

TUGAS
PEMROGRAMAN BERBASIS OBJEK



MUHAMMAD RAYHAN ZAMZAMI

KELAS TI 1-H

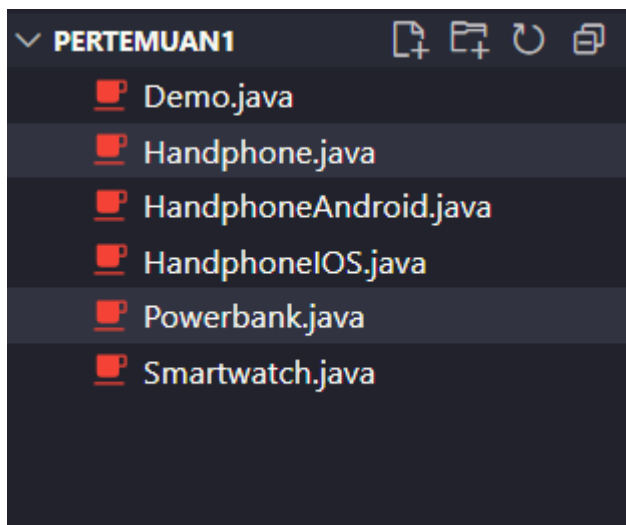
244107020027

TEKNIK INFORMATIKA

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG 2024

1. Disini saya menggunakan objek Handphone Android, HandphoneIOS(yang mewarisi class handphone), Smartwatch serta Powerbank jadi akan ada 5 Class Termasuk Handphone dan Demo sebagai Main



2. Pertama Kita akan membuat Class Handphone sebagai super class

```
Windsurf: Refactor | Explain
1 class Handphone {
2
3     String merk;
4     int kapasitasBaterai;
5
6     public Handphone(String merk, int kapasitasBaterai){
7         this.merk = merk;
8         this.kapasitasBaterai = kapasitasBaterai;
9     }
10
11     Windsurf: Refactor | Explain | Generate Javadoc | X
12     void nyalakan(){
13         System.out.println(merk + " nyala");
14     }
15
16     Windsurf: Refactor | Explain | Generate Javadoc | X
17     void matikan(){
18         System.out.println(merk + " mati");
19     }
20
21     Windsurf: Refactor | Explain | Generate Javadoc | X
22     void cetakStatus(){
23         System.out.println("Merk : " + merk);
24         System.out.println("Kapasitas Baterai : " + kapasitasBaterai + " mAh");
25     }
26 }
```

3. Selanjutnya membikin 2 class dengan konsep pewarisan

```
HandphoneAndroid.java > HandphoneAndroid > jumlahKamera
Windsurf: Refactor | Explain
1 class HandphoneAndroid extends Handphone{
2     String versiAndroid;
3     int jumlahKamera;
4
5     public HandphoneAndroid(String merk, int kapasitasBaterai, String versiAndroid, int jumlahKamera){
6         super(merk, kapasitasBaterai);
7         this.versiAndroid = versiAndroid;
8         this.jumlahKamera = jumlahKamera;
9     }
10
11     void bukaPlaystore(){
12         System.out.println(merk + " membuka Playstore");
13     }
14
15     void multiTasking(){
16         System.out.println(merk + " sedang melakukan multitasking di " + versiAndroid);
17     }
18
19     void cetakStatus(){
20         super.cetakStatus();
21         System.out.println("Versi Android : " + versiAndroid);
22         System.out.println("Jumlah Kamera : " + jumlahKamera);
23     }
24 }
25
```

```
HandphoneIOS.java > HandphoneIOS > gunakanUltraWide()
Windsurf: Refactor | Explain
1 class HandphoneIOS extends Handphone {
2     String versiIOS;
3     boolean ultraWideKamera;
4     int jumlahKamera;
5
6     public HandphoneIOS(String merk, int kapasitasBaterai, String versiIOS, int jumlahKamera){
7         super(merk, kapasitasBaterai);
8         this.versiIOS = versiIOS;
9         this.jumlahKamera = jumlahKamera;
10    }
11
12    void bukaAppStore(){
13        System.out.println(merk + " membuka App Store");
14    }
15
16    void gunakanUltraWide(){
17        if (ultraWideKamera == true){
18            System.out.println(merk + " sedang menggunakan kamera ultra wide");
19        }
20        else{
21            System.out.println(merk + " tidak memiliki kamera ultra wide");
22        }
23    }
24
25    void cetakStatus(){
26        super.cetakStatus();
27        System.out.println("Versi IOS : " + versiIOS);
28        System.out.println("Jumlah Kamera : " + jumlahKamera);
29        System.out.println("Ultra Wide Kamera : " + (ultraWideKamera? "ada" : "tidak ada"));
30    }
31 }
32
```

