1. **DEFINISI MASALAH**
2. Pegawai.java Buatlah class induk Pegawai dengan definisi sebagai berikut

- nama: String

- noIndukPegawai: String

- tahunMasuk: Date

- gajiPokok: double

- keluarga: boolean

- jumlahAnak: int

+ double getGajiPokok()

+ double getBonus()

+ double getTunjangan()

+ double getTotalGaji()

a. Bonus didapatkan dari lama kerja (tahun sekarang dikurangi tahun masuk)

- Jika masa kerja dibawah 1 tahun tidak mendapat bonus

- Jika masa kerja 1-5 tahun mendapat bonus 5% x gajiPokok x lama kerja

- Jika masa kerja diatas 5 tahun mendapat bonus 10% x gajiPokok x lama kerja

b. Tunjangan didapatkan bila pegawai sudah berkeluarga

- Jika sudah berkeluarga, mendapat tunjangan 5% dari gaji pokok

- Jika memiliki anak mendapat tunjangan 5% dari gaji pokok per Anak yang dimiliki (maksimal 2 anak)

c. Total gaji diperoleh dari gaji pokok + bonus + tunjangan

1. Manager.java Buatlah class Manager yang inherit class Pegawai dengan ketentuan Manager mendapat tunjangan jabatan sebesar 10% dari total gaji
2. Sales.java Buatlah class Sales yang inherit class Pegawai dengan definisi:

- hargaBarang: double

- stockBarang: int

- barangTerjual: int

+ double getBonusTambahan()

Sales memiliki bonusTambahan dengan ketentuan jika barangTerjual lebih dari 70% stockBarang, maka mendapat bonusTambahan 10% dari hargaBarang x barangTerjual Jika tidak, maka bonusTambahan yang diperoleh hanya 3% dari hargaBarang x barangTerjual

1. Programmer.java Buatlah class Programmer yang inherit class Pegawai dengan definisi

- jamKerja: int

+ double getBonusLembur()

Freelancer akan mendapat bonusLembur jika jamKerja lebih dari 10 jam, sejumlah 5% x gajiPokok x jamKerja - 10 jam.

Test Case:

1. Pegawai tahunMasuk 2015, berkeluarga, 2 anak, gajiPokok 3jt

2. Pegawai tahunMasuk 2010, tidak berkeluarga, gajiPokok 3jt

3. Pegawai tahunMasuk 2005, berkeluarga, 5 anak, gajiPokok 3jt

4. Manager tahunMasuk 2009, tidak berkeluarga, gajiPokok 4jt

5. Manager tahunMasuk 2014, berkeluarga, 2 anak, gajiPokok 4jt

6. Sales tahunMasuk 2012, berkeluarga, 4 anak, hargaBarang 25rb, stockBarang 100, barangTerjual 50, gajiPokok 2jt

7. Sales tahunMasuk 2014, berkeluarga, 1 anak, hargaBarang 10rb, stockBarang 250, barangTerjual 220, gajiPokok 2jt

8. Freelancer tahunMasuk 2011, tidak berkeluarga, jamKerja 10 jam, gajiPokok 2,5jt

9. Freelancer tahunMasuk 2013, tidak berkeluarga, jamKerja 12 jam, gajiPokok 2,5jt

Tampilkan totalGaji berserta perhitungannya (gajipokok, bonus, tunjangan) dari setiap Test Case

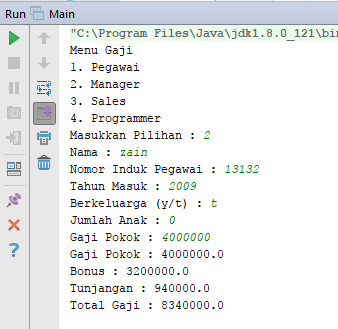
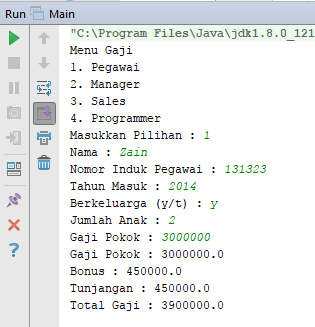
1. **SOURCE CODE**

