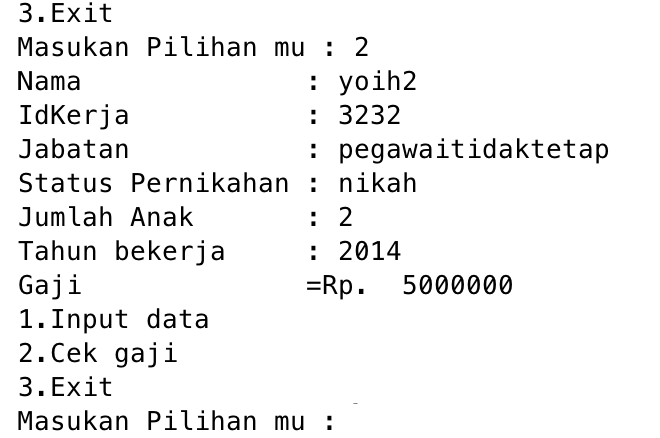
****

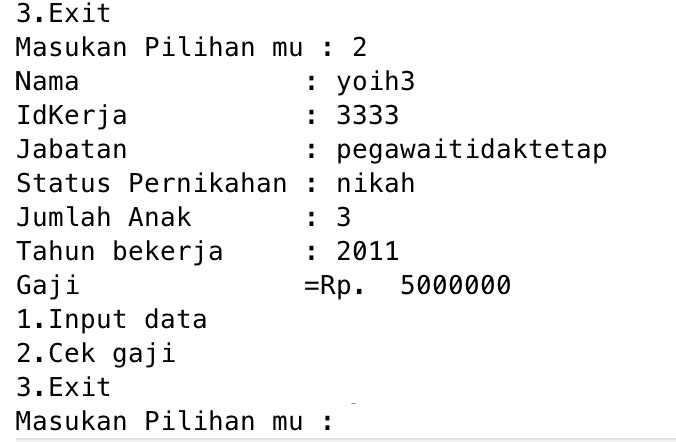
****

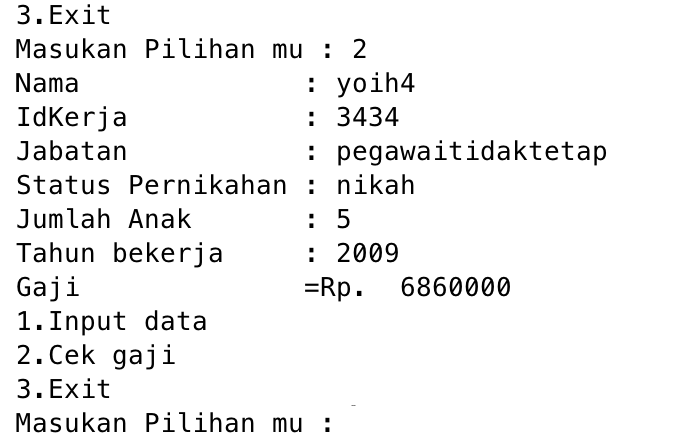
****

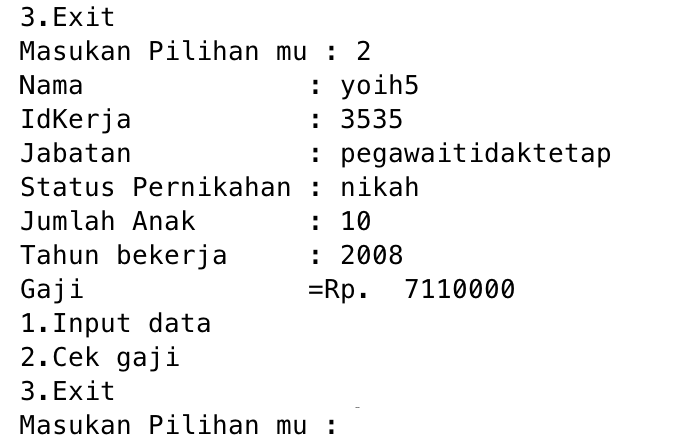
****

****

****

****

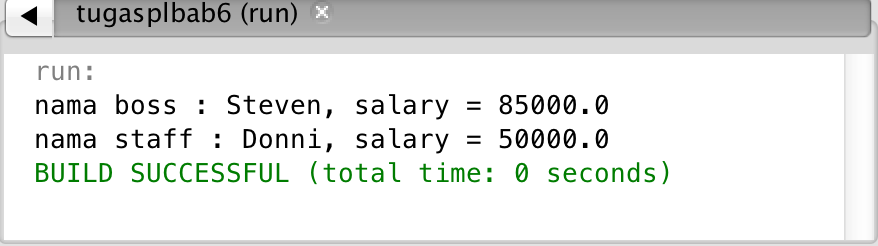
****

****

1. **PRAKTIKUM**

Pertanyaan

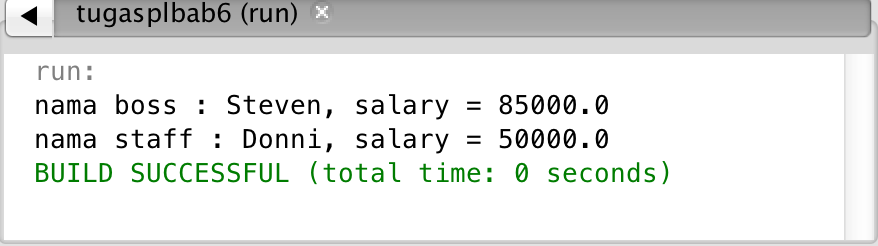
1. Jalankan code program diatas dan benahi jika menemukan kesalahan!

