

LAPORAN PRAKTIKUM MODUL 6
OVERRIDING DAN DOWNCASTING



DISUSUN OLEH :

MAHRAN RADIFAN ZHAFIR (105219001)

PRODI ILMU KOMPUTER
UNIVERSITAS PERTAMINA
JAKARTA SELATAN

2021

1. Pendahuluan

Overriding adalah salah satu metode yang digunakan untuk mengubah method yang dimiliki oleh parent sebuah child. Metode override digunakan dengan mengganti isi dari nama method yang sama yang ada pada child. Penggunaan overriding sangat berguna apabila terdapat perubahan pada method apabila child digunakan pada saat pemanggilan method dan perlu perubahan.

Downcasting adalah penurunan nilai atribut dari parent ke sebuah child. Metode ini berguna untuk menurunkan nilai atribut sewaktu waktu digunakan oleh childnya, sehingga atribut yang sudah ada di parent dapat digunakan pada childnya lagi

2. Pembahasan

```
public class superclass1 {  
  
    public void mirage(){  
        System.out.println("now you see me, now you dont");  
        System.out.println("Parent Method");  
    }  
    public void wraith(){  
        System.out.println("There is a void in your heart");  
        System.out.println("Parent Method");  
    }  
}
```

Gambar 1.1 Superclass yang akan di override

```

public class override extends superclass1 {
    String senjata;

    public void setSenjata(String senjata){
        this.senjata = senjata;
    }
    public String getSenjata(){
        return senjata;
    }
    public void senjata(){
        System.out.println(senjata);
    }
}

@Override
public void mirage(){
    System.out.println("nah boy you gettin' killed with " + senjata);
    System.out.println("Child Method");
}

public void wraith(){
    System.out.println("Good choice " + senjata + " ,for me");
    System.out.println("Child Method");
}
}

```

Gambar 1.2 Child yang melakukan override terhadap parent nya

Pada saat pemanggilan mirage() dan wraith(), harusnya child akan mengeluarkan string yang terdapat pada parent, tapi karena di override, terdapat perubahan fungsi dengan nama yang sama pada child.

Berikut adalah pemanggilan fungsi pada method override

```

public static void main(String[] args) throws Exception {
    String temp;
    int angka;
    superclass1 apex = new superclass1();
    override apexWeapon = new override();
    System.out.println("=====Override=====\n");
    apex.mirage();
    apex.wraith();
    temp = apexWeapon.senjata;
    if(temp == null){
        apex.mirage();
    } else {
        apexWeapon.mirage();
    }
    apexWeapon.setSenjata("R-99");
    temp = apexWeapon.senjata;
    if(temp == null){
        apex.mirage();
    } else {
        apexWeapon.mirage();
    }
}

```

Gambar 1.3 Pemanggilan fungsi pada main.java

```

public class superclass2 {
    String[] weapon = {"Operator" , "Vandal" , "Phantom"};
    public void weaponShow(){
        for (int i = 0; i < weapon.length; i++){
            System.out.println(weapon[i]);
        }
    }
    public void weaponExactShow(int i){
        if (i < weapon.length){
            System.out.println(weapon[i]);
            System.out.println("Parent Downcast");
        }
    }
}

```

Gambar 1.4 Penambahan atribut pada superclass untuk downcast

```

public class downcast extends superclass2 {
    public void showText(){
        System.out.println("boom, you are just getting outplayed");
        System.out.println("Child downcast");
    }
}

```

Gambar 1.5 Child dari superclass yang akan di downcast

```

superclass2 valorantChild = new downcast();
downcast child = (downcast)valorantChild;

```

Gambar 1.6 Downcasting parent ke child class

```

child.showText();
System.out.println(child.weapon[1]);
input.close();

```

Gambar 1.7 Pemanggilan variable yang diturunkan dari parent ke child

3. Kesimpulan

Pada overriding, terjadi perubahan fungsi pada method yang di wariskan oleh parent class terhadap child. Pada praktikum kali ini dapat dilihat bahwa saat memanggil nama method yang sama, tidak terjadi bentrok antara dua method, melainkan method yang sedang melakukan override akan menimpa method yang sudah ada.

Saat downcasting, child tidak mempunyai attribut apapun. Tapi saat sudah di downcasting, attribute yang dimiliki oleh parent dapat digunakan oleh child. Ini berguna apabila atribut pada parent sering digunakan oleh child nya itu sendiri