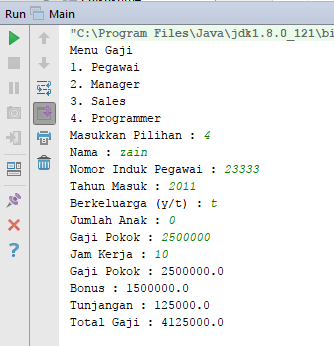
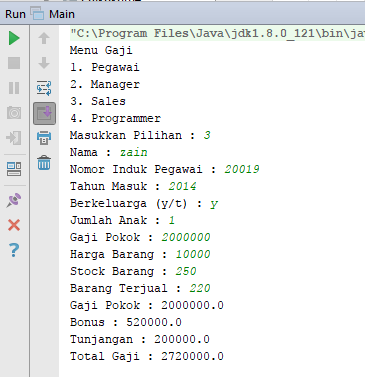
|  |  |
| --- | --- |
|  | Pegawai.java |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81 | import java.util.Calendar;  import java.util.Date;  /\*\*  \* Created by Zain Fikri H on 28/04/2017.  \*/  public class Pegawai {  private String nama;  private String noIndukPegawai;  private Date tahunMasuk;  private static double gajiPokok;  private boolean keluarga=true;  private static int jumlahAnak;  private static double tunjangan;  private static double bonus;  public String getNama() {  return nama;  }  public String getNoIndukPegawai() {  return noIndukPegawai;  }  public double getGajiPokok() {  return gajiPokok;  }  public boolean getKeluarga() {  return keluarga;  }  public int getJumlahAnak() {  return jumlahAnak;  }  public void setJumlahAnak(int jumlahAnak) {  this.jumlahAnak = jumlahAnak;  }  public Pegawai(String nama, String noIndukPegawai, int tahunMasuk, double gajiPokok, boolean keluarga) {  this.nama = nama;  this.noIndukPegawai = noIndukPegawai;  Date tahun = new Date();  tahun.setYear(tahunMasuk);  this.tahunMasuk = tahun;  this.gajiPokok = gajiPokok;  this.keluarga = keluarga;  }  public double setTunjangan(){  double tu=0;  if(getKeluarga()==true){  tu = 0.05\*getGajiPokok();  }if(jumlahAnak==1 || jumlahAnak==2){  tu += tu\*jumlahAnak;  }return tunjangan=tu;  }  public double setBonus(){  Calendar now = Calendar.getInstance();  double lamaKerja = now.get(Calendar.YEAR)-tahunMasuk.getYear();  double tu=0;  if(lamaKerja<1){  tu = 0;  }if(lamaKerja>=1 && lamaKerja<=5){  tu = 0.05\*gajiPokok\*lamaKerja;  }if(lamaKerja>5){  tu = 0.1\*gajiPokok\*lamaKerja;  }  return bonus=tu;  }  public double getTotalGaji(){  double totalGaji;  totalGaji = gajiPokok + setBonus() + setTunjangan();  return totalGaji;  }  } |
|  | Manager.java |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | /\*\*  \* Created by Zain Fikri H on 28/04/2017.  \*/  public class Manager extends Pegawai {  public Manager(String nama, String noIndukPegawai, int tahunMasuk, double gajiPokok, boolean keluarga) {  super(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga);  }  public double getTunjangan(){  double tunjangan = super.setTunjangan();  tunjangan = tunjangan + (0.1\*super.getTotalGaji());  return tunjangan;  }  public double getTotalGaji(){  double totalGaji = super.getTotalGaji();  return totalGaji+getTunjangan();  }  } |
|  | Sales.java |
| 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 | /\*\*  \* Created by Zain Fikri H on 28/04/2017.  \*/  public class Sales extends Pegawai {  private double hargaBarang;  private int stockBarang;  private int barangTerjual;  public Sales(String nama, String noIndukPegawai, int tahunMasuk, double gajiPokok, boolean keluarga, double hargaBarang, int stockBarang, int barangTerjual) {  super(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga);  this.hargaBarang = hargaBarang;  this.stockBarang = stockBarang;  this.barangTerjual = barangTerjual;  }  public double getBonusTambahan(){  double bonusTambahan;  if(barangTerjual>(0.7\*stockBarang)){  bonusTambahan = 0.1 \* (hargaBarang\*barangTerjual);  }else{  bonusTambahan = 0.03 \* (hargaBarang\*barangTerjual);  }return bonusTambahan;  }  public double getBonus(){  double bonus = super.setBonus();  return bonus+getBonusTambahan();  }  public double getTotalGaji(){  double totalGaji = super.getTotalGaji();  return totalGaji+getBonusTambahan();  }  } |
|  | Programmer.java |
