

Laporan Praktikum Pemograman Berbasis Objek

Disusun guna melengkapi tugas mata kuliah Laporan

Praktikum Pemograman Berbasis Objek

Yang diampu oleh:

Dian Wahyuningsih, S.Kom. MMSI.



Dan disusun oleh:

Febrian Perdana

2341720150

KELAS TI 2 E

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS TEKNIK INFORMATIKA POLITEKNIK

NEGERI MALANG

2024

Pertanyaan:

1. Saat Anda membuat konstruktor untuk melaksanakan instruksi nomor 4, berapa maksimal parameter yang dapat kalian gunakan? Mengapa?
2. Jalankan aplikasi yang telah Anda kerjakan dan buat objek hp lalu panggil method info untuk mengetahui posisi awal objek hp tersebut.
3. Lakukan pemanggilan method tambah volume. Apa yang terjadi? Mengapa demikian?
4. Lakukan pemanggilan method tambahVolume sampai batas maksimal yakni 100, kemudian panggil lagi method tambahVolume. Apa yang terjadi? Mengapa demikian?
5. Lakukan hal yang sama untuk method kurangVolume sampai batas minimal yakni 0, kemudian panggil lagi method kurangVolume. Apa yang terjadi? Mengapa demikian?
6. Buktikan bahwa saat objek hp dalam keadaan power on, kita dapat memanggil method mute, sehingga volume akan berubah menjadi 0.
7. Dan panggil kembali method mute untuk mengembalikan volume ke angka terakhir kali dinyatakan.
8. Kumpulkan jawaban beserta file Handphone.java dan DemoHp.java di drive dengan folder kuis1 (jadikan 1 folder atau dapat di zip/rar).

Jawaban

1. 6 dikarenakan saya menggunakan 6 atribut.

```
Informasi HandPhone:  
No. Seri: 123456  
Model: Samsung Galaxy  
Versi OS: Android 12  
Status: Mati  
Volume: 0
```

- 2.
3. Volume bertambah 5 dikarenakan

```
// Metode untuk menambah volume  
Codeium: Refactor | Explain | X  
public void tambahVolume() {  
    volume = currentVol; // Metode untuk unmute sambil menambah volume  
    if (on && volume < 100) {  
        volume += 5;  
        System.out.println("Volume ditambah menjadi: " + volume);  
        currentVol = volume;  
    } else if (!on) {  
        System.out.println(x: "Handphone mati. Tidak dapat mengubah volume.");  
    } else {  
        System.out.println(x: "Volume sudah maksimal.");  
    }  
}
```

4. Muncul teks Volume sudah maksimal dikarenakan

```
// Metode untuk menambah volume
Codeium: Refactor | Explain | X
public void tambahVolume() {
    volume = currentVol; // Metode untuk unmute sambil menambah volume
    if (on && volume < 100) {
        volume+= 5;
        System.out.println("Volume ditambah menjadi: " + volume);
        currentVol = volume;
    } else if (!on) {
        System.out.println(x:"Handphone mati. Tidak dapat mengubah volume.");
    } else {
        System.out.println(x:"Volume sudah maksimal.");
    }
}
```

5. Muncul teks volume sudah minimal diakarenakan

```
// Metode untuk mengurangi volume
Codeium: Refactor | Explain | X
public void kurangVolume() {
    volume = currentVol; // Metode untuk unmute sambil mengurangi volume
    if (on && volume > 0) {
        volume-= 5;
        System.out.println("Volume dikurangi menjadi: " + volume);
        currentVol = volume;
    } else if (!on) {
        System.out.println(x:"Handphone mati. Tidak dapat mengubah volume.");
    } else {
        System.out.println(x:"Volume sudah minimal.");
    }
}
```

```
Handphone dimute. Volume sekarang: 0
```

```
Menu
```

1. Info
2. Power
3. Tambah Volume
4. Kurang Volume
5. Mute
6. Unmute
7. Keluar

```
Pilih Operasi:
```

```
6
```

```
Handphone di-unmute. Volume sekarang: 30
```

```
Menu
```

1. Info
2. Power
3. Tambah Volume
4. Kurang Volume
5. Mute
6. Unmute
7. Keluar

```
Pilih Operasi:
```

```
■
```

6.

```
Handphone dimute. Volume sekarang: 0
```

```
Menu
```

1. Info
2. Power
3. Tambah Volume
4. Kurang Volume
5. Mute
6. Unmute
7. Keluar

```
Pilih Operasi:
```

```
6
```

```
Handphone di-unmute. Volume sekarang: 30
```

```
Menu
```

1. Info
2. Power
3. Tambah Volume
4. Kurang Volume
5. Mute
6. Unmute
7. Keluar

```
Pilih Operasi:
```

```
■
```

7.