**TRANSIENT**S

erialisasi: Jika sebuah variabel dideklarasikan sebagai transient, maka variabel tersebut akan diabaikan saat objek serialisasi. Dengan kata lain, nilai dari variabel transient tidak akan disertakan dalam byte stream yang merepresentasikan objek tersebut.

Deserialisasi: Ketika objek di-deserialisasi, variabel yang diberi keyword transient akan memiliki nilai default sesuai dengan tipe datanya. Misalnya, untuk tipe data objek seperti String, transient akan memiliki nilai null, untuk tipe primitif seperti int, transient akan memiliki nilai 0, dan untuk tipe data boolean, transient akan memiliki nilai false.

**STATIC**

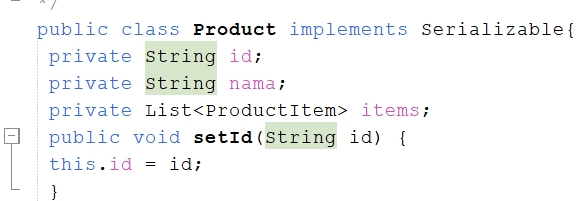
Serialisasi: Variabel static tidak disertakan dalam proses serialisasi. Ini karena variabel static tidak terkait dengan instance objek, melainkan dengan kelas itu sendiri. Oleh karena itu, nilai dari variabel static tidak relevan untuk disimpan bersama dengan data objek saat serialisasi.

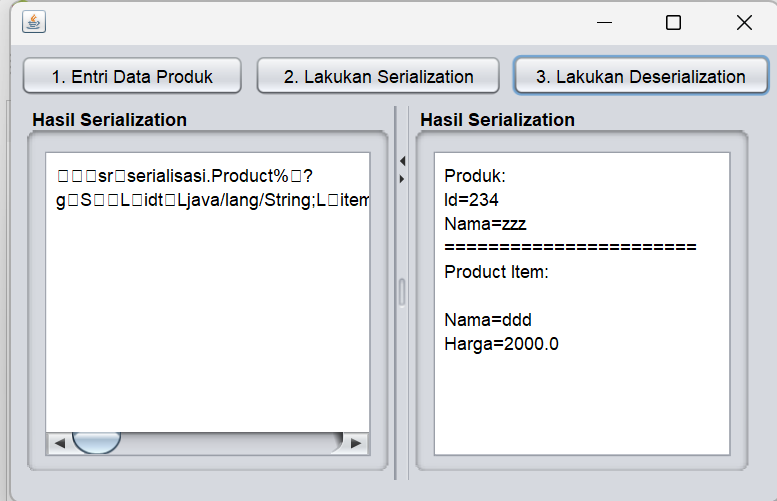
Deserialisasi: Setelah objek di-deserialisasi, variabel static akan mempertahankan nilai yang sama seperti sebelum proses serialisasi. Ini berlaku untuk semua instance dari kelas, karena variabel static bersifat bersama antar semua instance dari kelas yang sama.

**FINAL** Serialisasi: Variabel final yang bukan merupakan variabel static akan disertakan dalam proses serialisasi dan nilainya akan disimpan dalam byte stream. Ini berlaku meskipun variabel final tidak memiliki setter.

Deserialisasi: Setelah objek di-deserialisasi, variabel final akan memiliki nilai yang sama seperti saat objek tersebut di-serialisasi. Karena variabel final tidak dapat diubah setelah pertama kali diinisialisasi, nilai tersebut akan tetap konstan setelah deserialisasi.

**Contoh dan hasil sebelum ada transient,static,dan final**





Hasil setelah ada transient, Static, Final

