

NAMA : NINIK RAHAYU
NIM : H1D024024
SHIFT AWAL : I
SHIFT AKHIR : B

PERTEMUAN 7

Alur Kerja

Program diawali dengan pendefinisian interface `PaymentMethod` yang berfungsi sebagai kontrak berisi daftar method yang wajib dimiliki oleh semua metode pembayaran, yaitu `processPayment()`, `getPaymentDetails()`, `getTransactionFee()`, dan `getBalance()`. Setelah itu dibuat class `EWalletPayment` yang mengimplementasikan interface tersebut, sehingga class ini wajib mengisi seluruh method yang didefinisikan di `PaymentMethod`. Di dalam `EWalletPayment` terdapat atribut `namaLayanan`, `saldoPengguna`, dan `nominalPembayaran` yang diisi melalui constructor saat objek dibuat. Method `processPayment()` bekerja dengan cara menghitung total pembayaran ($\text{nominal} + \text{biaya transaksi}$), lalu mengecek apakah saldo cukup; jika cukup, saldo dikurangi dan menampilkan pesan berhasil, jika tidak cukup maka menampilkan pesan gagal. Method `getPaymentDetails()` mengembalikan informasi layanan e-wallet, `getTransactionFee()` mengembalikan biaya transaksi tetap, dan `getBalance()` mengembalikan saldo saat ini.

Selanjutnya program dijalankan melalui class `PaymentTest` pada method `main`. Di dalam `main`, pertama ditampilkan judul program, lalu dibuat objek `EWalletPayment` dengan data awal (`nama layanan`, `saldo awal`, dan `nominal pembayaran`). Setelah itu program menampilkan saldo awal, memproses pembayaran dengan memanggil `processPayment()`, menampilkan hasil transaksi (berhasil atau gagal), menampilkan sisa saldo setelah transaksi, dan terakhir menampilkan detail transaksi melalui `getPaymentDetails()`. Dengan alur ini, seluruh proses pembayaran e-wallet dari awal sampai akhir dapat diuji secara lengkap.

Fungsi Yang Digunakan

1. `String processPayment()`: Mengembalikan teks status keberhasilan atau kegagalan transaksi.
2. `String getPaymentDetails()`: Mengembalikan teks informasi layanan pembayaran (OVO).
3. `double getTransactionFee()`: Mengembalikan angka biaya admin (2000).
4. `double getBalance()`: Mengembalikan angka sisa saldo pengguna.
5. `double getNominalPembayaran()`: Mengembalikan angka nominal yang akan dibayar.

Prosedur Yang Digunakan

1. `tampilInfo()`
Digunakan pada `defaultMesin`, `mesinMotor`, `mesinTraktor`, dan `mesinTraktorListrik` untuk menampilkan informasi mesin sesuai jenisnya.
2. `suaraMesin()`
Digunakan pada `mesinMotor`, `mesinTraktor`, dan `mesinTraktorListrik` untuk menampilkan suara khas masing-masing mesin.

Constructor Yang Digunakan

1. EWalletPayment(String namaLayanan, double saldoPengguna, double nominalPembayaran)
Digunakan untuk mengisi awal nama layanan, saldo, dan nominal pembayaran saat objek dibuat.

Interface Yang Digunakan

Interface PaymentMethod berisi daftar method yang wajib dimiliki oleh class pembayaran, yaitu processPayment(), getPaymentDetails(), getTransactionFee(), dan getBalance(). Class EWalletPayment mengimplementasikan interface ini sehingga wajib mengisi semua method tersebut agar sistem pembayaran berjalan sesuai standar.

Output

```
D:\java\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2025.2.1\lib\idea_rt.jar=60854" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -[
=== PROGRAM SISTEM PEMBAYARAN (E-WALLET) ===
Saldo awal: 150000
Memproses pembayaran sebesar 50000...
Pembayaran berhasil!
Sisa saldo: 98000
Detail Transaksi: Pembayaran dilakukan melalui OVO
Process finished with exit code 0
```