| 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 | /\*\*  \* Created by Zain Fikri H on 28/04/2017.  \*/  public class Programmer extends Pegawai {  private int jamKKerja;  public Programmer(String nama, String noIndukPegawai, int tahunMasuk, double gajiPokok, boolean keluarga, int jamKKerja) {  super(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga);  this.jamKKerja = jamKKerja;  }  public double getBonusLembur(){  double bonusLembur = 0;  if(jamKKerja>10) {  bonusLembur = 0.05 \* getGajiPokok() \* (jamKKerja - 10);  }return bonusLembur;  }  public double setBonus(){  double bonus = super.setBonus();  return bonus+getBonusLembur();  }  public double getTotalGaji(){  double totalGaji = super.getTotalGaji();  return totalGaji+getBonusLembur();  }  } |
|  | Main.java |
| 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 | import java.util.Scanner;  /\*\*  \* Created by Zain Fikri H on 28/04/2017.  \*/  public class Main {  public static void main(String[] args) {  Scanner in = new Scanner(System.in);  boolean keluarga=true;  int anak=0;  System.out.println("Menu Gaji");  System.out.println("1. Pegawai");  System.out.println("2. Manager");  System.out.println("3. Sales");  System.out.println("4. Programmer");  System.out.print("Masukkan Pilihan : ");  int pilihan = in.nextInt();  in.nextLine();  System.out.print("Nama : ");  String nama = in.nextLine();  System.out.print("Nomor Induk Pegawai : ");  String noIndukPegawai = in.next();  System.out.print("Tahun Masuk : ");  int tahunMasuk = in.nextInt();  System.out.print("Berkeluarga (y/t) : ");  String berkeluarga = in.next();  if(berkeluarga=="y"){  keluarga=true;  }if(berkeluarga=="t"){  keluarga=false;  }System.out.print("Jumlah Anak : ");  anak = in.nextInt();  System.out.print("Gaji Pokok : ");  double gajiPokok = in.nextDouble();  switch (pilihan){  case 1 :  Pegawai pegawai = new Pegawai(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga);  pegawai.setJumlahAnak(anak);  pegawai.setBonus();  pegawai.setTunjangan();  System.out.println("Gaji Pokok : "+pegawai.getGajiPokok());  System.out.println("Bonus : "+pegawai.setBonus());  System.out.println("Tunjangan : "+pegawai.setTunjangan());  System.out.println("Total Gaji : "+pegawai.getTotalGaji());  break;  case 2 :  Manager manager = new Manager(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga);  manager.setJumlahAnak(anak);  manager.setBonus();  manager.setTunjangan();  System.out.println("Gaji Pokok : "+manager.getGajiPokok());  System.out.println("Bonus : "+manager.setBonus());  System.out.println("Tunjangan : "+manager.getTunjangan());  System.out.println("Total Gaji : "+manager.getTotalGaji());  break;  case 3 :  System.out.print("Harga Barang : ");  double harga = in.nextDouble();  System.out.print("Stock Barang : ");  int stock = in.nextInt();  System.out.print("Barang Terjual : ");  int barang = in.nextInt();  Sales sales = new Sales(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga, harga, stock, barang);  sales.setJumlahAnak(anak);  sales.setBonus();  sales.setTunjangan();  System.out.println("Gaji Pokok : "+sales.getGajiPokok());  System.out.println("Bonus : "+sales.getBonus());  System.out.println("Tunjangan : "+sales.setTunjangan());  System.out.println("Total Gaji : "+sales.getTotalGaji());  break;  case 4 :  System.out.print("Jam Kerja : ");  int jamKerja = in.nextInt();  Programmer programmer = new Programmer(nama, noIndukPegawai, tahunMasuk, gajiPokok, keluarga, jamKerja);  programmer.setJumlahAnak(anak);  programmer.setBonus();  programmer.setTunjangan();  System.out.println("Gaji Pokok : "+programmer.getGajiPokok());  System.out.println("Bonus : "+programmer.setBonus());  System.out.println("Tunjangan : "+programmer.setTunjangan());  System.out.println("Total Gaji : "+programmer.getTotalGaji());  break;  default :  break;  }  }  } |

