

LAPORAN HASIL PRAKTIKUM 05
PEMROMGRAMAN BERBASIS OBJEK



ATHAULLA HAFIZH

244107020030

TI 2A

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2025

3. Percobaan

Percobaan 1

Kode program class Processor

```
package athaullahafizh.relasticlass.percobaan1;

public class Processor {

    private String merk;
    private double cache;

    public Processor() {
    }

    public Processor(String merk, double cache) {
        this.merk = merk;
        this.cache = cache;
    }

    public void setMerk(String merk) {
        this.merk = merk;
    }

    public String getMerk() {
        return merk;
    }

    public void setCache(double cache) {
        this.cache = cache;
    }

    public double getCache() {
        return cache;
    }

    public void info() {
        System.out.printf("Merk Processor = %s\n", merk);
    }
}
```

```
        System.out.printf("Cache Memory = %.2f\n", cache);  
    }  
}
```

Kode program class Laptop

```
package athaullahafizh.relasticlass.percobaan1;  
  
class Laptop {  
  
    private String merk;  
    private Processor proc;  
  
    public Laptop() {  
    }  
  
    public Laptop(String merk, Processor proc) {  
        this.merk = merk;  
        this.proc = proc;  
    }  
  
    public void setMerk(String merk) {  
        this.merk = merk;  
    }  
  
    public String getMerk() {  
        return merk;  
    }  
  
    public void setProc(Processor proc) {  
        this.proc = proc;  
    }  
  
    public Processor getProc() {  
        return proc;  
    }  
}
```

```
        public void info() {  
            System.out.println("Merk Laptop = " + merk);  
            proc.info();  
        }  
    }  
}
```

Kode program class MainPercobaan1

```
package athaullahafizh.relasticlass.percobaan1;  
  
class MainPercobaan1 {  
    public static void main(String[] args) {  
  
        Processor p = new Processor("Intel i5", 3);  
  
        Laptop L = new Laptop("Thinkpad", p);  
  
        L.info();  
  
        Processor p1 = new Processor();  
        p1.setMerk("Intel i5");  
        p1.setCache(4);  
        Laptop L1 = new Laptop();  
        L1.setMerk("Thinkpad");  
        L1.setProc(p1);  
        L1.info();  
    }  
}
```

Output

```
Merk Laptop = Thinkpad  
Merk Processor = Intel i5  
Cache Memory = 3,00  
Merk Laptop = Thinkpad  
Merk Processor = Intel i5  
Cache Memory = 4,00
```

Pertanyaan

1. Method setter dan getter digunakan untuk mengakses dan mengubah nilai atribut private dari luar class. Setter untuk mengubah nilai atribut, getter untuk mengambil nilai atribut.
2. Constructor default tidak memiliki parameter dan biasanya untuk inisialisasi objek kosong. Constructor berparameter memiliki parameter untuk langsung mengisi nilai atribut saat objek dibuat.
3. Atribut yang bertipe object adalah "proc" dengan tipe Processor.
4. Baris yang menunjukkan relasi adalah "private Processor proc;" - ini menunjukkan class Laptop memiliki (has-a) class Processor.
5. Sintaks proc.info() digunakan untuk memanggil method info() dari objek Processor yang dimiliki oleh class Laptop.
6. p adalah objek Processor yang sudah dibuat sebelumnya.
Jika diubah menjadi new Processor("Intel i5", 3), hasilnya sama karena membuat objek Processor baru dengan parameter yang sama.

