

JOBSHEET 2
OBJECT ORIENTED PROGRAMMING



Arranged By :

Lenka Melinda Florianka

2241720074

Class 2I

INFORMATION TECHNOLOGY
D-IV INFORMATICS ENGINEERING
MALANG STATE POLYTECHNIC
2023

Practicum 1

1.

Karyawan
+ id: string + nama: string + jenis_kelamin: string + jabatan: string + gaji: double
+ getNama(): string + getJenisKelamin(): string + getJabatan(): string + getGaji(): double + setGaji(gaji double)

Perusahaan
+ daftar_karyawan: string
+ tambahKaryawan(karyawan Karyawan) + hapusKaryawan(karyawan Karyawan) + getJabatan(): string + cariKaryawan(id string) + hitungTotalGaji(): double

2. Class yang bisa dibuat:

- Karyawan
- Perusahaan

3. Atribut dan Tipe Data:

a. Karyawan:

- id (string)
- nama (string)
- jenis_kelamin (string)
- jabatan (string)
- gaji (double)

b. Perusahaan:

- daftar_karyawan (List of Karyawan)

4. Method-method:

a. Karyawan:

- getNama(): string: Mengembalikan nama karyawan.
- getJenisKelamin(): string: Mengembalikan jenis kelamin karyawan.
- getJabatan(): string: Mengembalikan jabatan karyawan.
- getGaji(): double: Mengembalikan gaji karyawan.
- setGaji(gaji: double): Mengatur gaji karyawan.

b. Perusahaan:

- tambahKaryawan(karyawan: Karyawan): Menambahkan karyawan ke daftar_karyawan.
- hapusKaryawan(karyawan: Karyawan): Menghapus karyawan dari daftar_karyawan.
- cariKaryawan(id: string): Karyawan: Mencari karyawan berdasarkan ID.
- hitungTotalGaji(): double: Menghitung total gaji seluruh karyawan.

Practicum 2

Code :

```
1 public class TestMahasiswa{
2     public static void main(String[] args) {
3         Mahasiswa mhs1 = new Mahasiswa();
4         mhs1.nim = 101;
5         mhs1.nama = "Lestari";
6         mhs1.alamat = "Jl. Vinola NO 1A";
7         mhs1.kelas = "1A";
8         mhs1.tampilBiodata();
9     }
10 }
11 }
```

```
1 public class Mahasiswa {
2     public int nim;
3     public String nama;
4     public String alamat;
5     public String kelas;
6
7     public void tampilBiodata(){
8         System.out.println("Nim      : " + nim);
9         System.out.println("Nama      : " + nama);
10        System.out.println("Alamat   : " + alamat);
11        System.out.println("Kelas   : " + kelas);
12    }
13 }
14 }
```

Output :

```
Nim      : 101
Nama      : Lestari
Alamat    : Jl. Vinola NO 1A
Kelas    : 1A
PS C:\Dev\OOP>
```

Questions :

- Jelaskan pada bagian mana proses pendeklarasian atribut pada program diatas!

Pendeklarasian atribut terletak pada awal penulisan program, lebih tepatnya sesudah class pertama kali dibuat, diawali dengan public, lalu tipe data atributnya, lalu nama atribut itu sendiri

- Jelaskan pada bagian mana proses pendeklarasian method pada program diatas!

Pendeklarasian method terletak sesudah pendeklarasian atribut, diawali dengan kata public void, lalu nama methodnya

- Berapa banyak objek yang di instansiasi pada program diatas!

Ada satu objek yang di instansiasi pada program diatas, yaitu mhs1

- Apakah yang sebenarnya dilakukan pada sintaks program "mhs1.nim=101" ?

Memberikan value 101 pada nim kepemilikan objek mhs1 berupa angka

- Apakah yang sebenarnya dilakukan pada sintaks program "mhs1.tampilBiodata()" ?