1. **PEMBAHASAN**

|  |  |
| --- | --- |
|  | Pegawai.java |
| 1  2  7  8  9  10  11  12  13  14  15  17  18  21  22  25  26  29  30  33  34  37  38  41  43  44  45  46  47  48  49  60  61  62  64  65  66  67  68  69  70  72  75  76  77  79 | Import Scanner  Import Date  Deklarasi class Pegawai  Deklarasi attribute String nama dengan modifier private  Deklarasi attribute String noIndukPegawai dengan modifier private  Deklarasi attribute Date tahunMasuk dengan modifier private  Deklarasi attribute static double gajiPokok dengan modifier private  Deklarasi attribute Boolean keluarga dengan modifier private  Deklarasi attribute static int jumlahAnak dengan modifier private  Deklarasi attribute static double tunjangan dengan modifier private  Deklarasi attribute static double bonus dengan modifier private  Deklarasi method getNama  Statement return nama  Deklarasi method getNoIndukPegawai  Statement return noIndukPegawai  Deklarasi method getGajiPokok  Statement return gajiPokok  Deklarasi method getKeluarga  Statement return keluarga  Deklarasi method getJumlahAnak  Statement return jumlahAnak  Deklarasi method setJumlahAnak berparameter  Attribute jumlahAnak sama dengan parameter jumlahAnak  Deklarasi constructor Pegawai berparameter  Attribute nama sama dengan parameter nama  Attribute noIndukPegawai sama dengan parameter noIndukPegawai  Deklarasi Date tahun  Statement setYear bervalue parameter tahunMasuk  Attribute tahunMasuk sama dengan parameter tahun  Attribute gajiPokok sama dengan parameter gajiPokok  Attribute keluarga sama dengan parameter keluarga  Deklarasi method setBonus  Instansiasi now bertipe data Calendar  Deklarasi lamaKerja dengan value now.get tahun sekarang dikurangi attribute tahunMasuk  Deklarasi dan inisialisasi variabel tu bertipe double  Seleksi kondisi if dengan kondisi lamaKerja kurang dari 1  Variabel tu sama dengan 0  Seleksi kondisi if dengan kondisi lamaKerja lebih sama dengan 1 dan kurang sama dengan 5  Variabel tu sama dengan 0,05 dikali gajiPokok dikali lamaKerja  Seleksi kondisi if dengan kondisi lamaKerja lebih dari 5  Variabel tu sama dengan 0,1 dikali gajiPokok dikali lamaKerja  Statement return bonus sama dengan variabel tu  Deklarasi method getTotalGaji  Deklarasi variabel totalGaji bertipe double  Variabel total gaji sama dengan gajipokok ditambah method setBonus dan ditambah setTunjangan  Statement return totalGajis |
|  | Manager.java |
| 4  5  7  11  12  13  15  17  18  19 | Deklarasi class Manager sebagai subclass Pegawai  Deklarasi constructor Manager berparameter  Statement super untuk memanggil class super dan memeasukan parameter constructor  Deklarasi method setTunjangan  Deklarasi dan inisialisasi variabel tunjangan sama dengan setTunjangan pada super  Variabel tunjangan ditambah tunjangan ditambah 0,1 dikali getTotalGaji  Statement return tunjangan  Deklarasi method getTotalGaji  Deklarasi dan inisialisasi dijumlah variabel tunjangan sama dengan getTotalGaji pada super  Statement return totalGaji ditambah setTunjangan |
|  | Sales.java |
| 4  5  6  7  8  11  13  14  15  18  19  20  21  23  24  26  29  30  31  33  34  35 | Deklarasi class Sales sebagai subclass Pegawai  Deklarasi attribute double hargaBarang bermodifier private  Deklarasi attribute int stockBarang bermodifier private  Deklarasi attribute int barangTerjual bermodifier private  Deklarasi constructor berparameter  Statement super untuk memanggil class super dan memeasukan parameter constructor  Attribute hargaBarang sama dengan parameter hargaBarang  Attribute stockBarang sama dengan parameter stockBarang  Attribute barangTerjual sama dengan parameter barangTerjual  Deklarasi method getBonusTambahan  Deklarasi double bonusTambahan  Seleksi kondisi if dengan kondisi barang terjual lebih besar dari 0,7 dikali stockBarang  Variabel bonusTambahan sama dengan 0,1 dikali hargaBarang dikali barangTerjual  Seleksi kondisi else  Variabel bonusTambahan sama dengan 0,03 dikali hargaBarang dikali barangTerjual  Statement return bonusTambahan  Deklarasi method getBonus  Deklarasi dan inisialiasi variabel bonus sama denga nilai setBonus pada class super  Statement return bonus ditambah value getBonusTambahan  Deklarasi method getTotalGaji  Deklarasi dan inisialiasi variabel totalGaji sama denga nilai getTotalGaji pada class super  Statement return bonus ditambah value getBonusTambahan |
|  | Programmer.java |