Percobaan 2

Kode program class Mobil

```
package athaullahafizh.relasiclass.percobaan2;

public class Mobil {

    private String merk;
    private int biaya;

    public Mobil() {
    }
}
```

```
public void setMerk(String merk) {  
    this.merk = merk;  
}  
  
public String getMerk() {  
    return merk;  
}  
  
public void setBiaya(int biaya) {  
    this.biaya = biaya;  
}  
  
public int getBiaya() {  
    return biaya;  
}  
  
public int hitungBiayaMobil(int hari) {  
    return biaya * hari;  
}  
}
```

Kode program class Sopir

```
package athallahafizh.relasticlass.percobaan2;  
  
public class Sopir {  
    private String nama;  
    private int biaya;  
  
    public Sopir() {  
    }  
  
    public void setName(String nama) {  
        this.nama = nama;  
    }  
}
```

```
public String getNama() {  
    return nama;  
}  
  
public void setBiaya(int biaya) {  
    this.biaya = biaya;  
}  
  
public int getBiaya() {  
    return biaya;  
}  
  
public int hitungBiayaSopir(int hari) {  
    return biaya * hari;  
}  
}
```

Kode program class Pelanggan

```
package athallahafizh.relasticlass.percobaan2;  
  
public class Pelanggan {  
  
    private String nama;  
    private Mobil mobil;  
    private Sopir sopir;  
    private int hari;  
  
    public Pelanggan() {  
    }  
  
    public void setNama(String nama) {  
        this.nama = nama;  
    }  
}
```

```
public String getNama() {  
    return nama;  
}  
  
public void setMobil(Mobil mobil) {  
    this.mobil = mobil;  
}  
  
public Mobil getMobil() {  
    return mobil;  
}  
  
public void setSopir(Sopir sopir) {  
    this.sopir = sopir;  
}  
  
public Sopir getSopir() {  
    return sopir;  
}  
  
public void setHari(int hari) {  
    this.hari = hari;  
}  
  
public int getHari() {  
    return hari;  
}  
  
public int hitungBiayaTotal() {  
    return mobil.hitungBiayaMobil(hari) +  
sopir.hitungBiayaSopir(hari);  
}  
}
```


Kode program class MainPercobaan2

```
package athaullahafizh.relasticlass.percobaan2;

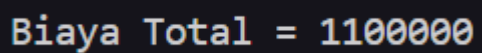
public class MainPercobaan2 {
    public static void main(String[] args) {
        Mobil m = new Mobil();
        m.setMerk("Avanza");
        m.setBiaya(350000);

        Sopir s = new Sopir();
        s.setNama("John Doe");
        s.setBiaya(200000);

        Pelanggan p = new Pelanggan();
        p.setNama("Jane Doe");
        p.setMobil(m);
        p.setSopir(s);
        p.setHari(2);

        System.out.println("Biaya Total = " + p.hitungBiayaTotal());
    }
}
```

Output



Pertanyaan

1. Baris yang menunjukkan relasi:
 - private Mobil mobil; (relasi dengan class Mobil)
 - private Sopir sopir; (relasi dengan class Sopir)
2. Method tersebut memiliki argument hari karena biaya dihitung berdasarkan jumlah hari sewa.

3. mobil.hitungBiayaMobil(hari) untuk menghitung biaya sewa mobil selama hari tertentu.
sopir.hitungBiayaSopir(hari) untuk menghitung biaya sopir selama hari tertentu.
4. p.setMobil(m) untuk mengaitkan objek mobil ke pelanggan.
p.setSopir(s) untuk mengaitkan objek sopir ke pelanggan.
5. p.hitungBiayaTotal() untuk menghitung total biaya sewa (mobil + sopir).
6. p.getMobil().getMerk() untuk mendapatkan merk mobil yang disewa oleh pelanggan.
Menggunakan chaining method untuk mengakses atribut dari objek yang berelasi.

