

Nama : I Kadek Angga Dwi Suputra

Nim : 2201010137

Kelas : C

Ujian Akhir Semester

A. Quis 1

Mall Management System adalah sebuah aplikasi yang dirancang untuk mengelola data penyewa, manajer, dan pemesanan di sebuah pusat perbelanjaan. Aplikasi ini bertujuan untuk memudahkan pengelolaan informasi terkait penyewa toko, manajer, dan pemesanan ruang di dalam mall.

- Alasan menggunakan Teori OOP adalah:

1. Encapsulation:

Untuk menjaga keamanan dan integritas data, setiap atribut dalam kelas diatur dengan access modifier yang sesuai (private atau protected), dan menyediakan metode getter dan setter untuk mengakses dan memodifikasi atribut tersebut.

2. Inheritance:

Untuk menghindari duplikasi kode dan mempermudah pengelolaan, kelas **`Orang`** digunakan sebagai superclass untuk kelas **`Penyewa`** dan **`Manajer`**, karena kedua kelas tersebut memiliki atribut dan metode yang sama seperti **`nama`** dan **`alamat`**.

3. Polymorphism:

Untuk memungkinkan pemanggilan metode yang sama pada objek dari kelas yang berbeda, namun dengan perilaku yang berbeda. Misalnya, metode **`tampilkanInfo`** yang di-override pada kelas **`Penyewa`** dan **`Manajer`** untuk menampilkan informasi yang lebih spesifik.

4. Access Modifier:

Untuk mengontrol visibilitas dari atribut dan metode, menggunakan modifier seperti **`private`**, **`protected`**, dan **`public`**.

- Implementasi Project

1. Class **`Orang.java`**

```
public class Orang {  
    protected String nama;  
    protected String alamat;  
    public Orang(String nama, String alamat) {
```

```

        this.nama = nama;

        this.alamat = alamat;
    }

    public String getNama() {
        return nama;
    }

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

    public String getAlamat() {
        return alamat;
    }

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

    public void tampilkanInfo() {
        System.out.println("Nama: " + nama);
        System.out.println("Alamat: " + alamat);
    }
}

```

2. Class **Penyewa.java**

```

public class Penyewa extends Orang {
    private String namaToko;
    private double jumlahSewa;

    public Penyewa(String nama, String alamat, String
namaToko, double jumlahSewa) {
        super(nama, alamat);
        this.namaToko = namaToko;
        this.jumlahSewa = jumlahSewa;
    }

    public String getNamaToko() {
        return namaToko;
    }
}

```

```

    }

    public void setNamaToko(String namaToko) {
        this.namaToko = namaToko;
    }

    public double getJumlahSewa() {
        return jumlahSewa;
    }

    public void setJumlahSewa(double jumlahSewa) {
        this.jumlahSewa = jumlahSewa;
    }

    @Override
    public void tampilkanInfo() {
        super.tampilkanInfo();
        System.out.println("Nama Toko: " + namaToko);
        System.out.println("Jumlah Sewa: " + jumlahSewa);
    }
}

```

3. Class **Manajer.java**

```

public class Manajer extends Orang {
    private String departemen;

    public Manajer(String nama, String alamat, String
departemen) {
        super(nama, alamat);
        this.departemen = departemen;
    }

    public String getDepartemen() {
        return departemen;
    }

    public void setDepartemen(String departemen) {
        this.departemen = departemen;
    }

    public void setujuPemesanan() {

```

```

        System.out.println("Pemesanan disetujui oleh manajer:
" + nama);
    }

    @Override
    public void tampilkanInfo() {
        super.tampilkanInfo();
        System.out.println("Departemen: " + departemen);
    }
}