| 4  5  6  9  11  14  15  16  17  19  22  23  24  27  28  29 | Deklarasi class Programmer sebagai subclass Pegawai  Deklarasi attribute int jamKerja bermodifier private  Deklarasi constructor berparameter  Statement super untuk memanggil class super dan memeasukan parameter constructor  Attribute jamKerja sama dengan parameter jamKerja  Deklarasi method getBonusLembur  Deklarasi dan inisialisasi variabel bonusLembur sama dengan 0  Seleksi kondisi if dengan kondisi jamKerja lebih dari 10  Variabel bonusLembur sama dengan 0,05 dikali getGajipokok dan dikali jamkerja dikurangi 10  Statement return bonusLembur  Deklarasi method setBonus  Deklarasi dan inisialisasi variabel bonus sama dengan setBonus pada class super  Statement return bonus ditambah getBonusLembur  Deklarasi method getTotalGaji  Deklarasi dan inisialisasi variabel totalGaji sama dengan getTotalGaji class super  Statement return totalGaji ditambah getBonusLembur |
|  | Main.java |
| 1  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  39  40  41  42  44  46  48  50  51  52  54  55  56  57  59  61  63  65  66  67  68  69  70  71  72  73  76  77  78  79  81  83  85  87  88  89  90  91  94  95  96  97  99  101  103  105  106  107 | Import Scanner  Deklarasi class Main  Deklarasi main method  Instansiasi Scanner  Deklarasi dan inisialisasi variabel keluarga  Deklarasi dan inisialisasi variabel anak  Print “Menu Gaji”  Print “1. Pegawai”  Print “2. Manager”  Print “3. Sales”  Print “4. Programmer”  Print “Masukkan Piliha : “  Deklarasi variabel pilihan dan input user  Scanner untuk input user  Print “Nama : “  Deklarasi variabel nama dan input user  Print “Nomor Induk Pegawai : “  Deklarasi variabel noIndukPegawai dan input user  Print “Tahun Masuk : “  Deklarasi variabel tahunMasuk dan input user  Print “Berkeluarga (y/t) : “  Deklarasi variabel berkeluarga dan input user  Seleksi kondisi if dengan kondisi berkeluarga sama dengan ‘y’  Variabel keluarga sama dengan true  Seleksi kondisi if dengan kondisi berkeluarga sama dengan ‘t’  Variabel berkeluarga sama dengan false  Print “Jumlah Anak : “  Variabel anak input user  Print “Gaji Pokok : “  Deklarasi variabel gajiPokok dan input user  Seleksi kondisi dengan kondisi sesuai kondisi pilihan  Case 1  Instansiasi class Pegawai  Pemanggilan setJumlahAnak pada class pegawai  Pemanggilan setBonus pada class pegawai  Pemanggilan setTunjangan pada class pegawai  Print “Gaji Pokok : “ dan pemanggilan getGajiPokok pada class pegawai  Print “Bonus : “ dan pemanggilan setBonus pada class pegawai  Print “Tunjangan : “ dan pemanggilan setTunjangan pada class pegawai  Print “Total Gaji : “ dan pemanggilan getTotalGaji pada class pegawai  Statement break  Case 2  Instansiasi class Manager  Pemanggilan setJumlahAnak pada class manager  Pemanggilan setBonus pada class manager  Pemanggilan setTunjangan pada class manager  Print “Gaji Pokok : “ dan pemanggilan getGajiPokok pada class manager  Print “Bonus : “ dan pemanggilan setBonus pada class manager  Print “Tunjangan : “ dan pemanggilan setTunjangan pada class manager  Print “Total Gaji : “ dan pemanggilan getTotalGaji pada class manager  Statement break  Case 3  Print “Harga Barang : “  Deklarasi variabel harga dan input user  Print “Stock Barang : “  Deklarasi variabel stock dan input user  Print “Barang Terjual : “  Deklarasi variabel baramg dan input user  Instansiasi class Sales  Pemanggilan setJumlahAnak pada class sales  Pemanggilan setBonus pada class sales  Pemanggilan setTunjangan pada class sales  Print “Gaji Pokok : “ dan pemanggilan getGajiPokok pada class sales  Print “Bonus : “ dan pemanggilan getBonus pada class sales  Print “Tunjangan : “ dan pemanggilan setTunjangan pada class sales  Print “Total Gaji : “ dan pemanggilan getTotalGaji pada class sales  Statement break  Case 4  Print “Jam Kerja : “  Deklarasi variabel jamKerja dan input user  Instansiasi class Programmer  Pemanggilan setJumlahAnak pada class programmer  Pemanggilan setBonus pada class programmer  Pemanggilan setTunjangan pada class programmer  Print “Gaji Pokok : “ dan pemanggilan getGajiPokok pada class programmer  Print “Bonus : “ dan pemanggilan setBonus pada class programmer  Print “Tunjangan : “ dan pemanggilan setTunjangan pada class programmer  Print “Total Gaji : “ dan pemanggilan getTotalGaji pada class programmer  Statement break  Case default  Statement break |

