

OBJECT ORIENTED PROGRAMMING

JOBSHEET 4 RELATION CLASS



Nama : Fahrudin Zaim Ibrahim Wicaksono
NIM : 2241720253
No. Absen : 9

STATE POLYTECHNIC OF MALANG
2023

Question 1

1. Set untuk memasukkan nilai ke object, get untuk mengambil nilai dari object
2. Konstruktor default adalah konstruktor yang tidak memiliki parameter. Digunakan untuk membuat objek dengan nilai default atau awal untuk atribut-atributnya. Jika tidak ada konstruktor yang didefinisikan dalam suatu class, secara otomatis Java akan membuat konstruktor default tanpa parameter.

Konstruktor berparameter adalah konstruktor yang memiliki parameter untuk menginisialisasi nilai-nilai awal atribut-atribut dalam objek. Digunakan ketika ingin memberikan nilai awal yang berbeda saat membuat objek. Konstruktor ini memungkinkan pengguna untuk langsung menentukan nilai atribut saat pembuatan objek.

3. Processor proc

```
public Laptop(String merk, Processor proc) {  
    this.merk = merk;  
    this.proc = proc;  
}
```

- 4.
5. Memanggil method info dari objek proc
6. p adalah variabel dari object proc
tidak ada perubahan

```
Merk Laptop = Zenbook  
Merk Processor = intel i9  
Cache Memory = 3.00
```

Question 2

1.

```
3 private Mobil mobil;  
4 private Sopir sopir;
```
2. Karena biaya yang terkait dengan sopir dan mobil seringkali berhubungan dengan periode waktu atau durasi penggunaan.
3. Untuk menghitung biaya mobil dan sopir yang harus dibayarkan oleh pelanggan
4. Untuk menetapkan objek Mobil yang disimpan dalam variabel 'm' dan objek Sopir yang disimpan dalam variabel 's' ke dalam objek Pelanggan yang disimpan dalam variabel 'p'.
5. Untuk menghitung biaya total yang harus dibayarkan oleh pelanggan
6. Untuk menampilkan merk mobil

```
Biaya Total = 1100000  
Avanza
```

Question 3

1. Untuk menampilkan info dari kelas pegawai

```

1  public class MainPertanyaan {
    Run | Debug
2      public static void main(String args[]) {
3          Pegawai masinis = new Pegawai(nip:"1234", nama:"Spongebob S
4          KeretaApi keretaApi = new KeretaApi(nama:"Gaya Baru", kelas
5          System.out.println(keretaApi.info());
6      }
7  }

```

2. Exception in thread "main" java.lang.NullPointerException: Cannot invoke "Pegawai.info()" because "this.asisten" is null
 at KeretaApi.info(KeretaApi.java:55)
 at MainPertanyaan.main(MainPertanyaan.java:5)
 PS C:\Users\fahru\Documents\OOP Week 2>

3. Karena this asisten is null alias tidak ada isinya.

```

50      public String info() {
51          String info = "";
52          info += "Nama      : " + this.nama + "\n";
53          info += "Kelas    : " + this.kelas + "\n";
54          info += "Masinis   : " + this.masinis.info() + "\n";
55          //info += "Asisten  : " + this.asisten.info() + "\n";
56          return info;
57      }
58  }

```

4. Nama : Gaya Baru
 Kelas : Bisnis
 Masinis : Nip : 1234
 Nama : Spongebob Squarepants

Question 4

1. 10
2. Mengecek apakah atribut penumpang dari objek Kursi tidak kosong (null). Jika tidak kosong, berarti ada objek Penumpang yang ditetapkan ke kursi ini.
3. Adanya konvensi dalam indeksasi array. Jika pengguna ingin menetapkan penumpang ke kursi nomor 1, indeks array yang sesuai adalah 0. Jadi, nilai nomor dikurangi 1 untuk menyesuaikan nomor kursi dengan indeks array yang sesuai.

```

Kode: A
Nomor: 1
Penumpang: KTP : 56789
Nama : Budi

Nomor: 2
Nomor: 3
Nomor: 4
Nomor: 5
Nomor: 6
Nomor: 7
Nomor: 8
Nomor: 9
Nomor: 10

```

4.

