

# **PEMROGRAMAN BERBASIS OBJEK**



## **ASSESMENT 1 PRAKTEK**

**Disusun oleh:**

Zidane Ahmad Firdaus / 607062300124

**TELKOM UNIVERSITY**

**BANDUNG**

**2024**

## Class Travel:

```
1 import java.util.ArrayList;
2
3 public class Travel {
4     private String paket;
5     private Kendaraan kendaraan;
6     private Hotel hotel;
7     private ArrayList<TempatWisata> tempatWisatalist = new ArrayList<>();
8     private ArrayList<String> travellerList = new ArrayList<>();
9
10    public Travel(String paket, Kendaraan kendaraan, Hotel hotel) {
11        this.paket = paket;
12        this.kendaraan = kendaraan;
13        this.hotel = hotel;
14    }
15
16    public void addWisata(TempatWisata tempatWisata) {
17        tempatWisatalist.add(tempatWisata);
18    }
19
20    public void addTraveller(String namaTraveller) {
21        travellerList.add(namaTraveller);
22    }
23
24    public void displayData() {
25        System.out.println("Nama Paket: " + paket);
26        System.out.println("Kendaraan: " + kendaraan.getJenis() + " (Kapasitas: " + kendaraan.getKapasitas() + ")");
27        System.out.println("Penginapan: " + hotel.getNama() + " (Biaya: " + hotel.getBiayaPerMalam() + " Beri )");
28
29        System.out.println(x:"\nDaftar Tempat Wisata:");
30        for (TempatWisata tw : tempatWisatalist) {
31            System.out.println("- " + tw.getNama() + " (Biaya: " + tw.getBiaya() + " Beri )");
32
33            System.out.println("- " + tw.getNama() + " (Biaya: " + tw.getBiaya() + " Beri )");
34        }
35
36        System.out.println(x:"\nDaftar Traveller:");
37        for (String traveller : travellerList) {
38            System.out.println("- " + traveller);
39        }
40    }
41
42    public String getPaket() {
43        return paket;
44    }
45
46    public Kendaraan getKendaraan() {
47        return kendaraan;
48    }
49
50    public Hotel getHotel() {
51        return hotel;
52    }
53
54    public ArrayList<TempatWisata> getTempatWisatalist() {
55        return tempatWisatalist;
56    }
57
58    public ArrayList<String> getTravellerList() {
59        return travellerList;
60    }
61 }
```

## Class Main:

```

public class main {
    Run | Debug
    public static void main(String[] args) {
        //pembuatan objek tempat wisata
        TempatWisata tw1 = new TempatWisata (nama:"Dressrosa", biaya:500);
        TempatWisata tw2 = new TempatWisata (nama:"Water Seven", biaya:300);
        TempatWisata tw3 = new TempatWisata (nama:"Sabaody Archipelago", biaya:200);

        //pembuatan objek untuk traveller
        Traveller t11 = new Traveller(nama:"Luffy", asal:"East Blue");
        Traveller t12 = new Traveller(nama:"Zoro", asal:"East Blue");
        Traveller t13 = new Traveller(nama:"Nami", asal:"East Blue");
        Traveller t14 = new Traveller(nama:"Usopp", asal:"East Blue");
        Traveller t15 = new Traveller(nama:"Sanji", asal:"Nort Blue");
        Traveller t16 = new Traveller(nama:"Chopper", asal:"Paradise");
        Traveller t17 = new Traveller(nama:"Robin", asal:"West Blue");
        Traveller t18 = new Traveller(nama:"Franky", asal:"Paradise");
        Traveller t19 = new Traveller(nama:"Brook", asal:"West Blue");
        Traveller t110 = new Traveller(nama:"Jinbe", asal:"Paradise");

        //pembuatan objek Travel Pertama
        //berelasi komposisi dengan class Kendaraan dan htel
        Travel tr1 = new Travel(paket:"Paket 1" , new Kendaraan (jenis:"Caravel" , kapasitas:50), new Hotel(nama:"Alabasta" , biaya:100));

        //menambahkan tempat wisata pada objek tr1
        //berelasi agregasi dengan class tempatwisata
        tr1.addWisata(tw1);
        tr1.addWisata(tw2);
    }
}

