

Jobsheet 4 – Class Relation
Praktikum Pemrograman Berbasis Objek



244107060144

Maria Savira

D-IV Sistem Informasi Bisnis / 2C

Praktikum

Konsultasi.java

Pegawai.java Pasien.java RumahSakitDemo.java Konsultasi.java X

RumahSakit > Konsultasi.java > Language Support for Java(TM) by Red Hat > Konsultasi

```
1 package RumahSakit;
2
3 import java.time.LocalDate;
4
5 public class Konsultasi {
6     private LocalDate tanggal;
7     private Pegawai dokter;
8     private Pegawai perawat;
9
10    public LocalDate getTanggal(){
11        return tanggal;
12    }
13
14    public void setTanggal(LocalDate tanggal){
15        this.tanggal = tanggal;
16    }
17
18    public Pegawai getDokter(){
19        return dokter;
20    }
21
22    public void setDokter(Pegawai dokter){
23        this.dokter = dokter;
24    }
25
26    public Pegawai getPerawat(){
27        return perawat;
28    }
29
30    public void setPerawat(Pegawai perawat){
31        this.perawat = perawat;
32    }
33 }
```

Pegawai.java Pasien.java RumahSakitDemo.java Konsultasi.java X

RumahSakit > Konsultasi.java > ...

```
5 public class Konsultasi {
34     public String getInfo(){
35         String info = "";
36         info += "\tTanggal : " + tanggal;
37         info += ", Dokter : " + dokter.getInfo();
38         info += ", Perawat : " + perawat.getInfo();
39         info += "\n";
40
41         return info;
42     }
43 }
```

Pasien.java

Pegawai.java Pasien.java 2 X RumahSakitDemo.java Konsultasi.java

RumahSakit > Pasien.java > Language Support for Java(TM) by Red Hat > Pasien

```
1 package RumahSakit;
2
3 import java.time.LocalDate;
4 import java.util.ArrayList;
5
6 public class Pasien {
7     private String noRekamMedis;
8     private String nama;
9     private ArrayList<Konsultasi> riwayatKonsultasi;
10
11     public String getNoRekamMedis(){
12         return noRekamMedis;
13     }
14
15     public void setNoRekamMedis(String noRekamMedis){
16         this.noRekamMedis = noRekamMedis;
17     }
18
19     public String getNama(){
20         return nama;
21     }
22
23     public void setNama(String nama){
24         this.nama = nama;
25     }
26
27     public Pasien(String noRekamMedis, String nama){
28         this.noRekamMedis = noRekamMedis;
29         this.nama = nama;
30         this.riwayatKonsultasi = new ArrayList<Konsultasi>();
31     }
```

Pegawai.java Pasien.java 2 X RumahSakitDemo.java Konsultasi.java

RumahSakit > Pasien.java > Language Support for Java(TM) by Red Hat > Pasien > setNama(String)




```
6 public class Pasien {
33     public String getInfo(){
34         String info = "";
35         info += "No Rekam Medis : " + this.noRekamMedis + "\n";
36         info += "Nama : " + this.nama + "\n";
37
38         if (!riwayatKonsultasi.isEmpty()){
39             info += "Riwayat Konsultasi : \n";
40
41             for (Konsultasi konsultasi : riwayatKonsultasi){
42                 info += konsultasi.getInfo();
43             }
44         }
```

```

45     else {
46         info += "Belum ada riwayat konsultasi";
47     }
48
49     info += "\n";
50
51     return info;
52 }
53
54 public void tambahKonsultasi(LocalDate tanggal, Pegawai dokter, Pegawai perawat){
55     Konsultasi konsultasi = new Konsultasi();
56     konsultasi.setTanggal(tanggal);
57     konsultasi.setDokter(dokter);
58     konsultasi.setPerawat(perawat);
59     riwayatKonsultasi.add(konsultasi);
60 }
61 }

