

Pengantar Pemrograman Berorientasi Objek (PBO)

Selamat datang di dunia Pemrograman Berorientasi Objek (PBO)! Setelah kalian terbiasa dengan pemrograman prosedural, kini saatnya kita melangkah ke cara berpikir yang baru dan lebih kuat dalam membuat program. PBO adalah sebuah pendekatan yang memandang program sebagai kumpulan **objek** yang saling berinteraksi, bukan sekadar urutan instruksi yang dijalankan satu per satu.

Konsep Dasar PBO

Bayangkan kita ingin membuat program untuk mengelola data siswa. Dengan pemrograman prosedural, kita akan membuat banyak variabel terpisah (misalnya, namaSiswa, umurSiswa, alamatSiswa) dan serangkaian fungsi untuk memprosesnya. Semakin banyak siswa, semakin rumit programnya. PBO menawarkan solusi yang lebih rapi dengan memodelkan dunia nyata ke dalam kode.

Ada empat konsep dasar yang harus kalian pahami:

1. Objek (Object)

Objek adalah entitas di dunia nyata yang memiliki **keadaan** (data) dan **perilaku** (fungsi).

Contoh sehari-hari:

- **Mobil:**
 - **Keadaan (Data):** Warna, merek, kecepatan.
 - **Perilaku (Fungsi):** Maju, mundur, belok, mengerem.
- **Siswa:**
 - **Keadaan (Data):** Nama, NIS, kelas, alamat.
 - **Perilaku (Fungsi):** Belajar, bertanya, mengerjakan tugas.

Dalam PBO, objek adalah instans (wujud nyata) dari sebuah **kelas**.

2. Kelas (Class)

Kelas adalah sebuah cetak biru, *template*, atau rancangan yang digunakan untuk membuat objek. Kelas mendefinisikan **atribut** (keadaan) dan **metode** (perilaku) yang akan dimiliki oleh setiap objek yang dibuat darinya.

Contoh sehari-hari:

Bayangkan kelas sebagai cetakan kue kering berbentuk bintang. Kalian tidak bisa makan cetakan itu, tapi cetakan itu bisa digunakan untuk membuat banyak kue bintang (objek) yang sama. Setiap kue bintang yang kalian buat akan memiliki bentuk yang sama (atribut) dan bisa dimakan (perilaku), meskipun rasanya bisa berbeda jika bahan pembuatnya diganti.

Dalam kode, kelas Mobil akan mendefinisikan bahwa setiap objek Mobil harus memiliki atribut seperti warna dan merek, serta metode seperti maju() dan belok().

3. Atribut (Attribute)

Atribut adalah data atau properti yang mendefinisikan keadaan dari sebuah objek. Atribut sering disebut juga sebagai **variabel anggota** (*member variable*).

Contoh sehari-hari:

Jika kita memiliki objek mobil, atributnya bisa berupa:

- warna = "merah"
- merek = "Toyota"
- kecepatan = 60 km/jam

4. Metode (Method)

Metode adalah fungsi atau prosedur yang mendefinisikan perilaku atau tindakan yang dapat dilakukan oleh sebuah objek. Metode sering disebut juga sebagai **fungsi anggota** (*member function*).

Contoh sehari-hari:

Jika kita memiliki objek mobil, metodenya bisa berupa:

- maju(): Untuk mengubah nilai kecepatan menjadi lebih tinggi.

- `mengerem()`: Untuk mengubah nilai kecepatan menjadi lebih rendah.
 - `belokKanan()`: Untuk mengubah arah mobil.
-

Keuntungan Menggunakan PBO

Mempelajari PBO tidak hanya tentang sintaks baru, tetapi tentang cara berpikir untuk menyelesaikan masalah yang lebih besar. Berikut adalah beberapa keuntungan utama PBO:

1. **Kode Lebih Terstruktur dan Rapi:** Dengan mengelompokkan data (atribut) dan fungsi (metode) ke dalam satu unit (kelas), kode menjadi lebih mudah dipahami dan diorganisir.
 2. **Kode Mudah Digunakan Kembali (*Reusability*):** Setelah kita membuat sebuah kelas (misalnya, kelas Siswa), kita bisa menggunakannya berulang kali untuk membuat banyak objek siswa tanpa perlu menulis ulang kode. Ini menghemat waktu dan tenaga.
 3. **Memudahkan Pemeliharaan Program:** Jika ada bagian dari program yang perlu diubah, kita hanya perlu mengubah di dalam satu kelas, tanpa harus mengubah kode di banyak tempat. Hal ini mengurangi risiko munculnya *bug* baru.
 4. **Mempermudah Kolaborasi Tim:** Dalam proyek besar, setiap anggota tim bisa fokus mengerjakan kelasnya masing-masing. Mereka cukup tahu cara menggunakan kelas yang dibuat orang lain tanpa harus tahu detail internalnya.
-

Perbandingan dengan Pemrograman Prosedural

Fitur	Pemrograman Prosedural	Pemrograman Berorientasi Objek (PBO)
Pendekatan	Fokus pada serangkaian prosedur atau fungsi .	Fokus pada objek dan kelas yang saling berinteraksi.
Struktur Program	Terdiri dari fungsi-fungsi yang terpisah dan terbagi-bagi.	Terdiri dari kelas-kelas yang mengelompokkan data (atribut) dan perilaku (metode) menjadi satu unit.
Pengelolaan Data	Data dan fungsi terpisah. Data bisa diakses oleh	Data (atribut) dan fungsi (metode) terikat dalam satu

	banyak fungsi.	objek.
Contoh Kode	Kita membuat fungsi_input_nama(nama) dan variabel_nama.	Kita membuat kelas Siswa dengan atribut nama dan metode inputNama().
Skalabilitas	Menjadi sulit dikelola saat program bertambah besar.	Lebih mudah mengelola program yang kompleks dan besar.

Dengan PBO, kita tidak lagi berpikir tentang "apa yang harus kita lakukan selanjutnya?" melainkan "entitas apa saja yang terlibat dan apa yang bisa mereka lakukan?". Ini adalah pergeseran pola pikir yang akan sangat membantu kalian dalam mengembangkan program yang lebih canggih dan terstruktur.