```

```

//menambahkan tempat wisata pada objek tr1
//berelasi agregasi dengan class tempatwisata
tr1.addWisata(tw1);
tr1.addWisata(tw2);

//menambahkan traveller pada objek tr1
//berelasi asosiasi dengan class traveller
tr1.addTraveller(t11.getNama());
tr1.addTraveller(t12.getNama());
tr1.addTraveller(t13.getNama());
tr1.addTraveller(t14.getNama());
tr1.addTraveller(t15.getNama());
tr1.addTraveller(t16.getNama());

//pambuatan untuk objek travel ke 2
//berelasi komposisi dengan class kendaraan hotel
Travel tr2 = new Travel (paket:"Paket 2", new Kendaraan (jenis:"Cruise Ship", kapasitas:100), new Hotel (nama:"Mermelada" , biaya:150));

//menambahkan tempat wisata pada objek tr2
//berelasi agregasi dengan class TempatWisata
tr2.addWisata(tw1);
tr2.addWisata(tw2);
tr2.addWisata(tw3);

//menambahkan traveller pada objektr2
//berelasi asosiasi dengan class traveller
tr2.addTraveller(t17.getNama());
tr2.addTraveller(t18.getNama());
tr2.addTraveller(t19.getNama());

```

```

//menambahkan traveller pada objektr2
//berelasi asosiasi dengan class traveller
tr2.addTraveller(t17.getNama());
tr2.addTraveller(t18.getNama());
tr2.addTraveller(t19.getNama());
tr2.addTraveller(t110.getNama());

System.out.println(x:"\n\n-----Data ke 1 -----");
tr1.displayData();
//menampilkan data paket1
System.out.println(x:"\n\n-----Data ke 2 -----");
tr2.displayData();

```

Class Traveller:

```
1
2 public class Traveller {
3     private String nama;
4     private String asal;
5
6     public Traveller(String nama, String asal) {
7         this.nama = nama;
8         this.asal = asal;
9     }
10
11     public String getNama() {
12         return nama;
13     }
14
15     public String getAsal() {
16         return asal;
17     }
18 }
19
```

Class TempatWisata:

```
✓ public class TempatWisata {
    private String nama;
    private int biaya;
2
3     public TempatWisata(String nama, int biaya) {
4         this.nama = nama;
5         this.biaya = biaya;
6     }
7
8     public String getNama() {
9         return nama;
10    }
11
12    public int getBiaya() {
13        return biaya;
14    }
15 }
16
```

Class Kendaraan:

```
public class Kendaraan {
    private String jenis;
    private int kapasitas;
2
3     public Kendaraan(String jenis, int kapasitas) {
4         this.jenis = jenis;
5         this.kapasitas = kapasitas;
6     }
7
8     public String getJenis() {
9         return jenis;
10    }
11
12    public int getKapasitas() {
13        return kapasitas;
14    }
15 }
16
```

Class Hotel:

```
public class Hotel {  
    private String nama;  
    private int biayaPerMalam;  
  
    public Hotel(String nama, int biayaPerMalam) {  
        this.nama = nama;  
        this.biayaPerMalam = biayaPerMalam;  
    }  
  
    public String getNama() {  
        return nama;  
    }  
  
    public int getBiayaPerMalam() {  
        return biayaPerMalam;  
    }  
}
```

SS HASIL:

```
-----Data ke 1 -----  
Nama Paket: Paket 1  
Kendaraan: Caravel (Kapasitas: 50)  
Penginapan: Alabasta Hotel (Biaya per malam: 50 Beri )
```

```
Daftar Tempat Wisata:  
- Dressrosa (Biaya: 500 Beri )  
- Water Seven (Biaya: 300 Beri )
```

Daftar Traveller:

```
- Luffy  
- Zoro  
- Nami  
- Usopp  
- Sanji  
- Chopper
```

```
-----Data ke 2 -----  
Nama Paket: Paket 2  
Kendaraan: Cruise Ship (Kapasitas: 100)  
Penginapan: Mermaid Hotel (Biaya per malam: 40 Beri )
```

```
Daftar Tempat Wisata:  
- Dressrosa (Biaya: 500 Beri )  
- Water Seven (Biaya: 300 Beri )  
- Sabaody Archipelago (Biaya: 200 Beri )
```

Daftar Traveller:

```
- Robin  
- Franky  
- Brook  
- Jinbe
```

```
PS D:\KULIAH ZIDANE\SEMESTER 3\PBO\PBO2024> █
```