1. **SCREENSHOT PROGRAM**

****

****

1. **PRAKTIKUM**
2. **Statis Method**
3. Jalankan code program diatas dan benahi jika menemukan kesalahan!

*Jawaban:*

Tidak ada kesalahan dalam program tersebut.

1. Bagaimana cara konstruktor pada subclass memanggil konstruktor di superclass nya? Apakah hal itu perlu dilakukan? Sertakan alasan anda !

*Jawaban:*

Caranya adalah dengan menggunakan keywords atau statement ‘super’ serta diikuti dengan parameternya. Hal tersebut perlu dilakukan karena kalau tidak maka subclass tidak akan bias memanggil class supernya.

1. Tambahkan constructor pada class Employee dengan parameter String name! amati perubahan apa yang terjadi, jelaskan jawaban anda!

*Jawaban:*Tidak ada yang error dan tidak ada perubahan. Karena tidak mengubah strukturnya yang sebelumnya sudah terdapat constractor juga.

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

*Jawaban:*

Terjadi error. Karena pada class super constractornya tidak terdapat parameter bonus.

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

*Jawaban:*

Keyboard this digunakan untuk memberikan indikasi bahwa variabel terrsebut menunjuk ke variabel classnya. Kalau dihapus maka tidak akan mengubah nilai dari variabel classnya.

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!

*Jawaban:*

Tidak ada yang error dan tidak ada perubahan. Karena tidak mengubah strukturnya yang sebelumnya sudah terdapat constractor juga.

1. 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 !

*Jawaban:*

Jika modifier diubah ke protected maka class super tersebut dabat diakses subclassnya namun tidak dapat diakses oleh class diluarnya. Kalau di public maka dapat diakses oleh semua class dan jika private maka tidak dapat diakses class lain.

1. Ubahlah access modifier method padakelas employee menjadi :
2. Private
3. Protected

Amati perubahanapa yang terjadi? Jelaskanjawabanandadengan detail

*Jawaban:*

* 1. Jika modifier method diubah ke private maka terjadi error, karena hanya dapat diakses oleh class tersebut.
  2. Jika procteted maka method tersebut dapat diakses oleh subclassnya atau class turunananya.

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 penerapan konsep inheritance 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.
* Keyword super merupakan keyword yang digunakan untuk pemanggilan parent class pada subclass. Pemanggilan ini berupa pemanggilan method maupun variabel yang ada dalam parent class.
* Contoh diagram UML pada konsep inheritance:

+bonus

Programmer

Manager

+tunjangan

+nama

+noPegawai

+gaji

Pegawai