Percobaan 3

Kode program class Pegawai

```
package athallahafizh.relasticlass.percobaan3;

public class Pegawai {

    private String nip;
    private String nama;

    public Pegawai(String nip, String nama) {
        this.nip = nip;
        this.nama = nama;
    }

    public void setNip(String nip) {
        this.nip = nip;
    }

    public String getNip() {
        return nip;
    }
}
```

```

    }

    public void setName(String nama) {
        this.nama = nama;
    }

    public String getName() {
        return nama;
    }

    public String info() {
        String info = "";
        info += "Nip: " + this.nip + "\n";
        info += "Nama: " + this.nama + "\n";
        return info;
    }
}

```

Kode program class KeretaApi

```

package athallahafizh.relasticlass.percobaan3;

class KeretaApi {

    private String nama;
    private String kelas;
    private Pegawai masinis;
    private Pegawai asisten;

    public KeretaApi(String nama, String kelas, Pegawai masinis) {
        this.nama = nama;
        this.kelas = kelas;
        this.masinis = masinis;
    }
}

```

```
    public KeretaApi(String nama, String kelas, Pegawai masinis,
Pegawai asisten) {

        this.nama = nama;
        this.kelas = kelas;
        this.masinis = masinis;
        this.asisten = asisten;
    }

    public void setName(String nama) {
        this.nama = nama;
    }

    public String getName() {
        return nama;
    }

    public void setKelas(String kelas) {
        this.kelas = kelas;
    }

    public String getKelas() {
        return kelas;
    }

    public void setMasinis(Pegawai masinis) {
        this.masinis = masinis;
    }

    public Pegawai getMasinis() {
        return masinis;
    }

    public void setAsisten(Pegawai asisten) {
        this.asisten = asisten;
    }
}
```

```

    public Pegawai getAsisten() {
        return asisten;
    }

    public String info() {
        String info = "";
        info += "Nama: " + this.nama + "\n";
        info += "Kelas: " + this.kelas + "\n";
        info += "Masinis: " + this.masinis.info() + "\n";
        if (this.asisten != null) {
            info += "Asisten: " + this.asisten.info() + "\n";
        }
        return info;
    }
}

```

Kode program class MainPercobaan3

```

package athaullahafizh.relasticlass.percobaan3;

public class MainPercobaan3 {
    public static void main(String[] args) {
        Pegawai masinis = new Pegawai("1234", "Spongebob Squarepants");
        Pegawai asisten = new Pegawai("4567", "Patrick Star");
        KeretaApi keretaApi = new KeretaApi("Gaya Baru", "Bisnis", masinis, asisten);
        System.out.println(keretaApi.info());
    }
}

```

Output

```
Nama: Gaya Baru
Kelas: Bisnis
Masinis: Nip: 1234
Nama: Spongebob Squarepants

Asisten: Nip: 4567
Nama: Patrick Star
```

Pertanyaan

1. `this.masinis.info()` dan `this.asisten.info()` digunakan untuk memanggil method `info()` dari objek Pegawai (masinis dan asisten) untuk menampilkan detail pegawai tersebut.

2. Program MainPertanyaan

```
package athallahafizh.relasticlass.percobaan3;

class MainPertanyaan {
    public static void main(String[] args) {
        Pegawai masinis = new Pegawai("1234", "Spongebob
Squarepants");
        KeretaApi keretaApi = new KeretaApi("Gaya Baru", "Bisnis",
masinis);
        System.out.println(keretaApi.info());
    }
}
```

3. Output akan error/NullPointerException karena asisten bernilai null, tetapi method `info()` tetap mencoba mengakses `this.asisten.info()` tanpa pengecekan null.
4. Perbaikan: Tambahkan pengecekan null pada method `info()` seperti:

```
if (this.asisten != null) {
    info += "Asisten: " + this.asisten.info() + "\n";
}
```

Percobaan 4

Kode program class Penumpang

```
package athaullahafizh.relasticlass.percobaan4;

public class Penumpang {
    private String ktp;
    private String nama;

    public Penumpang(String ktp, String nama) {
        this.ktp = ktp;
        this.nama = nama;
    }

    public void setKtp(String ktp) {
        this.ktp = ktp;
    }

    public String getKtp() {
        return ktp;
    }

    public void setName(String nama) {
        this.nama = nama;
    }

    public String getName() {
        return nama;
    }

    public String info() {
        String info = "";
        info += "Ktp: " + ktp + "\n";
        info += "Nama: " + nama + "\n";
        return info;
    }
}
```

```
}  
  
}
```

