

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN BERORIENTASI OBJEK**  
**MODUL 1**  
**Class And Object**



Oleh :  
Riska Vany Dewi  
2311103111  
S1SI-07-C

**PROGRAM STUDI S1 SISTEM INFORMASI**  
**FAKULTAS REKAYASA INDUSTRI**  
**UNIVERSITAS TELKOM PURWOKERTO**

**2024**



## I. GUIDED

### Buku.java

```
package latihanp3riskavanydewi;

/**
 *
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class Buku {
    String judul, penulis;
    int tahunTerbit;

    public Buku (String judul, String penulis, int tahunTerbit){
        this.judul = judul;
        this.penulis = penulis;
        this.tahunTerbit = tahunTerbit;
    }
}
```

#### Penjelasan:

Kelas `Buku` dalam kode Java di atas adalah representasi sederhana dari sebuah buku, dengan atribut `judul`, `penulis`, dan `tahunTerbit`. Dideklarasikan dalam package `latihanp3riskavanydewi`, kelas ini dilengkapi dengan konstruktor yang memungkinkan pengguna untuk membuat objek `Buku` dengan mengisi ketiga atribut tersebut saat objek dibuat. Penggunaan kata kunci `this` dalam konstruktor berfungsi untuk membedakan antara variabel instan dan parameter yang diterima, sehingga objek dapat menyimpan data buku dengan benar. Kelas ini dapat diperluas dengan menambahkan metode lain untuk mengakses atau memanipulasi data buku lebih lanjut.

### Mahasiswa.java

```
package latihanp3riskavanydewi;
```

```
/**
```

```
*
```

```
* @author Riska Vany Dewi
```

```
* 2311103111
```

```
* S1SI07C
```

```
*/
```

```
public class Mahasiswa {
```

```
    String nama;
```

```
    String nimMahasiswa;
```



### Penjelasan:

Kelas 'Mahasiswa' dalam kode Java di atas adalah representasi sederhana dari entitas mahasiswa, dengan dua atribut utama: 'nama' dan 'nimMahasiswa'. Dideklarasikan dalam package 'latihanp3riskavanydewi', kelas ini memiliki konstruktor yang memungkinkan pengguna untuk membuat objek 'Mahasiswa' dengan memberikan nilai untuk kedua atribut tersebut saat objek diinisialisasi. Penggunaan kata kunci 'this' dalam konstruktor berfungsi untuk membedakan antara variabel instan dan parameter yang diterima, sehingga objek dapat menyimpan informasi mahasiswa dengan benar. Kelas ini dapat dikembangkan lebih lanjut dengan menambahkan metode untuk mengakses atau memodifikasi data mahasiswa sesuai kebutuhan.

### Peminjaman.java

```
package latihanp3riskavanydewi;

/**
 *
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class Peminjaman {
    Buku buku;
    Mahasiswa mahasiswa;
    String tanggalPeminjaman;

    public Peminjaman(Buku buku, Mahasiswa mahasiswa, String
tanggalPeminjaman){
        this.buku = buku;
        this.mahasiswa = mahasiswa;
        this.tanggalPeminjaman = tanggalPeminjaman;
    }

    public void tampilkanPeminjaman(){
        System.out.println (mahasiswa.nama+ " meminjam buku dengan judul " +
            buku.judul+ " pada tanggal " +tanggalPeminjaman);
    }
}
```

### Penjelasan:

Kelas 'Peminjaman' dalam kode Java di atas merupakan representasi dari proses peminjaman buku oleh seorang mahasiswa, dengan tiga atribut: 'buku', 'mahasiswa', dan 'tanggalPeminjaman'. Dideklarasikan dalam package 'latihanp3riskavanydewi', kelas ini memiliki konstruktor yang memungkinkan pengguna untuk membuat objek 'Peminjaman' dengan mengisi informasi mengenai buku yang dipinjam, mahasiswa yang meminjam, dan tanggal peminjaman.

Metode `tampilkanPeminjaman()` digunakan untuk menampilkan informasi tentang peminjaman tersebut, menggabungkan nama mahasiswa, judul buku, dan tanggal peminjaman dalam format yang mudah dibaca. Kelas ini dapat dikembangkan lebih lanjut dengan menambahkan fitur seperti pengembalian buku atau pengecekan status peminjaman.

