

Jobsheet-2: Variabel, Tipe Data, dan Operator pada Dart



NIM : 2341760017
Nama : Marga Reta Novia Putri
Kelas : SIB-3D / 10

Tugas Praktikum

1. Buat program yang menerapkan semua jenis operator yang telah dipelajari dalam satu aplikasi konsol

Jawab:

- **Aritmatika (perhitungan kasir)**

```
double hargaItem1 = 25.000;  
double hargaItem2 = 40.000;  
int qty1 = 2;  
int qty2 = 1;  
double subtotal = (hargaItem1 * qty1) + (hargaItem2 * qty2);  
print('Subtotal: ${(hargaItem1 * qty1)} + ${(hargaItem2 * qty2)} = $subtotal');
```

- Operasi * dan + menghitung total tiap item dan menjumlahkannya ke subtotal.
- Gunakan subtotal sebagai dasar semua perhitungan selanjutnya.

- **Diskon (ternary operator)**

```
double diskon = subtotal > 50.000 ? subtotal * 0.1 : 0;  
double totalBayar = subtotal - diskon;  
print('Diskon: $diskon');  
print('Total Bayar: $totalBayar');
```

- ? : adalah operator ternary: jika subtotal > 50.000 benar, diskon 10% diterapkan; jika tidak, diskon = 0.
- totalBayar = subtotal dikurangi diskon.

- **Increment/Decrement**

```
int jumlahCustomer = 0;  
print('\nJumlah customer awal: $jumlahCustomer');  
jumlahCustomer++; // pre-order datang  
print('Customer datang, jumlah ++ => $jumlahCustomer');  
jumlahCustomer--; // cancel  
print('Ada yang batal, jumlah -- => $jumlahCustomer');
```

- ++ menambah 1; -- mengurangi 1.
- Saat dipakai sendiri (bukan di dalam ekspresi), hasilnya langsung menaikkan/menurunkan variabel.

- **Relasional (perbandingan)**

```
bool isDiskonBesar = diskon >= 10.000;  
print('\nDiskon >= 10.000 ? $isDiskonBesar');
```

- Operator relasional: >=, >, <, ==, != menghasilkan bool (true/false).

- isDiskonBesar merepresentasikan kondisi diskon memenuhi kriteria.

- Logika

```
bool member = true;
bool belanjaBanyak = subtotal > 100.000;
bool dapatBonus = member && belanjaBanyak;
print('Member && BelanjaBanyak => $dapatBonus');
```

- && (logical AND) menggabungkan dua kondisi; keduanya harus true agar hasil true.

- Assignment & Compound assignment

```
double kas = 100.000;
print('\nKas awal: $kas');
kas += totalBayar; // tambah hasil penjualan
print('Kas += totalBayar => $kas');
kas -= 5_000; // pengeluaran kecil
print('Kas -= 5000 => $kas');
```

- +=, -=, *=, /= (integer division assignment), dll. mempersingkat penugasan nilai baru.

- Null-aware operators

```
String? catatanPromo;
// catatanPromo = "Spesial member minggu ini!";
String tampilPromo = catatanPromo ?? 'Tidak ada promo';
print('\nPromo hari ini: $tampilPromo');
// jika belum diisi, isi default
catatanPromo ??= 'Promo default minggu depan';
print('catatanPromo setelah ??= : $catatanPromo');
```

- String? berarti variabel boleh null.
- ?? memberikan nilai default jika kiri null (catatanPromo ?? 'X').
- ??= hanya men-set variabel jika saat ini null.

- Type test

```
Object dataInput = 123.45;
if (dataInput is double) {
    print('\nInput bertipe double dengan ceil: ${dataInput.ceil()}');
} else if (dataInput is int) {
    print('Input bertipe int');
}
```

- is memeriksa tipe runtime; is! kebalikannya.
- Setelah is benar, Anda aman melakukan operasi tipe spesifik (casting implicit).

- Kondisional (if/else, ternary, switch)

```
String kategori;
if (subtotal >= 100.000) {
    kategori = 'Premium';
} else if (subtotal >= 50.000) {
    kategori = 'Standar';
} else {
    kategori = 'Ekonomis';
}
print('\nKategori belanja: $kategori');

// Ternary singkat
String statusLunas = (totalBayar <= kas) ? 'Lunas' : 'Piutang';
print('Status: $statusLunas');

// Switch
String hari = 'Sabtu';
switch (hari) {
    case 'Senin':
    case 'Selasa':
        print('Awal minggu kerja');
        break;
    case 'Sabtu':
    case 'Minggu':
        print('Weekend promo!');
        break;
    default:
        print('Hari biasa');
}
```

- if/else untuk cabang logika bertingkat.
- Ternary ?: untuk penugasan singkat berdasar kondisi.
- switch cocok kalau memeriksa satu nilai terhadap banyak case.

- Bitwise Operator

```
print('\nBitwise Demo:');
int kodePromo = 10;
int masker = 12;
print(kodePromo.toRadixString(2)); // "1010"
print('kodePromo & masker -> ${(kodePromo & masker).toRadixString(2)}');
print('kodePromo | masker -> ${(kodePromo | masker).toRadixString(2)}');
print('kodePromo ^ masker -> ${(kodePromo ^ masker).toRadixString(2)}');
print('~kodePromo -> ${(~kodePromo).toRadixString(2)}');
print('kodePromo << 1 -> ${(kodePromo << 1).toRadixString(2)}');
print('masker >> 2 -> ${(masker >> 2).toRadixString(2)}');
```

- & | ^ bekerja bit-per-bit. << dan >> geser bit.
- ~ adalah komplement bit — di Dart menghasilkan nilai bertanda (hasil bisa negatif karena representasi dua komplement).

2. Buat program Dart sederhana untuk menghitung BMI (Body Mass Index) berdasarkan berat dan tinggi yang diinput

Jawab:

```
void main() {  
  double berat = 50;      // contoh berat (kg)  
  double tinggi = 1.5;    // contoh tinggi (meter)  
  
  double bmi = berat / (tinggi * tinggi);  
  
  print('BMI Anda: ${bmi.toStringAsFixed(2)}');  
  
  if (bmi < 18.5) {  
    print('Kategori: Kekurangan berat badan');  
  } else if (bmi >= 18.5 && bmi < 24.9) {  
    print('Kategori: Normal (ideal)');  
  } else if (bmi >= 25 && bmi < 29.9) {  
    print('Kategori: Kelebihan berat badan');  
  } else {  
    print('Kategori: Obesitas');  
  }  
}
```

Link Github: <https://github.com/margaretanp/Pemrograman-Mobile>