Memanggil method pada class Mahasiswa untuk menampilkan biodata dari mhs1

- Instansiasi 2 objek lagi pada program diatas!

```
1  Mahasiswa mhs2 = new Mahasiswa();
2      mhs2.nim = 102;
3      mhs2.nama = "Dina";
4      mhs2.alamat = "Jl. Teratai NO 5A";
5      mhs2.kelas = "1A";
6      mhs2.tampilBiodata();
7
8      Mahasiswa mhs3 = new Mahasiswa();
9      mhs3.nim = 103;
10     mhs3.nama = "Budi";
11     mhs3.alamat = "Jl. Semanji NO 9A";
12     mhs3.kelas = "1A";
13     mhs3.tampilBiodata();
```

```
Nim      : 102
Nama     : Dina
Alamat  : Jl. Teratai NO 5A
Kelas  : 1A
```

```
Nim      : 103
Nama     : Budi
Alamat  : Jl. Semanji NO 9A
Kelas  : 1A
PS C:\Dev\OOP>
```

Practicum 3

Code :

```
1 public class TestBarang {
2     public static void main(String[] args) {
3         Barang brg1 = new Barang();
4         brg1.namaBrg = "Pensil";
5         brg1.jenisBrg = "ATK";
6         brg1.stok = 10;
7         brg1.tampilBarang();
8
9         // menampilkan dan mengisi argumen untuk menambahkan stok barang
10        System.out.println("Stok Baru adalah " +brg1.tambahStok(20));
11    }
12 }
```

```
1 public class Barang {
2     public String namaBrg;
3     public String jenisBrg;
4     public int stok;
5
6     public void tampilBarang(){
7         System.out.println("Nama Barang      : " +namaBrg);
8         System.out.println("Jenis Barang   : " +jenisBrg);
9         System.out.println("Stok         : " +stok);
10    }
11
12    //method dengan argumen dan nilai balik (return)
13    public int tambahStok(int brgMasuk){
14        int stokBaru = brgMasuk+stok;
15        return stokBaru;
16    }
17 }
```

Output :

```
Nama Barang      : Pensil
Jenis Barang     : ATK
Stok             : 10
Stok Baru adalah 30
PS C:\Dev\OOP>
```

Questions :

- Apakah fungsi argumen dalam suatu method?

Kegunaan Argumen didalam sebuah fungsi ialah memberikan nilai yang harus dieksekusi dalam fungsi tersebut

- Ambil kesimpulan tentang kegunaan dari kata kunci return , dan kapan suatu method harus memiliki return!

Penggunaan kata kunci "return" dalam suatu metode sangat tergantung pada tujuan dan kebutuhan dari metode tersebut. Ini memungkinkan kita untuk mengatur aliran program dan mengembalikan hasil perhitungan atau data yang relevan ke pemanggil metode.

Tugas

1. Suatu toko persewaan video game salah satu yang diolah adalah peminjaman, dimana data yang dicatat ketika ada orang yang melakukan peminjaman adalah id, nama member, nama game, dan harga yang harus dibayar. Setiap peminjaman bisa menampilkan data hasil peminjaman dan harga yang harus dibayar. Buatlah class diagram pada studi kasus diatas! Penjelasan:
 - Harga yang harus dibayar diperoleh dari lama sewa x harga.
 - Diasumsikan 1x transaksi peminjaman game yang dipinjam hanya 1 game saja.