4. Setelah itu saya membuat class lain yang tidak mengandung konsep pewarisan

```
Smartwatch.java 7 Smartwatch 7 cetakStatus()
Windsurf: Refactor | Explain
1 class Smartwatch {
2     String merk;
3     int tahananBaterai;
4
5     public Smartwatch(String merk, int tahananBaterai) {
6         this.merk = merk;
7         this.tahananBaterai = tahananBaterai;
8     }
9
10    void hitungDetakJantung(){
11        System.out.println(merk + " menghitung detak jantung");
12    }
13
14    void TampilkanNotifikasi(String pesan){
15        System.out.println(merk + " menampilkan notifikasi" + pesan);
16    }
17
18
19    void tampilkanWaktu(){
20        System.out.println(merk + " menampilkan waktu");
21    }
22
23    void cetakStatus(){
24        System.out.println("Merk : " + merk);
25        System.out.println("Tahanan Baterai : " + tahananBaterai + " Jam");
26    }
27 }
28
Windsurf: Refactor | Explain
1 class Powerbank {
2     String merk;
3     int Kapasitas;
4
5     public Powerbank(String merk, int Kapasitas) {
6         this.merk = merk;
7         this.Kapasitas = Kapasitas;
8     }
9
10    void isiDaya(){
11        System.out.println("Powerbank " + merk + " sedang mengisi daya handphone");
12    }
13
14    void cekBaterai(){
15        System.out.println("Powerbank " + merk + " kapasitas: " + Kapasitas + " mAh");
16    }
17
18    void cetakStatus(){
19        System.out.println("Merk : " + merk);
20        System.out.println("Kapasitas Baterai : " + Kapasitas + " mAh");
21    }
22 }
23
```

5. Dan terakhir class demo sebagai class main untuk menjalankan program

```
Windsurf: Refactor | Explain
public class Demo {
    Run | Debug | Windsurf: Refactor | Explain | Generate Javadoc | X
    public static void main(String[] args) {
        HandphoneAndroid android1 = new HandphoneAndroid(merk:"Samsung", kapasitasBaterai:5000, versiAndroid:"Android 10", jumlahKa...2);
        HandphoneIOS ios1 = new HandphoneIOS(merk:"iPhone", kapasitasBaterai:5000, versiIOS:"iOS 14", jumlahKamera...3);
        Smartwatch smartwatch1 = new Smartwatch(merk:"Xiaomi", tahananBaterai:4);
        Powerbank pb1 = new Powerbank(merk:"Sony", Kapasitas:5000);

        android1.nyalakan();
        android1.bukaPlaystore();
        android1.multiTasking();
        android1.cetakStatus();
        android1.matikan();

        System.out.println();

        ios1.nyalakan();
        ios1.bukaAppStore();
        ios1.gunakanUltraWide();
        ios1.cetakStatus();
        ios1.matikan();

        System.out.println();

        smartwatch1.tampilkanWaktu();
        smartwatch1.hitungDetakJantung();
        smartwatch1.TampilkanNotifikasi(pesan:"Pesan Whatsapp Masuk");
        smartwatch1.TampilkanNotifikasi(pesan:"Pesan Instagram Masuk");
        smartwatch1.cetakStatus();

        System.out.println();

        pb1.cekBaterai();
        pb1.isiDaya();
        pb1.cetakStatus();

        System.out.println();
    }
}
```

6. Output dari program di atas sebagai berikut

```
ws\Pertemuan1_f028903b\bin\Demo
Samsung nyala
Samsung membuka Playstore
Samsung sedang melakukan multitasking di Android 10
Merk : Samsung
Kapasitas Baterai : 5000 mAh
Versi Android : Android 10
Jumlah Kamera : 2
Samsung mati

iPhone nyala
iPhone membuka App Store
iPhone tidak memiliki kamera ultra wide
Merk : iPhone
Kapasitas Baterai : 5000 mAh
Versi IOS : iOS 14
Jumlah Kamera : 3
Ultra Wide Kamera : tidak ada
iPhone mati

Xiaomi menampilkan waktu
Xiaomi menghitung detak jantung
Xiaomi menampilkan notifikasiPesan Whatsapp Masuk
Xiaomi menampilkan notifikasiPesan Instagram Masuk
Merk : Xiaomi
Tahanan Baterai : 4 Jam

Powerbank Sony kapasitas: 5000 mAh
Powerbank Sony sedang mengisi daya handphone
Merk : Sony
Kapasitas Baterai : 5000 mAh
```

Source Code atau file lengkap dari tugas : <https://github.com/mrayhanz/program-berbasis-objek/tree/main/Pertemuan1>