

Nama : Verdi Eza Irawan

NIM : 20210040106

Kelas : TI21E

Praktikum Pemograman Berorientasi Objek

1. Analisa Percobaan 1

Pada percobaan 1 ini dibuat 3 class yang dimana 1 class bernama Parent menjadi parent class dari class child lalu ada class public bernama NilaiX.

Pada kelas parent ada deklarasi nilai x, lalu pada class child ada pembuatan objek tes yang dimana ketika objek “tes” dibuat dan objek itu memanggil fungsi info maka output yang akan dihasilkan yaitu 20, 10, 5. Walaupun variabel yang sama yaitu “x” tetapi yang membedakan adalah pemanggilan variabelnya.

Jika hanya “x” saja yang dipanggil maka x itu hanya nilai dari parameter (jika itu dalam sebuah fungsi) karena itu bernilai 20. Jika “this.x” maka nilai yang diambil adalah nilai x yang menempel pada objek itu karena itu bernilai 10. Sedang “super.x” dia akan mengambil nilai pada parent class karena itu bernilai 5.

2. Analisa Percobaan 2

Pada percobaan 2 terdapat 1 class parent bernama pegawai dan 1 child class bernama Manajer.

Ketika di run terjadi eror yang disebabkan oleh variabel yang tidak sama dari class parent dan class child, pada class pegawai mendeklarasikan nama, dan gaji sementara pada method/fungsi IsiData yang ada di class Manajer memanggil variabel yang tidak dideklarasikan pada class Pegawai

Solusi untuk kasus ini adalah atribut nama pada kelas pegawai access modifier diganti dari private menjadi public. Dan pemanggilan nama pada fungsi IsiData diganti menjadi `super.nama = n`

3. Analisa Percobaan 3

Jika hanya memasukan code seperti pada gambar ketika di run maka tidak akan muncul output dikarenakan tidak ada method, dan jika kita menambahkan kode method dibawah kode yang ada digambar untuk print x maka yang terjadi akan eror hal itu disebabkan karena class child masih public, jika kata public dihapus yang artinya akan menjadi “class Child extends parent” maka method pada class anak dapat berjalan tanpa eror dan akan memunculkan nilai x yang sudah dideklarasikan. Jadi eror tersebut bukanlah disebabkan karena kosongnya constructor.

4. Analisa percobaan 4

Ketika dicoba ternyata tidak ada eror, setelah dianalisa semua penggunaan dan penulisan sudah benar. Pemanggilan objek pertama menggunakan konstruktor dengan 3 parameter yaitu nama, salary dan Dept sedangkan objek kedua menggunakan konstruktor dengan 2 parameter yaitu nama dan Dept.

5. Analisa Percobaan 5

Setelah dicoba untuk dijalankan program ini tidak ada eror atau masalah, program ini akan menjalankan kelas yang dibuat menjalankan fungsi – fungsi yang ada pada kelas.

6. Analisa percobaan 6

Pada percobaan 6 terdapat 2 kelas yaitu kelas A sebagai parent class dan kelas B sebagai subclass dari A, sub class A akan mengganti nilai var_a dan var_b dari parent kelas nya. Ketika objek B dibuat, constuktor A akan tetap dijalankan.

7. Analisa percobaan 7

Walaupun sudah menggunakan super pada kelas anak untuk mengakses nilai dari parent kelas, nilai a dan b dari kelas anak akan tetap 0 karena pada dasarnya blueprint nya bernilai 0. Jadi objek Anak tidak akan melakukan “override” pada objek Bapak, selama dalam bentuk Objek.

8. Analisa Percobaan 8

Pada kelas Baby menurunkan Parent. terdapat super() pada fungsi konstruktor yang akan mengoveride kelas parentnya. this.babyName = babyName untuk passing nilai babyName pada objek dengan parameter contruktor babyName .