```

4. Class **Pemesanan.java**

```

public class Pemesanan {
    private String namaPenyewa;
    private String ruangan;
    private String tanggal;

    public Pemesanan(String namaPenyewa, String ruangan,
String tanggal) {
        this.namaPenyewa = namaPenyewa;
        this.ruangan = ruangan;
        this.tanggal = tanggal;
    }

    public String getNamaPenyewa() {
        return namaPenyewa;
    }

    public void setNamaPenyewa(String namaPenyewa) {
        this.namaPenyewa = namaPenyewa;
    }

    public String getRuangan() {
        return ruangan;
    }

    public void setRuangan(String ruangan) {
        this.ruangan = ruangan;
    }
}

```

```

    public String getTanggal() {
        return tanggal;
    }

    public void setTanggal(String tanggal) {
        this.tanggal = tanggal;
    }

    public void tampilkanInfo() {
        System.out.println("Penyewa: " + namaPenyewa);
        System.out.println("Ruangan: " + ruangan);
        System.out.println("Tanggal: " + tanggal);
    }
}

```

5. Class`MallManagement.java`

```

import java.util.ArrayList;
import java.util.Scanner;

public class MallManagement {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        ArrayList<Penyewa> penyewaList = new ArrayList<>();
        ArrayList<Manajer> manajerList = new ArrayList<>();
        ArrayList<Pemesanan> pemesananList = new
ArrayList<>();

        while (true) {

            System.out.print("Apakah Anda ingin menambahkan
penyewa, manajer, atau pemesanan? (1 untuk Penyewa, 2 untuk
Manajer, 3 untuk Pemesanan, 0 untuk Keluar): ");

            int pilihan = scanner.nextInt();

            scanner.nextLine(); // Konsumsi newline

            if (pilihan == 0) {
                break;
            }

            switch (pilihan) {
                case 1:

```

```

        System.out.print("Masukkan nama penyewa:
");

        String namaPenyewa = scanner.nextLine();
        System.out.print("Masukkan alamat penyewa:
");

        String alamatPenyewa = scanner.nextLine();
        System.out.print("Masukkan nama toko: ");
        String namaToko = scanner.nextLine();
        System.out.print("Masukkan jumlah sewa:
");

        double jumlahSewa = scanner.nextDouble();
        scanner.nextLine(); // Konsumsi newline
        penyewaList.add(new Penyewa(namaPenyewa,
alamatPenyewa, namaToko, jumlahSewa));
        break;
    case 2:
        System.out.print("Masukkan nama manajer:
");

        String namaManajer = scanner.nextLine();
        System.out.print("Masukkan alamat manajer:
");

        String alamatManajer = scanner.nextLine();
        System.out.print("Masukkan departemen: ");
        String departemen = scanner.nextLine();
        manajerList.add(new Manajer(namaManajer,
alamatManajer, departemen));
        break;

    case 3:
        System.out.print("Masukkan nama penyewa
untuk pemesanan: ");
        String namaPenyewaPemesanan =
scanner.nextLine();

        System.out.print("Masukkan ruangan: ");
        String ruangan = scanner.nextLine();
        System.out.print("Masukkan tanggal: ");

```

```

        String tanggal = scanner.nextLine();

        pemesananList.add(new
Pemesanan(namaPenyewaPemesanan, ruangan, tanggal));

        break;

        default:

            System.out.println("Pilihan tidak valid.
Silakan coba lagi.");
    }
}

System.out.println("\nDaftar Penyewa:");
for (Penyewa penyewa : penyewaList) {
    penyewa.tampilkanInfo();
    System.out.println("-----");
}

System.out.println("\nDaftar Manajer:");
for (Manajer manajer : manajerList) {
    manajer.tampilkanInfo();
    System.out.println("-----");
}

System.out.println("\nDaftar Pemesanan:");
for (Pemesanan pemesanan : pemesananList) {
    pemesanan.tampilkanInfo();
    System.out.println("-----");
}

scanner.close();
}
}

```

B. Quis 2

1. Implementasi Desain aplikasi dari Quis 1

Aplikasi manajemen mall yang mengelola data penyewa, manajer, dan pemesanan ruangan. Program ini menggunakan konsep Pemrograman Berorientasi Objek (OOP) dengan kelas-kelas seperti `Orang`, `Penyewa`, `Manajer`, dan `Pemesanan`.