Kode program class Kursi

```
package athaullahafizh.relasticlass.percobaan4;  
  
public class Kursi {  
    private String nomor;  
    private Penumpang penumpang;  
  
    public Kursi(String nomor) {  
        this.nomor = nomor;  
    }  
  
    public void setNomor(String nomor) {  
        this.nomor = nomor;  
    }  
  
    public String getNomor() {  
        return nomor;  
    }  
  
    public void setPenumpang(Penumpang penumpang) {  
        this.penumpang = penumpang;  
    }  
  
    public Penumpang getPenumpang() {  
        return penumpang;  
    }  
  
    public String info() {  
        String info = "";  
        info += "Nomor: " + nomor + "\n";  
        if (this.penumpang != null) {
```



```
        info += "Penumpang: " + penumpang.info() + "\n";
    }
    return info;
}
}
```

Kode program class Gerbong

```
package athaullahafizh.relasticlass.percobaan4;

public class Gerbong {
    private String kode;
    private Kursi[] arrayKursi;

    public Gerbong(String kode, int jumlah) {
        this.kode = kode;
        this.arrayKursi = new Kursi[jumlah];
        this.initKursi();
    }

    public void setKode(String kode) {
        this.kode = kode;
    }

    public String getKode() {
        return kode;
    }

    public void setArrayKursi(Kursi[] arrayKursi) {
        this.arrayKursi = arrayKursi;
    }

    public Kursi[] getArrayKursi() {
        return arrayKursi;
    }
}
```

```

private void initKursi() {
    for (int i = 0; i < arrayKursi.length; i++) {
        this.arrayKursi[i] = new Kursi(String.valueOf(i + 1));
    }
}

public String info() {
    String info = "";
    info += "Kode: " + kode + "\n";
    for (Kursi kursi : arrayKursi) {
        info += kursi.info();
    }
    return info;
}

public void setPenumpang(Penumpang penumpang, int nomor) {
    this.arrayKursi[nomor - 1].setPenumpang(penumpang);
}
}

```

Kode program class MainPercobaan4

```

package athaullahafizh.relasiclass.percobaan4;

public class MainPercobaan4 {
    public static void main(String[] args) {
        Penumpang p = new Penumpang("12345", "Mr. Krab");
        Gerbong gerbong = new Gerbong("A", 10);
        gerbong.setPenumpang(p, 1);
        System.out.println(gerbong.info());
    }
}

```

Output

```
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
```

Pertanyaan

1. Jumlah kursi dalam Gerbong A adalah 10 kursi (sesuai parameter constructor).
2. Kode tersebut mengecek apakah kursi memiliki penumpang atau tidak. Jika ada penumpang (not null), maka informasi penumpang akan ditampilkan. Jika tidak ada (null), maka tidak menampilkan info penumpang.
3. Nilai nomor dikurangi 1 karena array dimulai dari indeks 0, sedangkan nomor kursi dimulai dari 1. Jadi kursi nomor 1 berada di indeks 0, kursi nomor 2 di indeks 1, dst.
4. Yang terjadi adalah objek budi akan menggantikan penumpang sebelumnya (Mr. Krab) di kursi nomor 1, karena tidak ada pengecekan apakah kursi sudah terisi atau belum.

```
Penumpang budi = new Penumpang(ktp:"67890", nama:"Budi");
gerbong.setPenumpang(budi, nomor:1);
System.out.println(x:"Setelah menambah Budi:");
System.out.println(gerbong.info());
```

5. Modifikasi sudah dibuat dengan method setPenumpangSafe() yang mengecek apakah kursi sudah terisi sebelum menempatkan penumpang baru.

