

Nama : Ai Dina Agustin

NIM : 20210040065

Kelas : TI21A

Laporan Praktikum Pemrograman Berorientasi Objek

Percobaan 7

- **Script program**

```
class Bapak{

    int a;

    int b;


    void show_variabel(){

        System.out.println("Nilai a="+ a);

        System.out.println("Nilai b="+ b);

    }

}

class Anak extends Bapak{

    int c;

    void show_variabel(){

        System.out.println("Nilai a="+ a);

        System.out.println("Nilai b="+ b);

        System.out.println("Nilai c="+ c);

    }

}
```

```

public class InheritExample {

    public static void main(String[] args) {

        Bapak objectBapak = new Bapak();

        Anak objectAnak = new Anak();


        objectBapak.a=1;

        objectBapak.b=1;

        System.out.println("Object Bapak (Superclass) :");

        objectBapak. show_variabel();

        objectAnak.c=5;

        System.out.println("Object Anak (Superclass dari Bapak):");

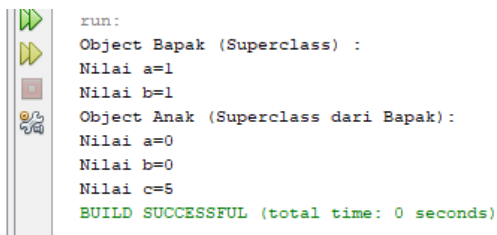
        objectAnak.show_variabel();

    }

}

```

• Output



```

run:
Object Bapak (Superclass) :
Nilai a=1
Nilai b=1
Object Anak (Superclass dari Bapak):
Nilai a=0
Nilai b=0
Nilai c=5
BUILD SUCCESSFUL (total time: 0 seconds)

```

Pada percobaan ke 7 menggunakan 3 buah class yaitu class Bapak sebagai parent class, kemudian class Anak sebagai Child class/ turunan dari class Bapak dan class InheritExample yang didalamnya terdapat main method untuk menampilkan output program yang telah dibuat.

Dalam program 7 menggunakan inheritance dan overriding pada method di class Parent dan Child class nya. Class Bapak disini mempunyai 2 atribut yaitu a dan b yang keduanya menggunakan tipe data int. Kemudian mempunyai method show_variabel yang menggunakan keyword void dan didalam method tersebut ada pemanggilan nilai variabel dari atribut yang sudah dibuat tadi. Berikut adalah syntax nya :

```
System.out.println("Nilai a="+ a);  
  
System.out.println("Nilai b="+ b);
```

Kemudian dalam class Anak menggunakan keyword extends yang berfungsi untuk mendapatkan sifat/atribut dari class Parent yang akan digunakan di class child. Dalam kelas Anak mempunyai 1 atribut yaitu c dengan tipe data int. kemudian mempunyai method yang di overriding dari class parent, methodnya sama bernama show_variabel dan di dalam method tersebut ada pemanggilan nilai dari atribut a dan b yang ada pada class Bapak dan atribut c yang ada class Anak. Berikut adalah syntax nya :

```
System.out.println("Nilai a="+ a);  
  
System.out.println("Nilai b="+ b);  
  
System.out.println("Nilai c="+ c);
```

Kemudian ada class InheritExample yang didalamnya ada main method. Dalam main method tersebut ada deklarasi pembuatan object pada class Bapak yang menggunakan keyword new. Berikut adalah syntaxnya :

```
Bapak objectBapak = new Bapak();
```

Kemudian ada deklarasi pembuatan object pada class Anak yang menggunakan keyword new. Berikut adalah syntaxnya :

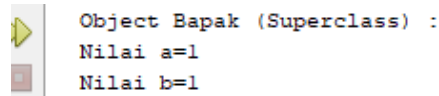
```
Anak objectAnak = new Anak();
```

Kemudian mengisi nilai a dan b yang ada pada class Bapak menggunakan objek yang tadi dibuat. Syntaxnya adalah :

```
objectBapak.a=1;  
  
objectBapak.b=1;
```

```
System.out.println("Object Bapak (Superclass) :");
```

Kemudian pemanggilan method `show_variabel` yang ada pada class Bapak. Dengan menggunakan syntax **`objectBapak.show_variabel()`**:. Output nya adalah :



```
Object Bapak (Superclass) :  
Nilai a=1  
Nilai b=1
```

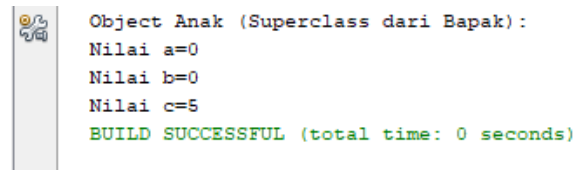
Dari gambar output diatas terlihat nilai a dan b sesuai dengan yang tadi telah diisi.

Kemudian selanjutnya mengisi nilai c yang ada pada class Anak menggunakan objek yang tadi dibuat. Syntaxnya adalah :

```
objectAnak.c=5;
```

```
System.out.println("Object Anak (Superclass dari Bapak):");
```

Kemudian pemanggilan method `show_variabel` yang ada pada class Anak. Dengan menggunakan syntax **`objectAnak.show_variabel()`**:. Output nya adalah :



```
Object Anak (Superclass dari Bapak):  
Nilai a=0  
Nilai b=0  
Nilai c=5  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Dapat terlihat bahwa nilai c sesuai dengan nilai yang tadi dimasukkan namun untuk nilai a dan b tidak sama denganyang tadi dimasukkan pada object class Bapak. Hal tersebut terjadi karena nilai a dan b yang dimasukkan menggunakan object class Bapak tadi tidak tersimpan dalam atribut yang ada pada kelas Bapak. Agar nilai a dan b yang ada pada class Anak sama dengan yang ada pada class Bapak maka harus dilakukan perubahan program dan menggunakan keyword `super`.

Program yang sudah diperbaiki :

```
class Bapak{  
  
    int a = 1;  
  
    int b = 1;  
  
    void show_variabel(){
```

```
        System.out.println("Nilai a="+ a);

        System.out.println("Nilai b="+ b);

    }

}
```

```
class Anak extends Bapak{

    int c;

    void show_variabel(){

        super.show_variabel();

        System.out.println("Nilai c="+ c);

    }

}
```

```
public class InheritExample {

    public static void main(String[] args) {

        Bapak objectBapak = new Bapak();

        Anak objectAnak = new Anak();

        System.out.println("Object Bapak (Superclass) :");

        objectBapak. show_variabel();

        objectAnak.c=5;

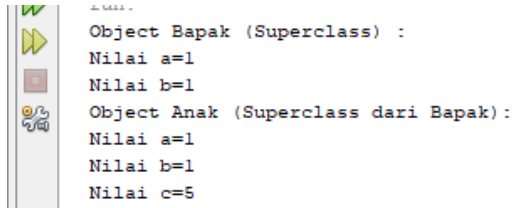
        System.out.println("Object Anak (Superclass dari Bapak):");

    }

}
```

```
        objectAnak.show_variabel();  
    }  
}
```

Output :



```
Object Bapak (Superclass) :  
Nilai a=1  
Nilai b=1  
Object Anak (Superclass dari Bapak):  
Nilai a=1  
Nilai b=1  
Nilai c=5
```

Dari output diatas terlihat bahwa nilai a dan b yang ada pada class Anak sama dengan yang ada pada class Bapak.