```

Pegawai.java

 Pegawai.java X
  Pasien.java 2
  RumahSakitDemo.java

RumahSakit >  Pegawai.java > ...

```

1  package RumahSakit;
2
3  public class Pegawai{
4      private String nip;
5      private String nama;
6
7      public Pegawai (String nip, String nama){
8          this.nip = nip;
9          this.nama = nama;
10     }
11
12     public String getNip(){
13         return nip;
14     }
15
16     public void setNip(String nip){
17         this.nip = nip;
18     }
19
20     public String getNama(){
21         return nama;
22     }
23
24     public void setNama(String nama){
25         this.nama = nama;
26     }
27
28     public String getInfo(){
29         return nama + " (" + nip + ") ";
30     }
31 }

```

RumahSakitDemo.java

```
Pegawai.java Pasien.java 2 RumahSakitDemo.java X Konsultasi.java
RumahSakit > RumahSakitDemo.java > ...
1 package RumahSakit;
2
3 import java.time.LocalDate;
4
5 public class RumahSakitDemo {
    Run | Debug | Run main | Debug main
6     public static void main(String[] args) {
7         Pegawai ani = new Pegawai(nip:"1234", nama:"dr. Ani");
8         Pegawai bagus = new Pegawai(nip:"4567", nama:"dr. Bagus");
9
10        Pegawai desi = new Pegawai(nip:"1234", nama:"Ns. Desi");
11        Pegawai eka = new Pegawai(nip:"4567", nama:"Ns. Eka");
12
13        Pasien pasien1 = new Pasien(noRekamMedis:"343298", nama:"Puspa Widya");
14        pasien1.tambahKonsultasi(LocalDate.of(year:2021, month:8, dayOfMonth:11), ani, desi);
15        pasien1.tambahKonsultasi(LocalDate.of(year:2021, month:9, dayOfMonth:11), bagus, eka);
16
17        System.out.println(pasien1.getInfo());
18
19        Pasien pasien2 = new Pasien(noRekamMedis:"997744", nama:"Yenny Anggraeni");
20        System.out.println(pasien2.getInfo());
21    }
22 }
```

Output

```
No Rekam Medis : 343298
Nama : Puspa Widya
Riwayat Konsultasi :
    Tanggal : 2021-08-11, Dokter : dr. Ani (1234) , Perawat : Ns. Desi (1234)
    Tanggal : 2021-09-11, Dokter : dr. Bagus (4567) , Perawat : Ns. Eka (4567)

No Rekam Medis : 997744
Nama : Yenny Anggraeni
Belum ada riwayat konsultasi
```

Pertanyaan

Berdasarkan percobaan 1, jawablah pertanyaan-pertanyaan yang terkait:

1. Di dalam *class* Pegawai, Pasien, dan Konsultasi, terdapat method *setter* dan *getter* untuk masing-masing atributnya. Apakah gunanya *method setter* dan *getter* tersebut?
2. Di dalam *class* Konsultasi tidak secara eksplisit terdapat constructor dengan parameter. Apakah ini berarti *class* Konsultasi tidak memiliki constructor?
3. Perhatikan *class* Konsultasi, atribut mana saja yang bertipe *object*?
4. Perhatikan *class* Konsultasi, pada baris manakah yang menunjukkan bahwa *class* Konsultasi memiliki relasi dengan *class* Pegawai?
5. Perhatikan pada *class* Pasien, apa yang dilakukan oleh kode `konsultasi.getInfo()`?

6. Pada method `getInfo()` dalam class Pasien, terdapat baris kode:
`if (!riwayatKonsultasi.isEmpty())`
Apakah yang dilakukan oleh baris tersebut?
7. Pada constructor class Pasien, terdapat baris kode:
`this.riwayatKonsultasi = new ArrayList<>();`
Apakah yang dilakukan oleh baris tersebut? Apakah yang terjadi jika baris tersebut dihilangkan?