File Utama (Nama file sesuai dengan nama project, contoh `LatihanP3SenaWijayanto.java`)

```
package latihanp3riskavanydewi;

/**
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class LatihanP3RiskaVanyDewi {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Buku buku = new Buku ("The Hunger Games", "Suzanne Collins", 2008);
        Mahasiswa mahasiswa = new Mahasiswa ("Riska Vany Dewi", "2311103111");
        Peminjaman peminjaman = new Peminjaman (buku, mahasiswa,
"07/10/2024");

        peminjaman.tampilkanPeminjaman();

    }

}
```

#### Penjelasan:

Kelas `LatihanP3RiskaVanyDewi` dalam kode Java di atas adalah kelas utama yang berfungsi sebagai titik masuk untuk menjalankan program. Dalam metode `main`, objek `Buku`, `Mahasiswa`, dan `Peminjaman` dibuat dengan memberikan informasi yang relevan untuk masing-masing kelas. Objek `buku` diinisialisasi dengan judul "The Hunger Games", penulis "Suzanne Collins", dan tahun terbit 2008. Kemudian, objek `mahasiswa` dibuat dengan nama "Riska Vany Dewi" dan NIM "2311103111". Selanjutnya, objek `peminjaman` dibuat untuk mengaitkan buku dan mahasiswa dengan tanggal peminjaman "07/10/2024". Terakhir, metode `tampilkanPeminjaman()` dipanggil untuk menampilkan informasi peminjaman di konsol, yang mencetak nama mahasiswa, judul buku, dan tanggal peminjaman dalam format yang jelas dan informatif. Kelas ini menyatukan semua komponen yang telah didefinisikan sebelumnya dan menunjukkan cara mereka berinteraksi.

## II. UNGUIDED

1. Kembangkan Sistem Peminjaman Buku dengan menambahkan fitur untuk mengembalikan bukudan mencatat tanggal pengembalian.

Pengembalian.java

```
package latihanp3riskavanydewi;

/**
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class Pengembalian {
    Buku buku;
    Mahasiswa mahasiswa;
    String tanggalPengembalian;

    public Pengembalian(Buku buku, Mahasiswa mahasiswa, String tanggalPengembalian){
        this.buku = buku;
        this.mahasiswa = mahasiswa;
        this.tanggalPengembalian = tanggalPengembalian;
    }

    public void tampilkanPengembalian(){
        System.out.println (mahasiswa.nama+ " mengembalikan buku dengan judul " +
            buku.judul+ " pada tanggal " +tanggalPengembalian);
    }
}
```

Penjelasan:

Kelas Pengembalian bertanggung jawab untuk mencatat proses pengembalian buku yang telah dipinjam, dengan atribut untuk menyimpan informasi mengenai buku, mahasiswa, dan tanggal pengembalian. Konstruktor kelas ini menginisialisasi atribut dengan parameter yang diberikan, dan metode tampilkanPengembalian() menampilkan informasi terkait pengembalian, termasuk siapa yang mengembalikan buku dan kapan.

File Utama

```
package latihanp3riskavanydewi;

/**
 *
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class LatihanP3RiskaVanyDewi {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Buku buku = new Buku ("The Hunger Games", "Suzanne Collins", 2008);
        Mahasiswa mahasiswa = new Mahasiswa ("Riska Vany Dewi", "2311103111");
        Peminjaman peminjaman = new Peminjaman (buku, mahasiswa, "07/10/2024");

        peminjaman.tampilkanPeminjaman();

        Pengembalian pengembalian = new Pengembalian (buku, mahasiswa, "13/10/2024");

        pengembalian.tampilkanPengembalian();
    }
}
```

```
}
```

Penjelasan:

Kelas LatihanP3RiskaVanyDewi berfungsi sebagai titik masuk program, di mana objek Buku, Mahasiswa, Peminjaman, dan Pengembalian dibuat dan dikelola. Dalam metode main, informasi peminjaman dan pengembalian ditampilkan dengan memanggil metode yang relevan, sehingga memberikan gambaran lengkap mengenai kegiatan peminjaman buku dalam sistem.

Output :

```
Output - LatihanP3RiskaVanyDewi (run)
run:
Riska Vany Dewi meminjam buku dengan judul The Hunger Games pada tanggal 07/10/2024
Riska Vany Dewi mengembalikan buku dengan judul The Hunger Games pada tanggal 13/10/2024
BUILD SUCCESSFUL (total time: 0 seconds)
```

## 2. Rata-rata Nilai

Mahasiswa.java

```
package unguided2;

/**
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class Mahasiswa {
    String nama;
    String nimMahasiswa;

    public Mahasiswa (String nama, String nimMahasiswa){
        this.nama = nama;
        this.nimMahasiswa = nimMahasiswa;
    }
}
```



```
}
```

Penjelasan:

Kelas `Mahasiswa` dalam kode di atas merepresentasikan entitas mahasiswa dengan dua atribut utama: `nama` dan `nimMahasiswa`, yang masing-masing menyimpan nama dan nomor induk mahasiswa. Dideklarasikan dalam package `unguided2`, kelas ini memiliki konstruktor yang memungkinkan pengguna untuk menginisialisasi objek `Mahasiswa` dengan memberikan nilai untuk kedua atribut tersebut saat objek dibuat. Penggunaan kata kunci `this` dalam konstruktor membantu membedakan antara atribut kelas dan parameter yang diterima, sehingga memastikan data mahasiswa dapat dikelola dengan jelas dan terstruktur dalam aplikasi yang lebih besar.

