



Types of Function

ชญาณิช น้อยพันธ์ 630710319

Types of Function in Dart 01

Types of Function in C 02

Types of Function in Java 03

Types of Function in Python 04

Reference 05

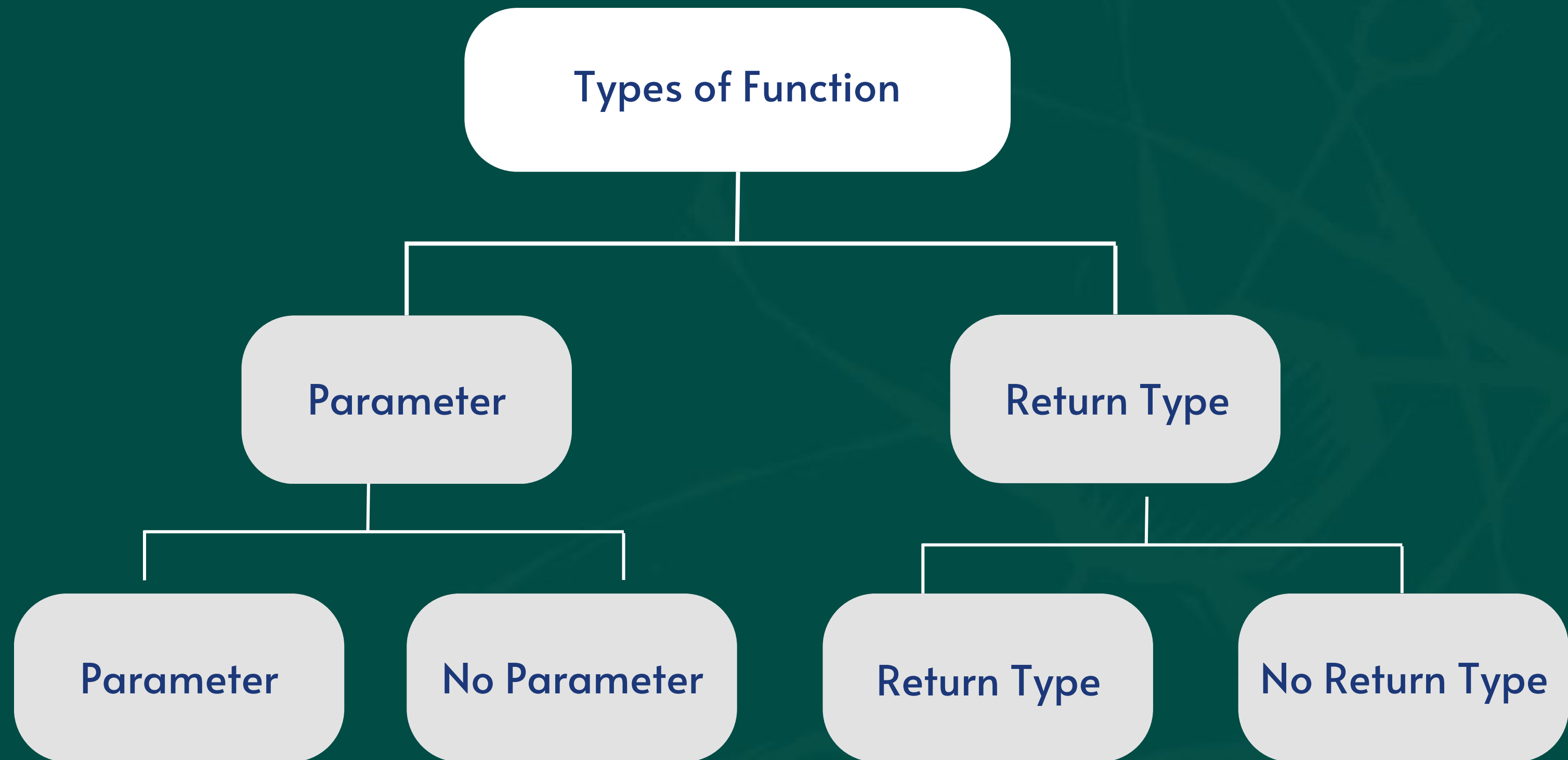


Types of Function in Dart

Function คือ block ของ code ที่จะรันเมื่อถูกเรียก เราสามารถใส่ data ที่เรียกว่า parameters เข้าไปในฟังก์ชันได้ โดยฟังก์ชันจะคืนค่า (return) เป็นผลลัพธ์

ประเภทของ Function มีดังนี้

1. No Parameter And No Return Type
2. Parameter And No Return Type
3. No Parameter And Return Type
4. Parameter And Return Type



Function With No Parameter And No Return Type

- ฟังก์ชันนี้ไม่ต้องส่งพารามิเตอร์และ
ฟังก์ชัน
- ไม่มีการคืนค่าจาก
ฟังก์ชัน

Example:

```
void main() {  
    printName();  
}  
void printName() {  
    print("My name is John Doe.");  
}
```

Output:

```
---> My name is John Doe.
```

จากตัวอย่าง printName() คือฟังก์ชันที่มีคีย์เวิร์ด void หมายความว่าไม่มี
ประเภทการส่งคืน และวงเล็บว่างแสดงว่าไม่มีพารามิเตอร์ที่ส่งผ่านไปยังฟังก์ชัน