- Class `Orang` adalah kelas dasar yang menyimpan informasi umum seperti nama dan alamat. Kelas ini memiliki metode untuk mengakses (getter dan setter) dan menampilkan informasi (metode `tampilkanInfo`). Kelas `Penyewa` dan `Manajer` merupakan subclass dari `Orang` yang mewarisi properti dan metode dari kelas dasar tersebut.
- Class `Penyewa` menambahkan properti `namaToko` dan `jumlahSewa`, serta metode untuk mengakses dan mengubah properti ini. Kelas ini juga meng-override metode `tampilkanInfo` untuk menampilkan informasi tambahan tentang penyewa. Kelas `Manajer` menambahkan properti `departemen` dan metode `setujuiPemesanan` untuk menyetujui pemesanan. Metode `tampilkanInfo` juga di-override untuk menampilkan informasi departemen manajer.
- Class `Pemesanan` menyimpan informasi tentang pemesanan ruangan oleh penyewa, termasuk nama penyewa, ruangan yang dipesan, dan tanggal pemesanan. Kelas ini memiliki metode untuk mengakses dan mengubah informasi tersebut serta menampilkan informasi pemesanan.
- Dalam metode `main`, program menggunakan `Scanner` untuk membaca input dari pengguna dan menyimpannya dalam daftar (ArrayList) yang sesuai (`penyewaList`, `manajerList`, `pemesananList`). Pengguna dapat menambahkan penyewa, manajer, atau pemesanan melalui serangkaian prompt, dan program akan menyimpan informasi yang diberikan dalam objek yang sesuai dan menambahkannya ke daftar yang relevan. Setelah pengguna selesai menambahkan data (dengan memilih untuk keluar dari loop), program akan menampilkan semua penyewa, manajer, dan pemesanan yang telah dimasukkan. Setiap objek dalam daftar akan memanggil metode `tampilkanInfo` untuk menampilkan detailnya.

2. Source code

```
package mallmanagement;

/**
 *
 * @author kadek
 * I Kadek Angga Dwi Suputra
 * 2201010137
 */
import java.util.ArrayList;
import java.util.Scanner;

public class MallManagement {

    // Kelas Person
    static class Orang {
```



```

        protected String nama;
        protected String alamat;

        public Orang(String nama, String alamat) {
            this.nama = nama;
            this.alamat = alamat;
        }

        public String getNama() {
            return nama;
        }

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

        public String getAlamat() {
            return alamat;
        }

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

        public void tampilkanInfo() {
            System.out.println("Nama: " + nama);
            System.out.println("Alamat: " + alamat);
        }
    }

    // Kelas Tenant
    static class Penyewa extends Orang {
        private String namaToko;
        private double jumlahSewa;

        public Penyewa(String nama, String alamat, String
namaToko, double jumlahSewa) {
            super(nama, alamat);
            this.namaToko = namaToko;
            this.jumlahSewa = jumlahSewa;
        }

        public String getNamaToko() {
            return namaToko;
        }

        public void setNamaToko(String namaToko) {
            this.namaToko = namaToko;
        }

        public double getJumlahSewa() {

```

```

        return jumlahSewa;
    }

    public void setJumlahSewa(double jumlahSewa) {
        this.jumlahSewa = jumlahSewa;
    }

    @Override
    public void tampilkanInfo() {
        super.tampilkanInfo();
        System.out.println("Nama Toko: " + namaToko);
        System.out.println("Jumlah Sewa: " + jumlahSewa);
    }
}

// Kelas Manager
static class Manajer extends Orang {
    private String departemen;

    public Manajer(String nama, String alamat, String
departemen) {
        super(nama, alamat);
        this.departemen = departemen;
    }

    public String getDepartemen() {
        return departemen;
    }

    public void setDepartemen(String departemen) {
        this.departemen = departemen;
    }

    public void setujuipemesanan() {
        System.out.println("Pemesanan        disetujui        oleh
manajer: " + nama);
    }

    @Override
    public void tampilkanInfo() {
        super.tampilkanInfo();
        System.out.println("Departemen: " + departemen);
    }
}

// Kelas Booking
static class Pemesanan {
    private String namaPenyewa;
    private String ruangan;
    private String tanggal;