Nilai.java

```
package unguided2;

/**
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class Nilai {
    int nilai1;
    int nilai2;
    int nilai3;

    public Nilai (int nilai1, int nilai2, int nilai3){
        this.nilai1 = nilai1;
        this.nilai2 = nilai2;
        this.nilai3 = nilai3;
    }
}
```

Penjelasan:

Kelas `Nilai` dalam kode di atas merepresentasikan kumpulan nilai yang dimiliki oleh seorang



mahasiswa, dengan tiga atribut integer: `nilai1`, `nilai2`, dan `nilai3`, yang masing-masing menyimpan nilai tertentu. Kelas ini dideklarasikan dalam package `unguided2` dan memiliki konstruktor yang memungkinkan pengguna untuk menginisialisasi objek `Nilai` dengan memberikan tiga nilai saat objek dibuat. Penggunaan kata kunci `this` dalam konstruktor berfungsi untuk membedakan antara atribut kelas dan parameter konstruktor, memastikan bahwa nilai yang benar diassign ke atribut yang sesuai. Dengan demikian, kelas ini menyediakan struktur yang jelas untuk menyimpan dan mengelola nilai-nilai akademis, yang dapat digunakan dalam konteks yang lebih luas, seperti perhitungan rata-rata atau evaluasi kinerja mahasiswa.

RataRata.java

```
package unguided2;

/**
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class RataRata {
    Nilai nilai;

    public RataRata(Nilai nilai) {
        this.nilai = nilai;
    }

    public double hitungRataRata(){
        return(nilai.nilai1 + nilai.nilai2 + nilai.nilai3) / 3.0;
    }
}
```

Penjelasan :

Kelas `RataRata` dalam kode di atas berfungsi untuk menghitung rata-rata dari tiga nilai yang disimpan dalam objek `Nilai`. Kelas ini dideklarasikan dalam package `unguided2` dan memiliki satu atribut, yaitu `nilai`, yang merupakan referensi ke objek `Nilai`.

Konstruktor `RataRata` menerima parameter `Nilai` untuk menginisialisasi atribut `nilai`, memungkinkan akses ke nilai-nilai yang ingin dihitung rata-ratanya. Metode `hitungRataRata()` melakukan perhitungan dengan menjumlahkan ketiga nilai (`nilai1`, `nilai2`, `nilai3`) dan membaginya dengan 3.0 untuk menghasilkan hasil dalam tipe `double`, sehingga dapat memberikan nilai rata-rata yang lebih presisi. Dengan demikian, kelas ini menyediakan fungsi yang diperlukan untuk menganalisis kinerja akademis mahasiswa berdasarkan nilai yang diberikan.

File Utama

```
package unguided2;

/**
 * @author Riska Vany Dewi
 * 2311103111
 * S1SI07C
 */
public class Unguided2 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Mahasiswa mahasiswa = new Mahasiswa ("Riska Vany Dewi", "2311103111");
        Nilai nilai = new Nilai (95, 87, 91);
        RataRata rataRata = new RataRata(nilai);
        double hasilRataRata = rataRata.hitungRataRata();

        System.out.println("Nama Mahasiswa : " + mahasiswa.nama);
        System.out.println("NIM : " + mahasiswa.nimMahasiswa);
        System.out.println("Rata-rata Nilai : " + hasilRataRata);
    }
}
```

```
}
```

#### Penjelasan:

Kelas `Unguided2` dalam kode di atas berfungsi sebagai titik utama untuk menjalankan program, mengintegrasikan semua komponen yang telah didefinisikan sebelumnya, yaitu `Mahasiswa`, `Nilai`, dan `RataRata`. Dalam metode `main`, objek `Mahasiswa` dibuat dengan nama "Riska Vany Dewi" dan NIM "2311103111". Selanjutnya, objek `Nilai` diinisialisasi dengan tiga nilai (95, 87, dan 91) yang mencerminkan nilai akademis mahasiswa. Kelas `RataRata` kemudian digunakan untuk menghitung rata-rata dari nilai-nilai tersebut, dengan memanggil metode `hitungRataRata()` yang mengembalikan hasil perhitungan. Akhirnya, program menampilkan nama mahasiswa, NIM, dan rata-rata nilai ke konsol, memberikan gambaran yang jelas tentang informasi mahasiswa dan kinerja akademisnya. Dengan demikian, kelas ini berfungsi untuk menghubungkan semua bagian dari sistem dan menampilkan hasil secara terstruktur.

#### Output:

```
Output - Unguided2 (run)

run:
Nama Mahasiswa : Riska Vany Dewi
NIM : 2311103111
Rata-rata Nilai : 91.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