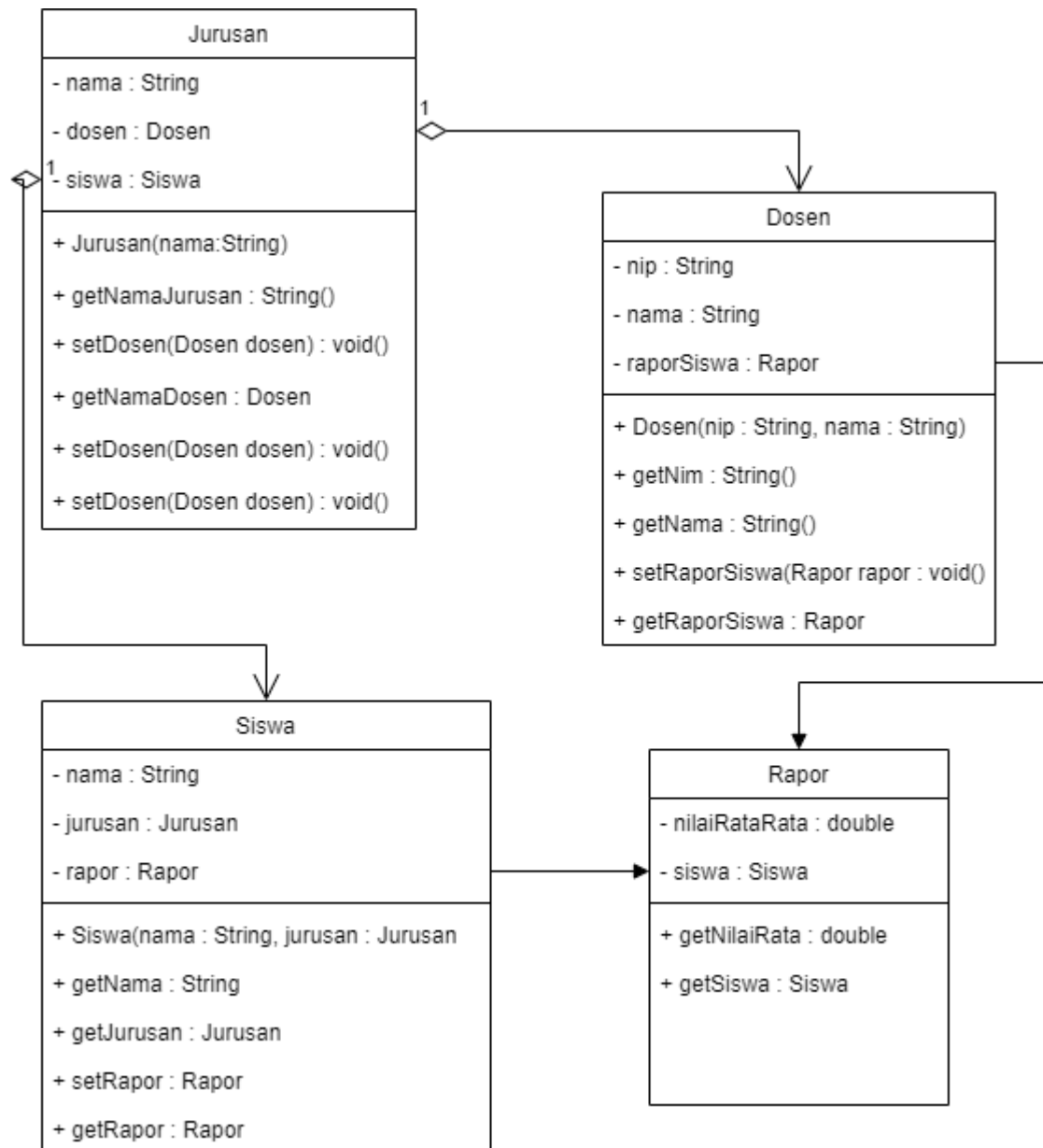
```
public void setPenumpang(Penumpang penumpang) {  
    if (this.penumpang == null) {  
        this.penumpang = penumpang;  
    } else {  
        System.out.println(x:"Kursi sudah ditempati!");  
    }  
}
```

5.

```
Kursi sudah ditempati!  
Kode: A  
Nomor: 1  
Penumpang: KTP : 12345  
Nama : Mr. Krab
```

```
Nomor: 2  
Nomor: 3  
Nomor: 4  
Nomor: 5  
Nomor: 6  
Nomor: 7  
Nomor: 8  
Nomor: 9  
Nomor: 10
```

Assignment



```
1 public class Dosen {
2     private String nip;
3     private String nama;
4     private Rapor raporSiswa;
5
6     public Dosen(String nip, String nama) {
7         this.nip = nip;
8         this.nama = nama;
9     }
10
11     public void setRaporSiswa(Rapor rapor) {
12         this.raporSiswa = rapor;
13     }
14
15     public Rapor getRaporSiswa() {
16         return raporSiswa;
17     }
18
19     public String getNip() {
20         return nip;
21     }
22
23     public String getNama() {
24         return nama;
25     }
26 }
27
```

```
1 public class Jurusan {
2     private String namaJurusan;
3     private Dosen dosen;
4
5     public Jurusan(String namaJurusan) {
6         this.namaJurusan = namaJurusan;
7     }
8
9     public void setDosen(Dosen dosen) {
10         this.dosen = dosen;
11     }
12
13     public Dosen getDosen() {
14         return dosen;
15     }
16
17     public String getNamaJurusan() {
18         return namaJurusan;
19     }
20 }
```

```
1 public class Rapor {
2     private double nilaiRataRata;
3     private Siswa siswa;
4
5     public Rapor(double nilaiRataRata, Siswa siswa) {
6         this.nilaiRataRata = nilaiRataRata;
7         this.siswa = siswa;
8     }
9
10    public double getNilaiRataRata() {
11        return nilaiRataRata;
12    }
13
14    public Siswa getSiswa() {
15        return siswa;
16    }
17 }
18
```

```
1 public class Siswa {
2     private String nama;
3     private Jurusan jurusan;
4     private Rapor rapor;
5
6     public Siswa(String nama, Jurusan jurusan) {
7         this.nama = nama;
8         this.jurusan = jurusan;
9     }
10
11    public void setRapor(Rapor rapor) {
12        this.rapor = rapor;
13    }
14
15    public String getNama() {
16        return nama;
17    }
18
19    public Jurusan getJurusan() {
20        return jurusan;
21    }
22
23    public Rapor getRapor() {
24        return rapor;
25    }
26 }
```

Informasi Siswa:

Nama: Ibra

Jurusan: Informatika

Informasi Rapor Siswa:

Nilai Rata-Rata: 85.7

Informasi Dosen:

NIP: 12345

Nama: Mr. Bagus

Informasi Rapor Siswa yang diajar oleh Dosen:

Nilai Rata-Rata: 85.7