Penambahan di Class Gerbong

```
public void setPenumpangSafe(Penumpang penumpang, int nomor) {  
    if (this.arrayKursi[nomor - 1].getPenumpang() == null) {  
        this.arrayKursi[nomor - 1].setPenumpang(penumpang);  
        System.out.println("Penumpang berhasil ditambahkan ke kursi " + nomor);  
    } else {  
        System.out.println("Kursi " + nomor + " sudah terisi!");  
    }  
}
```

Penambahan di Class MainPercobaan4

```
System.out.println(x:"Menggunakan setPenumpangSafe:");  
Penumpang ani = new Penumpang(ktp:"11111", nama:"Ani");  
gerbong.setPenumpangSafe(ani, nomor:1);  
gerbong.setPenumpangSafe(ani, nomor:2);
```

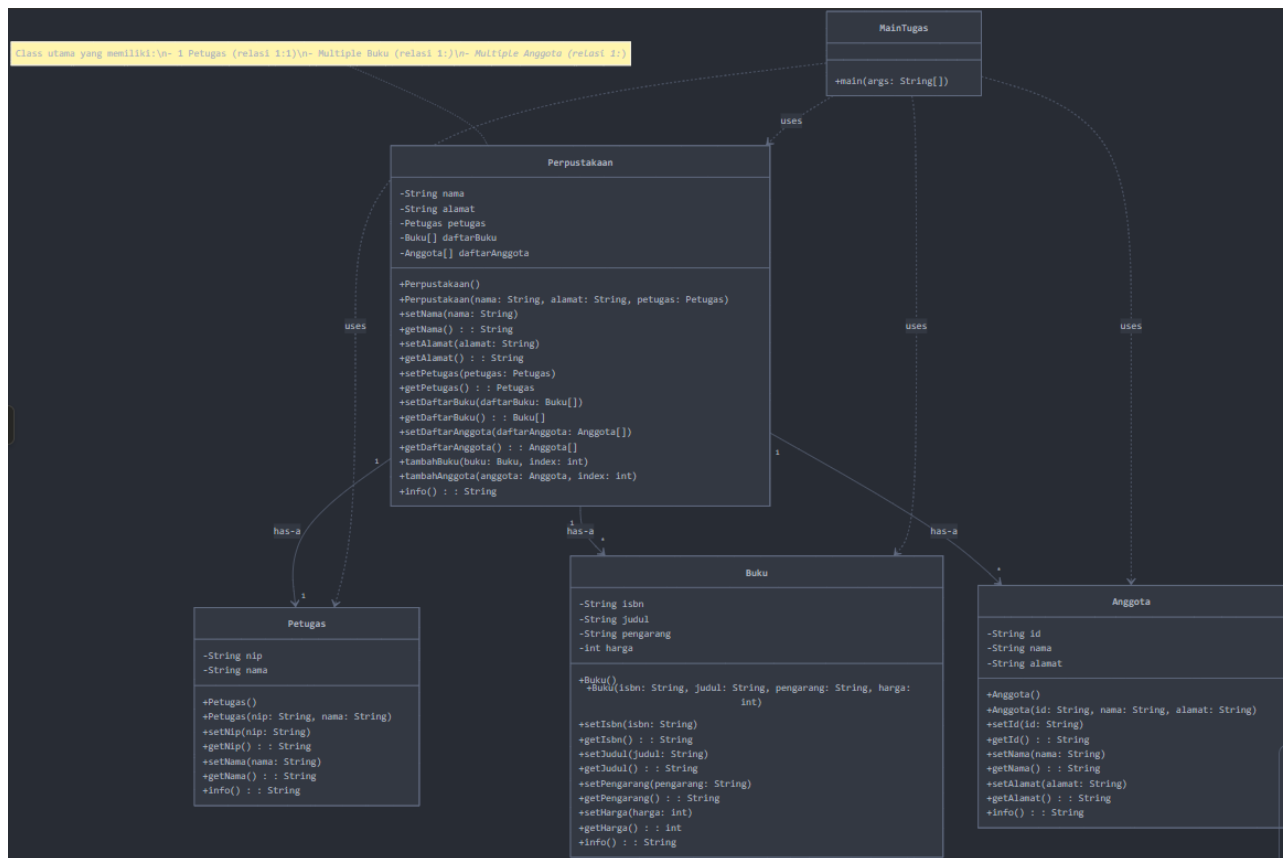
Output / Hasil Akhir

```
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  
  
Setelah menambah Budi:  
Kode: A  
Nomor: 1  
Penumpang: Ktp: 67890  
Nama: Budi  
  
Nomor: 2  
Nomor: 3  
Nomor: 4  
Nomor: 5  
Nomor: 6  
Nomor: 7  
Nomor: 8  
Nomor: 9  
Nomor: 10
```

```
Menggunakan setPenumpangSafe:  
Kursi 1 sudah terisi!  
Penumpang berhasil ditambahkan ke kursi 2
```

