

Nama: Aqsha Ramadhan Dimas Haryanto
NIM: H1D024089
SHIFT BARU: D
LAPORAN RESPONSI 2 PBO PERTEMUAN: 4

A. Alur kerja

1. Inisialisasi & Header Program dimulai dengan mencetak judul "DATA CUSTOMER BELANJAKU".
2. Pembuatan Objek Customer Biasa (Budi)
 - Instansiasi: Program membuat objek budi menggunakan class Customer.
 - Penyimpanan Data: Constructor Customer dijalankan untuk menyimpan Nama ("Budi"), ID ("CST-001"), dan Total Belanja (500.000).
 - Output: Program memanggil budi.tampilkanInfo(). Karena Budi adalah Customer biasa, program langsung menjalankan method di class Customer yang mencetak Nama, ID, dan Total Belanja.
3. Pembuatan Objek Member (Siti)
 - Instansiasi: Program membuat objek siti menggunakan class Member.
 - Penyimpanan Data Berjenjang (Chaining):
 1. Constructor Member menerima 5 data.
 2. Baris pertama super(...) memindahkan 3 data dasar (Nama, ID, Belanja) ke Constructor Customer untuk disimpan.
 3. Setelah urusan Parent selesai, Constructor Member menyimpan sisa 2 data (Poin, Level) ke variabel miliknya sendiri.
 4. Tampilan Info Member (Overriding + Super)
 - Program memanggil siti.tampilkanInfo().
 - Program masuk ke method milik Member (Child).
 - Langkah 1: Menemukan kode super.tampilkanInfo(). Program "melompat" sebentar ke class Customer untuk mencetak baris pertama (Nama, ID, Belanja).
 - Langkah 2: Kembali ke class Member untuk melanjutkan baris kode berikutnya, yaitu mencetak baris kedua (Poin, Level).

B. Fungsi yang digunakan

1. super(...) dalam Constructor
 - Implementasi: super(nama, id, totalBelanja);
 - Fungsi: Memanggil constructor milik Parent Class (Customer).
 - Kenapa penting? Karena Customer tidak memiliki constructor kosong, maka Member wajib menyetor data dasar saat pertama kali dibuat. Ini memastikan bahwa bagian "Customer" dari objek "Member" sudah siap sebelum data tambahan diisi.
2. super.namaMethod() dalam Method Overriding
 - Implementasi: super.tampilkanInfo();
 - Fungsi: Memanggil method asli milik Parent Class dari dalam method Child Class.
 - Manfaat: Mencegah duplikasi kode (*Don't Repeat Yourself*). Kita tidak perlu menulis ulang perintah System.out.println untuk Nama dan ID di class Member. Kita cukup "meminjam" tampilan yang sudah dibuat di Customer, lalu menambahkan hiasan tambahan (Poin & Level).
3. Method Overriding
 - Implementasi: Method tampilkanInfo() ada di Customer dan ditulis ulang di Member.
 - Fungsi: Memberikan perilaku spesifik pada subclass. Meskipun nama methodnya sama, Member memiliki cara menampilkan data yang lebih detail dibandingkan Customer biasa.
4. Protected Access Modifier

- Implementasi: protected String nama;
- Fungsi: Variabel di Customer menggunakan protected agar class Member memiliki akses langsung terhadap data tersebut jika sewaktu-waktu dibutuhkan (meskipun dalam kasus tampilkanInfo ini kita mengaksesnya lewat method super, namun secara struktur data, Member tetap bisa melihat variabel tersebut).

Hasil Output Program:

```
[Running] cd "d:\Responsi-2\pert4\" && javac UjiCustomer.java && java UjiCustomer
== DATA CUSTOMER BELANJAKU ==
Status: Customer Biasa
Nama: Budi Santoso | ID: CST-001 | Total Belanja: Rp 500000

Status: Member
Nama: Siti Aminah | ID: MBR-110 | Total Belanja: Rp 1250000
Poin Reward: 240 | Level: Gold
```