Jawaban

1. Method setter untuk memberikan nilai pada atribut private, sedangkan method getter untuk mengambil value nilainya dari class lain.
2. Class yang constructornya tidak disebutkan secara eksplisit bukan berarti tidak memiliki constructor. Karena semua class memiliki constructor secara default saat class tersebut dibuat.
3. Pada class Konsultasi, semua atributnya memiliki tipe data Object. `LocalDate` merupakan bagian dari package `java.time` yaitu class `LocalDate`. Sehingga atribut tanggal mereferensikan objek `LocalDate`. Begitupula dengan atribut dokter dan perawat yang mereferensikan objek `Pegawai`.
4. Dimulai dari baris berikut :

```
public class Konsultasi {  
    private LocalDate tanggal;  
    private Pegawai dokter;  
    private Pegawai perawat;
```

Sejak diinisiasikan atribut dokter dan perawat yang mereferensikan kepada objek `Pegawai`.
Lalu dilanjut pada baris berikut :

```
    public Pegawai getDokter(){  
        return dokter;  
    }  
  
    public void setDokter(Pegawai dokter){  
        this.dokter = dokter;  
    }  
  
    public Pegawai getPerawat(){  
        return perawat;  
    }  
  
    public void setPerawat(Pegawai perawat){  
        this.perawat = perawat;  
    }
```

Dengan setter dan getter untuk perawat dan dokter agar objek `Pegawai` yang baru dibuat dapat diidentifikasi dengan tepat.

5. Pada class Pasien, konsultasi.getInfo merupakan kode program untuk memanggil setiap objek konsultasi yang berada dalam ArrayList<Konsultasi> riwayatKonsultasi yang dimiliki oleh setiap pasien melalui perulangan foreach.
6. Kode program tersebut bermaksud sebagai validasi, untuk mengecek apakah riwayatKonsultasi dari pasien terkait kosong atau tidak. Apabila tidak kosong, maka syarat if terpenuhi. Lalu setiap data riwayatKonsultasi pasien akan ditampilkan. Namun jika kosong, maka akan muncul pemberitahuan bahwa pasien belum melakukan konsultasi.
7. Jika baris tersebut dihilangkan, maka pasien tidak akan memiliki riwayatKonsultasi. Lalu seluruh objek konsultasi tidak akan terkait kepada pasien siapapun. Karena arrayList tersebut digunakan agar riwayatKonsultasi terkait dengan salah satu pasien.