Tugas

Class Diagram



Kode program class Buku

```
package athaullahafizh.relasiclass.tugas;

public class Buku {
    private String isbn;
    private String judul;
    private String pengarang;
    private int harga;

    public Buku() {
    }

    public Buku(String isbn, String judul, String pengarang, int
harga) {
        this.isbn = isbn;
    }
}
```

```
        this.judul = judul;
        this.pengarang = pengarang;
        this.harga = harga;
    }

    public void setIsbn(String isbn) {
        this.isbn = isbn;
    }

    public String getIsbn() {
        return isbn;
    }

    public void setJudul(String judul) {
        this.judul = judul;
    }

    public String getJudul() {
        return judul;
    }

    public void setPengarang(String pengarang) {
        this.pengarang = pengarang;
    }

    public String getPengarang() {
        return pengarang;
    }

    public void setHarga(int harga) {
        this.harga = harga;
    }

    public int getHarga() {
        return harga;
    }
}
```

```
public String info() {  
    String info = "";  
    info += "ISBN: " + isbn + "\n";  
    info += "Judul: " + judul + "\n";  
    info += "Pengarang: " + pengarang + "\n";  
    info += "Harga: " + harga + "\n";  
    return info;  
}  
}
```

Kode program class Anggota

```
package athallahafizh.relasticlass.tugas;  
  
public class Anggota {  
    private String id;  
    private String nama;  
    private String alamat;  
  
    public Anggota() {  
    }  
  
    public Anggota(String id, String nama, String alamat) {  
        this.id = id;  
        this.nama = nama;  
        this.alamat = alamat;  
    }  
  
    public void setId(String id) {  
        this.id = id;  
    }  
  
    public String getId() {  
        return id;  
    }  
}
```

```

    public void setNama(String nama) {
        this.nama = nama;
    }

    public String getNama() {
        return nama;
    }

    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }

    public String getAlamat() {
        return alamat;
    }

    public String info() {
        String info = "";
        info += "ID Anggota: " + id + "\n";
        info += "Nama: " + nama + "\n";
        info += "Alamat: " + alamat + "\n";
        return info;
    }
}

```

Kode program class Petugas

```

package athallahafizh.relasticlass.tugas;

public class Petugas {
    private String nip;
    private String nama;

    public Petugas() {
    }
}

```



```
public Petugas(String nip, String nama) {  
    this.nip = nip;  
    this.nama = nama;  
}  
  
public void setNip(String nip) {  
    this.nip = nip;  
}  
  
public String getNip() {  
    return nip;  
}  
  
public void setNama(String nama) {  
    this.nama = nama;  
}  
  
public String getNama() {  
    return nama;  
}  
  
public String info() {  
    String info = "";  
    info += "NIP: " + nip + "\n";  
    info += "Nama Petugas: " + nama + "\n";  
    return info;  
}  
}
```