Function With Parameter And No Return Type

- ฟังก์ชันนี้ต้องส่งพารามิเตอร์และ
ไม่มีการคืนค่าจากฟังก์ชัน

Example:

```
void add(int a, int b) {  
    int sum = a + b;  
    print("The sum is $sum");  
}  
void main() {  
    int num1 = 10;  
    int num2 = 20;  
    add(num1, num2);  
}
```

Output:

```
---> The sum is 30
```

จากตัวอย่าง `add(int a, int b)` คือฟังก์ชันที่มีคีย์เวิร์ด `void` หมายความว่าไม่มีประเภทการส่งคืน และวงเล็บไม่ว่าง แสดงว่ามีการส่งค่าพารามิเตอร์ไปยังฟังก์ชัน จากตัวอย่างส่งพารามิเตอร์ประเภท `int`

Function With No Parameter And Return Type

- ฟังก์ชันนี้ไม่ต้องส่งพารามิเตอร์
และมีการคืนค่าจากฟังก์ชัน

Example:

```
void main() {  
    String name = primeMinisterName();  
    print("The Name from function is $name.");  
}  
String primeMinisterName() {  
    return "John Doe";  
}
```

Output:

```
---> The name from function is John Doe
```

จากตัวอย่าง primeMinisterName() คือฟังก์ชันที่มีคีย์เวิร์ด String นำหน้าชื่อฟังก์ชัน หมายความว่าส่งกลับค่า String และวงเล็บว่างแสดงว่าไม่มีพารามิเตอร์ที่ถูกส่งผ่านไปยังฟังก์ชัน

Function With Parameter And Return Type

- ฟังก์ชันที่ต้องส่งพารามิเตอร์ และมีการคืนค่าจากฟังก์ชัน

Example:

```
double calculateInterest(double principal, double rate, double time) {  
    double interest = principal * rate * time / 100;  
    return interest;  
}  
void main() {  
    double principal = 5000;  
    double time = 3;  
    double rate = 3;  
    double result = calculateInterest(principal, rate, time);  
    print("The simple interest is $result.");  
}
```

Output:

```
---> The simple interest is 450.0.
```

จากตัวอย่าง `calculateInterest(double principal, double rate, double time)` คือ ฟังก์ชันที่มีคีย์เวิร์ด `double` นำหน้าชื่อฟังก์ชัน หมายความว่าส่งกลับค่า `double` และ วงเล็บไม่ว่าง แสดงว่ามีการส่งค่าพารามิเตอร์ไปยังฟังก์ชัน จากตัวอย่างส่งพารามิเตอร์ ประเภท `double`

Types of Functions in C

ฟังก์ชันในภาษา C แบ่งออกเป็น 2 ประเภทใหญ่ๆ คือ

1. Predefined Functions (หรือที่เรียกว่า library functions)
2. User-defined functions (กำหนดโดยผู้ใช้)

Predefined Functions

คือ ฟังก์ชันที่กำหนดไว้ในไลบรารีของระบบแล้ว ซึ่งผู้ใช้สามารถเรียกใช้ได้แต่จะต้องมีความรู้เกี่ยวกับการทำงานในการใช้ฟังก์ชันประเภทนี้จะต้องรวมไฟล์ส่วนหัว เช่น ต้องการใช้ฟังก์ชัน `printf()` จะมีอยู่ในไฟล์ส่วนหัว `<stdio.h>`

Example 1:

```
#include <stdio.h>           // header file is included in which printf() function exists
int main(){
    printf("Hello everyone! I am a bot."); // printf() function is used for printing the sentence
}
```

Output:

```
---> Hello everyone! I am a bot.
```

Predefined Functions

จากตัวอย่าง มีการเรียกใช้ฟังก์ชันมากกว่า 1 ฟังก์ชัน ซึ่งเราต้องการรับค่าอินพุต หารากที่สองและพิมพ์ผลลัพธ์ จึงต้องมีการใช้สามฟังก์ชันดังนี้

printf(): มีอยู่ใน <stdio.h>

scanf() : มีอยู่ใน <stdio.h>

sqrt() : มีอยู่ใน <math.h> ไฟล์ส่วนหัวทั้งหมดเหล่านี้จะต้องรวมไว้ที่จุดเริ่มต้นของโปรแกรม

Example 2:

```
#include <stdio.h>
#include <math.h>

int main(){
    double num, root;

    printf("enter the number you want to find square root of: "); // printf function is used to take input
    scanf("%lf", &num);

    root = sqrt(num); // square root is done and stored in root

    printf(" The square root of %.2lf is %.2lf.", num, root);
    return 0;
}
```

Output:

```
---> enter the number you want to find square root of: 225
      The square root of 225.00 is 15.00.
```

Example:

```
#include<stdio.h>

// Declaration
void Add();

void main(){
    Add();
}

void Add(){
    int Sum, x = 15, y = 50;
    Sum = x + y;

    printf("\n Total sum of x = %d and y = %d is = %d", x, y, Sum);
}
```