2. Buatlah program dari class diagram yang sudah anda buat di no 1!

Code :

```
1 class Peminjaman {
2     private int id;
3     private String namaMember;
4     private String namaGame;
5     private int hargaDibayar;
6
7     public Peminjaman(int id, String namaMember, String namaGame, int hargaDibayar) {
8         this.id = id;
9         this.namaMember = namaMember;
10        this.namaGame = namaGame;
11        this.hargaDibayar = hargaDibayar;
12    }
13
14    // Getter dan setter
15    public int getId() {
16        return id;
17    }
18
19    public String getNamaMember() {
20        return namaMember;
21    }
22
23    public String getNamaGame() {
24        return namaGame;
25    }
26
27    public int getHargaDibayar() {
28        return hargaDibayar;
29    }
30
31    public void setId(int id) {
32        this.id = id;
33    }
34
35    public void setNamaMember(String namaMember) {
36        this.namaMember = namaMember;
37    }
38
39    public void setNamaGame(String namaGame) {
40        this.namaGame = namaGame;
41    }
42
43    public void setHargaDibayar(int hargaDibayar) {
44        this.hargaDibayar = hargaDibayar;
45    }
46
47    public int hitungHarga(int lamaSewa) {
48        return lamaSewa * hargaDibayar;
49    }
50 }
```

```
1 public class Game {
2     private int id;
3     private String nama;
4     private int harga;
5
6     public Game(int id, String nama, int harga) {
7         this.id = id;
8         this.nama = nama;
9         this.harga = harga;
10    }
11
12    // Getter dan setter
13    public int getId() {
14        return id;
15    }
16
17    public String getNama() {
18        return nama;
19    }
20
21    public int getHarga() {
22        return harga;
23    }
24
25    public void setId(int id) {
26        this.id = id;
27    }
28
29    public void setNama(String nama) {
30        this.nama = nama;
31    }
32
33    public void setHarga(int harga) {
34        this.harga = harga;
35    }
36 }
```

```
1 class Member {
2     private int id;
3     private String nama;
4     private int harga;
5
6     public Member(int id, String nama, int harga) {
7         this.id = id;
8         this.nama = nama;
9         this.harga = harga;
10    }
11
12    // Getter dan setter
13    public int getId() {
14        return id;
15    }
16
17    public String getNama() {
18        return nama;
19    }
20
21    public int getHarga() {
22        return harga;
23    }
24
25    public void setId(int id) {
26        this.id = id;
27    }
28
29    public void setNama(String nama) {
30        this.nama = nama;
31    }
32
33    public void setHarga(int harga) {
34        this.harga = harga;
35    }
36 }
```

```
1 public class Main {
2     public static void main(String[] args) {
3         // Membuat objek Game
4         Game game1 = new Game(1, "Game A", 50);
5         Game game2 = new Game(2, "Game B", 60);
6
7         // Membuat objek Member
8         Member member1 = new Member(1, "Anggota 1", 10);
9         Member member2 = new Member(2, "Anggota 2", 20);
10
11        // Membuat objek Peminjaman
12        Peminjaman peminjaman1 = new Peminjaman(1, member1.getNama(), game1.getNama(), game1.getHarga());
13        Peminjaman peminjaman2 = new Peminjaman(2, member2.getNama(), game2.getNama(), game2.getHarga());
14
15        // Menghitung harga peminjaman berdasarkan lama sewa
16        int lamaSewa1 = 3; // Misalnya, peminjaman 3 hari
17        int lamaSewa2 = 2; // Misalnya, peminjaman 2 hari
18
19        int hargaPeminjaman1 = peminjaman1.hitungHarga(lamaSewa1);
20        int hargaPeminjaman2 = peminjaman2.hitungHarga(lamaSewa2);
21
22        // Menampilkan informasi peminjaman
23        System.out.println("Informasi Peminjaman 1:");
24        System.out.println("Nama Member: " + peminjaman1.getNamaMember());
25        System.out.println("Nama Game: " + peminjaman1.getNamaGame());
26        System.out.println("Harga Peminjaman: $" + hargaPeminjaman1);
27
28        System.out.println();
29
30        System.out.println("Informasi Peminjaman 2:");
31        System.out.println("Nama Member: " + peminjaman2.getNamaMember());
32        System.out.println("Nama Game: " + peminjaman2.getNamaGame());
33        System.out.println("Harga Peminjaman: $" + hargaPeminjaman2);
34    }
35 }
```

Output :

```
Informasi Peminjaman 1:
Nama Member: Anggota 1
Nama Game: Game A
Harga Peminjaman: $150

Informasi Peminjaman 2:
Nama Member: Anggota 2
Nama Game: Game B
Harga Peminjaman: $120
PS C:\Dev\OOP>
```