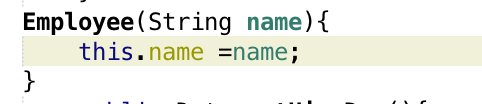


Tidak terdapat kesalahan pada code program

1. Bagaimana cara konstruktor pada subclass memanggil konstruktor di superclass nya? Apakah hal itu perlu dilakukan? Sertakan alasan anda !
2. Tambahkan constructor pada class Employee dengan parameter String name! amati perubahan apa yang terjadi, jelaskan jawaban anda!

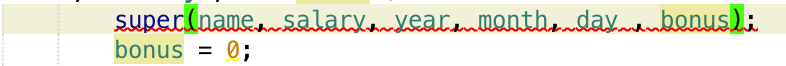
-Tidak terdapat kesalahan dan program masih dapat berjalan dengan baik dan tidak ada perubahan



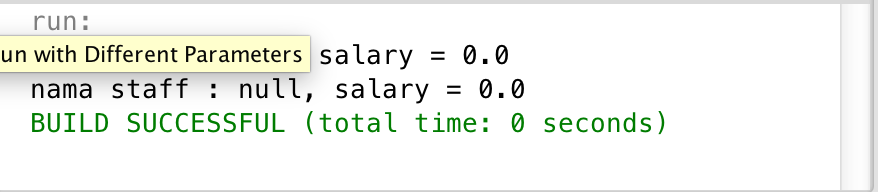


1. Pada Class Manager baris ke 5, setelah variable day tambahkan variable bonus! Amati apa yang terjadi dan mengapa demikian?

* terdapat error dikarenakan pada konstruktor superclass nya tidak terdapat variabel bonus



1. Untuk apa digunakan keyword this pada class manager dan employee? Hapus keyword this dan amati apa yang terjadi?



-keyword this digunakan untuk menginisialisasi variabel yg terdapat pada class tersebut

-beberapa variabel yang sebelumnya menggunakan keyword this tidak terinisialisasi

1. Tambahkan constructor pada class Employee dengan parameter Bertipe data string bernama name yang nantinya bila constructor ini akan dipanggil akan menginisialisasi variable name! Amati perubahannya pada class anak dan jelaskan! Benahi bila terjadi kesalahan!

-tidak terdapat perubahan

7.Pada bab sebelumnya anda telah belajar mengenai konsep encapsulation, jelaskan mengapa pada super class menggunakan modifier protected? Apa yang terjadi jika modifier anda ubah menjadi private atau public? Jelaskan !

-Agar dapat diakses pada subclass nya , jika diubah menjadi private tidak dapat diakses

8. Ubahlah acces modifier method pada kelas employee menjadi : a. a.Private

b. Protected Amati perubahan apa yang terjadi? Jelaskan jawaban anda dengan detail!

-Terdapat error dikarenakan method-method yang terdapat pada superclass tidak dapat diakses subclassnya

1. **KESIMPULAN**

Inheritance (Pewarisan) merupakan salah satu dari tiga konsep dasar OOP. Konsep inheritance ini mengadopsi dunia riil dimana suatu entitas/obyek dapat mempunyai entitas/obyek turunan. Dengan konsep inheritance, sebuah class dapat mempunyai class turunan.

Cara menerapkan keyword super adalah Dengan menambahkan kata kunci extends setelah deklarasi nama class, kemudian

diikuti dengan nama parent class-nya. Kata kunci extends tersebut memberitahu kompiler Java bahwa kita akan melakukan perluasan class, contoh deklarasi :

public class Programmer extends Pegawai { ... }

public class Manager extends Pegawai { ... }

Super adalah sebuah Java Keyword yang digunakan untuk mengakses anggota kelas diwariskan oleh kelas di mana ia muncul. Memungkinkan subclass untuk mengakses ditimpa metode dan anggota tersembunyi dari superclass nya. Keyword Super juga digunakan untuk meneruskan panggilan dari konstruktor ke konstruktor di superclass. Juga digunakan untuk menentukan batas bawah pada sebuah parameter ketik Generics.

Super.namamethod()

UML Adalah sebuah Bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun dan mendokumentasikan dari sebuah system pengembangan software OOP (Object Oriented Programming).