```

```

        public Pemesanan(String namaPenyewa, String ruangan,
String tanggal) {
            this.namaPenyewa = namaPenyewa;
            this.ruangan = ruangan;
            this.tanggal = tanggal;
        }

        public String getNamaPenyewa() {
            return namaPenyewa;
        }

        public void setNamaPenyewa(String namaPenyewa) {
            this.namaPenyewa = namaPenyewa;
        }

        public String getRuangan() {
            return ruangan;
        }

        public void setRuangan(String ruangan) {
            this.ruangan = ruangan;
        }

        public String getTanggal() {
            return tanggal;
        }

        public void setTanggal(String tanggal) {
            this.tanggal = tanggal;
        }

        public void tampilkanInfo() {
            System.out.println("Penyewa: " + namaPenyewa);
            System.out.println("Ruangan: " + ruangan);
            System.out.println("Tanggal: " + tanggal);
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        ArrayList<Penyewa> penyewaList = new ArrayList<>();
        ArrayList<Manajer> manajerList = new ArrayList<>();
        ArrayList<Pemesanan> pemesananList = new ArrayList<>();

        while (true) {
            System.out.print("Apakah Anda ingin menambahkan
penyewa, manajer, atau pemesanan? (1 untuk Penyewa, 2 untuk
Manajer, 3 untuk Pemesanan, 0 untuk Keluar): ");
            int pilihan = scanner.nextInt();
            scanner.nextLine(); // Konsumsi newline

```

```

        if (pilihan == 0) {
            break;
        }

        switch (pilihan) {
            case 1:
                System.out.print("Masukkan nama penyewa:");

                String namaPenyewa = scanner.nextLine();
                System.out.print("Masukkan alamat penyewa:");

                String alamatPenyewa = scanner.nextLine();
                System.out.print("Masukkan nama toko: ");
                String namaToko = scanner.nextLine();
                System.out.print("Masukkan jumlah sewa: ");
                double jumlahSewa = scanner.nextDouble();
                scanner.nextLine(); // Konsumsi newline
                penyewaList.add(new Penyewa(namaPenyewa,
alamatPenyewa, namaToko, jumlahSewa));
                break;
            case 2:
                System.out.print("Masukkan nama manajer:");

                String namaManajer = scanner.nextLine();
                System.out.print("Masukkan alamat manajer:");

                String alamatManajer = scanner.nextLine();
                System.out.print("Masukkan departemen: ");
                String departemen = scanner.nextLine();
                manajerList.add(new Manajer(namaManajer,
alamatManajer, departemen));
                break;
            case 3:
                System.out.print("Masukkan nama penyewa
untuk pemesanan: ");
                String namaPenyewaPemesanan =
scanner.nextLine();
                System.out.print("Masukkan ruangan: ");
                String ruangan = scanner.nextLine();
                System.out.print("Masukkan tanggal: ");
                String tanggal = scanner.nextLine();
                pemesananList.add(new
Pemesanan(namaPenyewaPemesanan, ruangan, tanggal));
                break;
            default:
                System.out.println("Pilihan tidak valid.
Silakan coba lagi.");
        }
    }

    System.out.println("\nDaftar Penyewa:");

```

```

        for (Penyewa penyewa : penyewaList) {
            penyewa.tampilkanInfo();
            System.out.println("-----");
        }

        System.out.println("\nDaftar Manajer:");
        for (Manajer manajer : manajerList) {
            manajer.tampilkanInfo();
            System.out.println("-----");
        }

        System.out.println("\nDaftar Pemesanan:");
        for (Pemesanan pemesanan : pemesananList) {
            pemesanan.tampilkanInfo();
            System.out.println("-----");
        }

        scanner.close();
    }
}

```