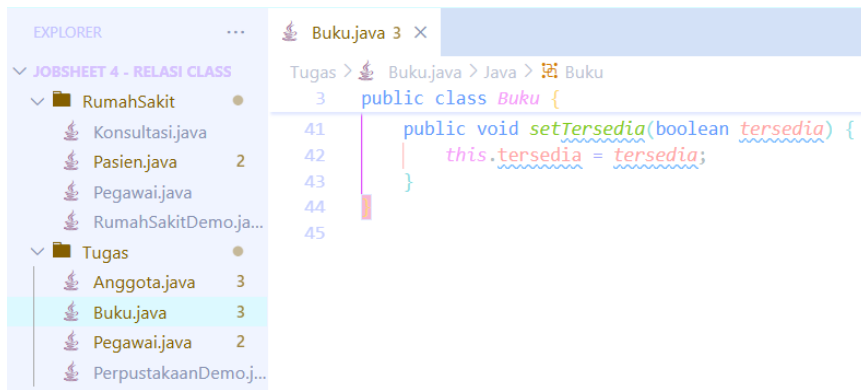
Tugas

Buku.java

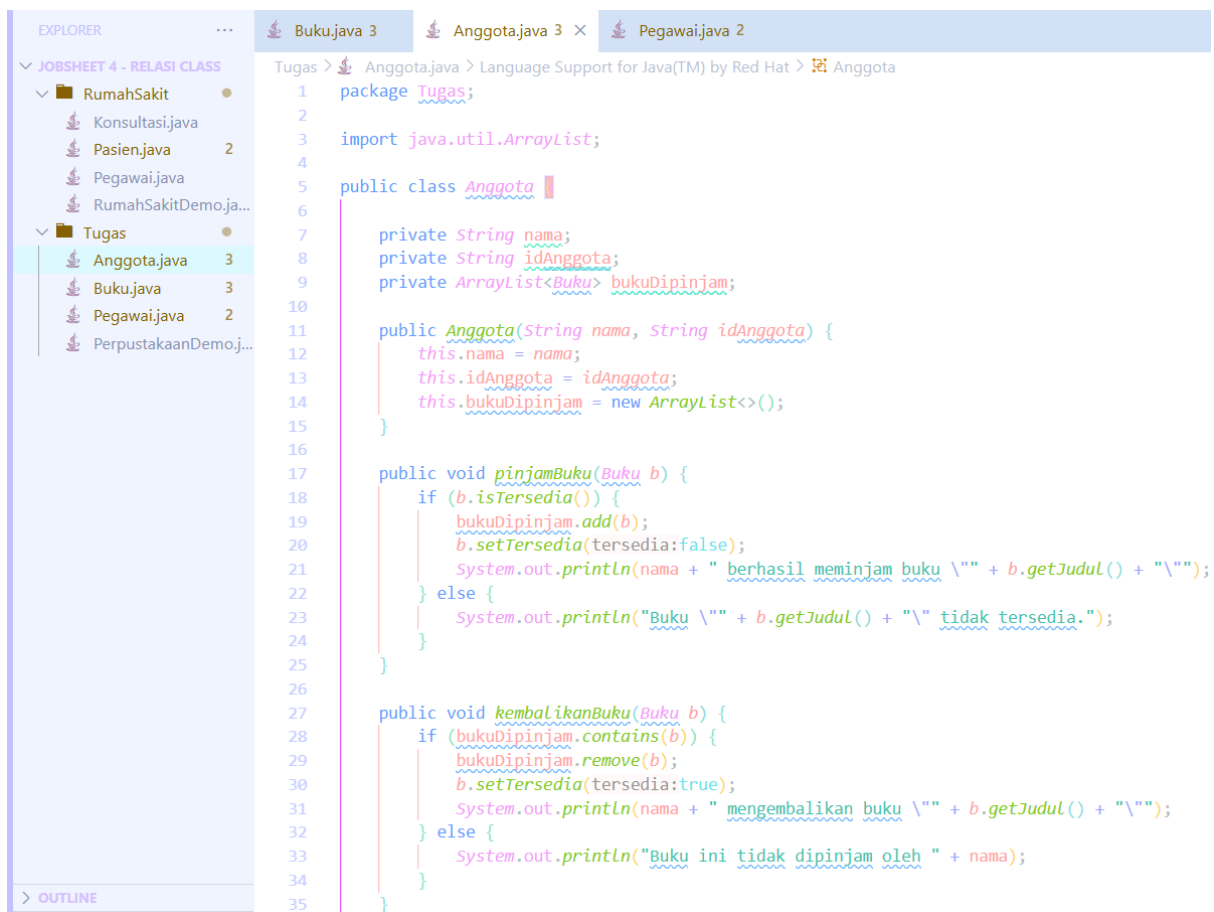
```

1 package Tugas;
2
3 public class Buku {
4
5     private String judul;
6     private String penulis;
7     private int tahunTerbit;
8     private boolean tersedia;
9
10    public Buku(String judul, String penulis, int tahunTerbit) {
11        this.judul = judul;
12        this.penulis = penulis;
13        this.tahunTerbit = tahunTerbit;
14        this.tersedia = true;
15    }
16
17    public void tampilkanInfo() {
18        System.out.println("Judul : " + judul);
19        System.out.println("Penulis : " + penulis);
20        System.out.println("Tahun : " + tahunTerbit);
21        System.out.println("Status : " + (tersedia ? "Tersedia" : "Dipinjam"));
22        System.out.println();
23    }
24
25    public String getJudul() {
26        return judul;
27    }
28
29    public String getPenulis() {
30        return penulis;
31    }
32
33    public int getTahunTerbit() {
34        return tahunTerbit;
35    }
36
37    public boolean isTersedia() {
38        return tersedia;
39    }

```



Anggota.java



EXPLORER

...

JOBSHEET 4 - RELASI CLASS

RumahSakit

Konsultasi.java

Pasien.java 2

Pegawai.java

RumahSakitDemo.ja...

Tugas

Anggota.java 3

Buku.java 3

Pegawai.java 2

PerpustakaanDemo.j...