3. Buatlah program sesuai dengan class diagram berikut ini:

Code :

```
1 public class Lingkaran {
2     // Variabel-variabel instance
3     private double phi;
4     private double r;
5
6     // Konstruktor
7     public Lingkaran(double phi, double r) {
8         this.phi = phi;
9         this.r = r;
10    }
11
12    // Method untuk menghitung luas lingkaran
13    public double hitungLuas() {
14        return phi * r * r;
15    }
16
17    // Method untuk menghitung keliling lingkaran
18    public double hitungKeliling() {
19        return 2 * phi * r;
20    }
21
22    // Getter dan setter untuk phi
23    public double getPhi() {
24        return phi;
25    }
26
27    public void setPhi(double phi) {
28        this.phi = phi;
29    }
30
31    // Getter dan setter untuk r
32    public double getR() {
33        return r;
34    }
35
36    public void setR(double r) {
37        this.r = r;
38    }
39
40    public static void main(String[] args) {
41        // Contoh penggunaan class Lingkaran
42        Lingkaran lingkaran = new Lingkaran(3.14, 5.0);
43        System.out.println("Luas Lingkaran: " + lingkaran.hitungLuas());
44        System.out.println("Keliling Lingkaran: " + lingkaran.hitungKeliling());
45
46        // Mengubah nilai phi dan r
47        lingkaran.setPhi(3.14159265359);
48        lingkaran.setR(6.0);
49        System.out.println("Luas Lingkaran (setelah perubahan): " + lingkaran.hitungLuas());
50        System.out.println("Keliling Lingkaran (setelah perubahan): " + lingkaran.hitungKeliling());
51    }
52 }
```

Output :

```
Luas Lingkaran: 78.5
Keliling Lingkaran: 31.400000000000002
Luas Lingkaran (setelah perubahan): 113.09733552924001
Keliling Lingkaran (setelah perubahan): 37.699111843080004
PS C:\Dev\OOP>
```

4. Buatlah program sesuai dengan class diagram berikut ini:

Code :

```
1 public class Barangg {
2     // Variabel-variabel instance
3     private String kode;
4     private String namaBarang;
5     private int hargaDasar;
6     private float diskon;
7
8     // Konstruktor
9     public Barangg(String kode, String namaBarang, int hargaDasar, float diskon) {
10         this.kode = kode;
11         this.namaBarang = namaBarang;
12         this.hargaDasar = hargaDasar;
13         this.diskon = diskon;
14     }
15
16     // Method untuk menghitung harga jual
17     public int hitungHargaJual() {
18         float hargaDiskon = hargaDasar - (diskon / 100 * hargaDasar);
19         return Math.round(hargaDiskon);
20     }
21
22     // Method untuk menampilkan data barang
23     public void tampilData() {
24         System.out.println("Kode: " + kode);
25         System.out.println("Nama Barang: " + namaBarang);
26         System.out.println("Harga Dasar: Rp " + hargaDasar);
27         System.out.println("Diskon: " + diskon + "%");
28         System.out.println("Harga Jual: Rp " + hitungHargaJual());
29     }
30
31     public static void main(String[] args) {
32         // Contoh penggunaan class Barangg
33         Barangg barang1 = new Barangg("B001", "Laptop", 5000000, 10.0f);
34         Barangg barang2 = new Barangg("B002", "Smartphone", 3000000, 5.5f);
35
36         // Menampilkan data barang
37         System.out.println("Data Barang 1:");
38         barang1.tampilData();
39
40         System.out.println("\nData Barang 2:");
41         barang2.tampilData();
42     }
43 }
44
```

Output :


```
Data Barang 1:
Kode: B001
Nama Barang: Laptop
Harga Dasar: Rp 5000000
Diskon: 10.0%
Harga Jual: Rp 4500000
```

```
PS C:\Dev\00P>
```