

FUNCTION

Apa itu Function?

Bayangkan *function* seperti mesin kecil di dalam program. *Function* menerima input, memprosesnya, dan menghasilkan output. Dengan *function*, kamu bisa mengelompokkan kode yang melakukan tugas tertentu, sehingga kodemu jadi lebih mudah dibaca, diuji, dan digunakan kembali.

Struktur Function

```
returnType functionName(parameter1, parameter2, ...) {  
    // Kode yang akan dijalankan  
    return value;  
}
```

- returnType: Tipe data yang dihasilkan oleh *function* (misalnya, int, String, bool, atau void jika tidak menghasilkan nilai).
- functionName: Nama *function* yang unik dan deskriptif.
- parameter: Input yang diterima oleh *function* (opsional).
- return value: Nilai yang dihasilkan oleh *function* (opsional).

Contoh Kode

```
int tambah(int a, int b) {  
    int hasil = a + b;  
    return hasil;  
}  
  
void sapa(String nama) {  
    print('Halo, $nama!');  
}
```

Jenis-jenis Function

- **Function dengan Parameter:** Menerima input berupa parameter.

```
void tambah(int a, int b) {  
    print(a + b);  
}  
  
void main() {  
    tambah(5, 3); // Output: 8  
}
```

- **Function tanpa Parameter:** Tidak menerima input.

```
void sapa() {  
    print("Halo!");  
}  
  
void main() {  
    sapa(); // Output: Halo!  
}
```

- **Function dengan *Return Value*:** Menghasilkan nilai.

```
String sapa() {  
    return "Halo!";  
}  
  
void main() {  
    print(sapa()); // Output: Halo!  
}
```

- **Function tanpa *Return Value*:** Tidak menghasilkan nilai (void).

```
void cetakNama(String nama) {  
    print("Nama saya $nama");  
}  
  
void main() {  
    cetakNama("Cika"); // Output: Nama saya Cika  
}
```

- ***Optional Parameter*:** Parameter yang boleh diisi atau tidak.

```
String salam(String nama, [String? ucapan]) {  
    return "Halo, $nama! ${ucapan ?? ''}";  
}  
  
void main() {  
    print(salam("Andi")); // Output: Halo, Andi!  
    print(salam("Andi", "Selamat pagi")); // Output: Halo, Andi! Selamat pagi  
}
```

- **Named Parameter:** Parameter yang dipanggil dengan namanya.

```
void info(String nama, {int? umur}) {  
    print("Nama: $nama");  
    if (umur != null) {  
        print("Umur: $umur");  
    }  
}  
  
void main() {  
    info("Budi"); // Output: Nama: Budi  
    info("Budi", umur: 20); // Output: Nama: Budi, Umur: 20  
}
```

- **Arrow Function:** Penulisan *function* yang lebih ringkas.

```
int kaliDua(int angka) => angka * 2;  
  
void main() {  
    print(kaliDua(5)); // Output: 10  
}
```



UNIT KEGIATAN MAHASISWA PROGRAMMING UNIVERSITAS MULTI DATA PALEMBANG

Jalan Rajawali No. 14, 9 Ilir, Ilir Timur III Palembang 30113
Web: ukmprogramming.mdp.ac.id Email: procom@mdp.ac.id



Latihan

- Buat *function* untuk menghitung luas persegi panjang.
- Buat *function* untuk mengecek apakah sebuah angka genap atau ganjil.
- Buat *function* untuk menampilkan daftar nama.

Tips

- Berikan nama *function* yang jelas dan deskriptif.
- Gunakan parameter untuk membuat *function* lebih fleksibel.
- Gunakan return untuk menghasilkan nilai dari *function*.
- Bagi kode yang kompleks menjadi beberapa *function* yang lebih kecil.

Dengan memahami *function*, kamu bisa membuat kode Dart yang lebih terstruktur, efisien, dan mudah dipelihara. Selamat belajar!