Buku.java 3

Anggota.java 3 X

Pegawai.java 2

Tugas > Anggota.java > Language Support for Java(TM) by Red Hat > Anggota

5

public class Anggota {

37 public void tampilkanBukuDipinjam() {

38 System.out.println("Daftar buku yang dipinjam oleh " + nama + ":");

39 if (bukuDipinjam.isEmpty()) {

40 System.out.println(x:"- Tidak ada buku yang sedang dipinjam");

41 } else {

42 for (Buku b : bukuDipinjam) {

43 System.out.println("- " + b.getJudul() + " (" + b.getPenulis() + ")");

44 }

45 }

46 System.out.println();

47 }

48

49 public String getName() {

50 return nama;

51 }

52

53 public String getIdAnggota() {

54 return idAnggota;

55 }

56

Pegawai.java

EXPLORER

...

JOBSHEET 4 - RELASI CLASS

RumahSakit

Konsultasi.java

Pasien.java 2

Pegawai.java

RumahSakitDemo.ja...

Tugas

Anggota.java 3

Buku.java 3

Pegawai.java 2

PerpustakaanDemo.j...

Buku.java 3

Anggota.java 3

Pegawai.java 2 X

Tugas > Pegawai.java > ...

1

package Tugas;

2

3 public class Pegawai {

4

5 private String namaPegawai;

6 private String idPegawai;

7

8 public Pegawai(String namaPegawai, String idPegawai) {

9 this.namaPegawai = namaPegawai;

10 this.idPegawai = idPegawai;

11 }

12

13 public Buku inputBukuBaru(String judul, String penulis, int tahunTerbit) {

14 System.out.println("Pegawai " + namaPegawai + " menambahkan buku baru: " + judul);

15 return new Buku(judul, penulis, tahunTerbit);

16 }

17

18 public String getNamePegawai() {

19 return namaPegawai;

20 }

21

22 public String getIdPegawai() {

23 return idPegawai;

24 }

25 }

PerpustakaanDemo.java



```
1 package Tugas;
2
3 public class PerpustakaanDemo {
4
5     Run | Debug | Run main | Debug main
6     public static void main(String[] args) {
7
8         Pegawai p1 = new Pegawai(namaPegawai:"Sinta", idPegawai:"P001");
9
10        Buku b1 = p1.inputBukuBaru(judul:"Laskar Pelangi", penulis:"Andrea Hirata", tahunTerbit:2005);
11        Buku b2 = p1.inputBukuBaru(judul:"Bumi Manusia", penulis:"Pramoedya Ananta Toer", tahunTerbit:1980);
12
13        b1.tampilkanInfo();
14        b2.tampilkanInfo();
15
16        Anggota a1 = new Anggota(nama:"Maria", idAnggota:"A101");
17
18        a1.pinjamBuku(b1);
19        a1.tampilkanBukuDipinjam();
20
21        a1.pinjamBuku(b1);
22
23        a1.kembalikanBuku(b1);
24        a1.tampilkanBukuDipinjam();
25
26        a1.pinjamBuku(b2);
27        a1.tampilkanBukuDipinjam();
28    }
```

Output

```
Pegawai Sinta menambahkan buku baru: Laskar Pelangi
Pegawai Sinta menambahkan buku baru: Bumi Manusia
Judul   : Laskar Pelangi
Penulis : Andrea Hirata
Tahun   : 2005
Status  : Tersedia

Judul   : Bumi Manusia
Penulis : Pramoedya Ananta Toer
Tahun   : 1980
Status  : Tersedia

Maria berhasil meminjam buku "Laskar Pelangi"
Daftar buku yang dipinjam oleh Maria:
- Laskar Pelangi (Andrea Hirata)

Buku "Laskar Pelangi" tidak tersedia.
Maria mengembalikan buku "Laskar Pelangi"
Daftar buku yang dipinjam oleh Maria:
- Tidak ada buku yang sedang dipinjam

Maria berhasil meminjam buku "Bumi Manusia"
Daftar buku yang dipinjam oleh Maria:
- Bumi Manusia (Pramoedya Ananta Toer)
```

Link GitHub :

Semester-3/Pemrograman Berorientasi Objek/Jobsheet 4 - Relasi Class at main · MariaSavira/Semester-3