



Mata Kuliah : Pemrograman Berorientasi Objek (PBO)
Program Studi : D4 – Sistem Informasi Bisnis / D4 – Teknik Informatika
Semester : 3

Kelas : TI - 2D
NIM : 244107020109
Nama : Aisya Aswy Nur Aidha
Jobsheet Ke- : 1

JAWABAN PERTANYAAN

1. Perbedaan antara objek dengan class

➤ Class

- Class merupakan blueprint (cetak biru) dari objek
- Berisi atribut dan method yang akan dimiliki oleh objek
- Tidak nyata (hanya berupa rancangan atau template)

➤ Objek

- Objek merupakan instance (perwujudan nyata) dari class
- Memiliki nilai konkret dari atribut dan dapat menjalankan method yang didefinisikan di class
- Nyata dalam program, dapat dibuat berkali kali dari satu class

➤ Analogi sederhana

- Class = resep kue (blueprint, hanya intruksi)
- Objek = kue yang jadi (hasil nyata yang bisa dimakan)

Jadi Class adalah definisi, sedangkan Objek adalah implementasi/hasil nyata dari definisi tersebut

2. Alasan mengapa warna dan tipe mesin dapat menjadi atribut dari objek mobil karena karakteristik utama yang membedakan setiap mobil, dengan menjadikan keduanya sebagai atribut, setiap objek mobil yang dibuat dari class dapat memiliki variasi unik, misalnya mobil berwarna merah dengan mesin bensin atau mobil hitam dengan mesin diesel, sehingga program dapat merepresentasikan kondisi dunia nyata secara lebih realistis.



3. Salah satu kelebihan utama dari pemrograman berorientasi objek dibanding dengan pemrograman struktural adalah program menjadi lebih modular, fleksibel dan mudah dikembangkan, lebih efisien untuk pengembangan sistem berskala besar dan mempermudah proses pemeliharaan program
4. Melakukan pendefinisian dua buah atribut dalam satu baris kode seperti “public String nama, alamat;” itu diperbolehkan tetapi tetap lebih baik dipisah agar kode lebih rapi, mudah dibaca dan gampang dikembangkan saat ada perubahan
5. Pada class SepedaGunung, atribut merk, kecepatan, dan gear tidak lagi di dalam class tersebut karena class tersebut merupakan turunan dari class Sepeda. Melalui konsep pewarisan (inheritance), semua atribut dan method yang ada di class induk (Sepeda) otomatis dimiliki oleh class turunannya (SepedaGunung). Dengan begitu, kita tidak perlu mendefinisikan ulang atribut tersebut. Hal ini membuat kode lebih sederhana, rapi, serta mudah dikelola, karena jika ada perubahan pada atribut di class Sepeda, class SepedaGunung juga akan ikut menyesuaikan tanpa harus menuliskan ulang.