Kode program class Perpustakaan

```
package athallahafizh.relasticlass.tugas;  
  
public class Perpustakaan {  
    private String nama;
```

```
private String alamat;
private Petugas petugas;
private Buku[] daftarBuku;
private Anggota[] daftarAnggota;

public Perpustakaan() {
    this.daftarBuku = new Buku[10];
    this.daftarAnggota = new Anggota[10];
}

public Perpustakaan(String nama, String alamat, Petugas petugas)
{
    this.nama = nama;
    this.alamat = alamat;
    this.petugas = petugas;
    this.daftarBuku = new Buku[10];
    this.daftarAnggota = new Anggota[10];
}

public void setNama(String nama) {
    this.nama = nama;
}

public String getNama() {
    return nama;
}

public void setAlamat(String alamat) {
    this.alamat = alamat;
}

public String getAlamat() {
    return alamat;
}

public void setPetugas(Petugas petugas) {
```

```
        this.petugas = petugas;
    }

    public Petugas getPetugas() {
        return petugas;
    }

    public void setDaftarBuku(Buku[] daftarBuku) {
        this.daftarBuku = daftarBuku;
    }

    public Buku[] getDaftarBuku() {
        return daftarBuku;
    }

    public void setDaftarAnggota(Anggota[] daftarAnggota) {
        this.daftarAnggota = daftarAnggota;
    }

    public Anggota[] getDaftarAnggota() {
        return daftarAnggota;
    }

    public void tambahBuku(Buku buku, int index) {
        this.daftarBuku[index] = buku;
    }

    public void tambahAnggota(Anggota anggota, int index) {
        this.daftarAnggota[index] = anggota;
    }

    public String info() {
        String info = "";
        info += "Nama Perpustakaan: " + nama + "\n";
        info += "Alamat: " + alamat + "\n";
        info += petugas.info();
    }
}
```

```

        info += "\nDaftar Buku:\n";
        for (Buku buku : daftarBuku) {
            if (buku != null) {
                info += buku.info() + "\n";
            }
        }
        info += "Daftar Anggota:\n";
        for (Anggota anggota : daftarAnggota) {
            if (anggota != null) {
                info += anggota.info() + "\n";
            }
        }
        return info;
    }
}

```

Kode program class MainTugas

```

package athallahafizh.relasticlass.tugas;

public class MainTugas {
    public static void main(String[] args) {
        Petugas p = new Petugas("P001", "Budi Santoso");

        Perpustakaan perpustakaan = new Perpustakaan("Perpustakaan Kota",
            "Jl. Merdeka No. 1", p);

        Buku b1 = new Buku("978-1234567890", "Pemrograman Java",
            "John Doe", 85000);

        Buku b2 = new Buku("978-0987654321", "Basis Data", "Jane
            Smith", 90000);

        Anggota a1 = new Anggota("A001", "Ahmad", "Jl. Sudirman No.
            5");

        Anggota a2 = new Anggota("A002", "Siti", "Jl. Thamrin No.
            10");

        perpustakaan.tambahBuku(b1, 0);
        perpustakaan.tambahBuku(b2, 1);
    }
}

```

```
        perpustakaan.tambahAnggota(a1, 0);  
        perpustakaan.tambahAnggota(a2, 1);  
  
        System.out.println(perpustakaan.info());  
        System.out.println(perpustakaan.getPetugas().getNama());  
    }  
}
```

Output

```
Nama Perpustakaan: Perpustakaan Kota  
Alamat: Jl. Merdeka No. 1  
NIP: P001  
Nama Petugas: Budi Santoso  
  
Daftar Buku:  
ISBN: 978-1234567890  
Judul: Pemrograman Java  
Pengarang: John Doe  
Harga: 85000  
  
ISBN: 978-0987654321  
Judul: Basis Data  
Pengarang: Jane Smith  
Harga: 90000  
  
Daftar Anggota:  
ID Anggota: A001  
Nama: Ahmad  
Alamat: Jl. Sudirman No. 5  
  
ID Anggota: A002  
Nama: Siti  
Alamat: Jl. Thamrin No. 10  
  
Budi Santoso
```