Output:

```
---> Total sum of x = 15 and y = 50 is = 65
```

Function With No Parameter And No Return Type

Example:

```
#include<stdio.h>

void Add(int, int);

void main(){
    int x, y;

    printf("\n Enter two integer values to add: \n");
    scanf("%d %d",&x, &y);

    // dynamic values are called
    Add(x, y);
}

void Add(int x, int y){
    int Sum;

    Sum = x + y;

    printf("\n Total sum of %d and %d is = %d \n", x, y, Sum);
}
```

Output:

```
---> Enter two integer values to add:
10
20
Total sum of 10 and 20 is = 30
```

Function With Parameter And No Return Type

Example:

```
#include<stdio.h>

int Multiply();
int main(){
    int Mul;
    Mul = Multiply();    // no argument passed
    printf("\n The multiplication of x and y is = %d \n", Mul );
    return 0;
}

int Multiply(){
    int Mul, x = 5, y = 10;
    Mul = x * y;

    return Mul;
}
```

Output:

```
---> The multiplication of x and y is = 50
```

Function With No Parameter And Return Type

Example:

```
#include <stdio.h>
void main(){
    int sub(int,int); // return value and arguments of function
    int a=15,b=7;
    int result = sub(a,b);
    printf("a-b = %d",result);
}

int sub(int x,int y){
    return(x-y); // this is return value,'int type'
}
```

Output:

```
---> a-b = 8
```

Function With Parameter And Return Type

Types of Functions in Java

ฟังก์ชันในภาษา Java แบ่งออกเป็น 2 ประเภท คือ

1. Built-in Functions : ฟังก์ชันเหล่านี้ถูกกำหนดไว้ล่วงหน้า และเราสามารถใช้งานได้ทุกเมื่อที่ต้องการในโปรแกรม Java ของเรา ตัวอย่างเช่น `pow()`, `sqrt()`, `min()` เป็นต้น
2. User Defined Functions : ฟังก์ชันเหล่านี้ถูกกำหนดหรือสร้างโดยโปรแกรมเมอร์สำหรับการทำงานเฉพาะในโปรแกรม

Example:

```
public class Example
{
    public static void message()
    {
        System.out.println("Hello I am learning how to create void function in Java.");
    }

    public static void main(String args[])
    {
        message();
    }
}
```

Output:

```
---> Hello I am learning how to create void function in Java.
```

**Function With No Parameter
And No Return Type**

Example:

```
public class Example
{
    public static void sum(int a, int b)
    {
        int c;
        c=a+b;
        System.out.println("Sum of " + a + " and " + b + " is " + c);
    }

    public static void main(String args[])
    {
        sum(10,20);
    }
}
```

Output:

```
---> Sum of 10 and 20 is 30
```

**Function With Parameter
And No Return Type**

Example:

```
public class Example
{
    public static float pi()
    {
        return 3.142f;
    }
    public static void main(String args[])
    {
        System.out.println("The value of PI is 22/7 = " + pi());
    }
}
```

Output:

```
---> The value of PI is 22/7 = 3.142
```

Function With No Parameter
And Return Type

Example:

```
public class Example
{
    public static int sum(int a, int b)
    {
        int c;
        c=a+b;
        return c;
    }

    public static void main(String args[])
    {
        int x=10, y=20, z;
        z=sum(x,y);
        System.out.println("Sum of " + x + " and " + y + " is " + z);
    }
}
```

Output:

```
---> Sum of 10 and 20 is 30
```

Function With Parameter
And Return Type

Types of Functions in Python

ฟังก์ชันในภาษา Python แบ่งออกเป็น 2 ประเภท คือ

1. Standard library functions : เป็นฟังก์ชันในตัว Python ที่พร้อมใช้งาน เช่น `sqrt()`, `pow()`, `print()`
2. User Defined Functions : ฟังก์ชันที่เราสามารถสร้างการทำงานตามความต้องการของเราได้

Example:

```
def geet1():  
    print('\nHello!')  
  
geet1()
```

Output:

```
---> Hello!
```

Function With No Parameter
And No Return Type

Example:

```
def geet2(name):  
    print('\nHello, {} !'.format(name))  
  
geet2('Jane')
```

Output:

```
---> Hello, Jane !
```

Function With Parameter
And No Return Type

Example:

```
def geet3():  
    greeting = '\nHello !'  
    return greeting  
  
grtng = geet3()  
print(grtng)
```

Output:

```
---> Hello !
```

Function With No Parameter
And Return Type

Example:

```
def geet4(name):  
    greeting = '\nHello, {}'.format(name)  
    return greeting  
  
grtng = geet4('Bob')  
print(grtng)
```

Output:

```
---> Hello, Bob
```

Function With Parameter
And Return Type

Reference

https://www.programiz.com/c-programming/types-user-defined-functions#google_vignette

<https://www.scaler.com/topics/types-of-functions-in-c/>

<https://www.dremendo.com/java-programming-tutorial/java-function>

<https://www.programiz.com/python-programming/function>

https://web.engr.oregonstate.edu/~webbky/ENGR103_files/Section%206%20User-Defined%20Functions.pdf